UNITED STATES DISTRICT COURT
EASTERN DISTRICT OF TEXAS
MARSHALL DIVISION

UNIVERSAL CONNECTIVITY TECHNOLOGIES INC., Plaintiff, v. LENOVO GROUP LIMITED, Defendant.	Case No. 2:23-cv-00449-JRG

DEFENDANT'S INVALIDITY AND ELIGIBILITY CONTENTIONS

Pursuant to the Court's Patent Rules, Standing Order Regarding Subject Matter Eligibility Contentions, and Docket Control Order (Dkt. No. 27), Defendant Lenovo Group Limited respectfully submits these invalidity and eligibility contentions with respect to the asserted claims of U.S. Patent Nos. 7,154,905 ("the '905 Patent"), 7,187,307 ("the '307 Patent"), 7,746,798 ("the '798 Patent"), 9,232,265 ("the '265 Patent"), 8,680,712 ("the '712 Patent"), 7,856,520 ("the '520 Patent"), 7,921,231 ("the '231 Patent"), and 9,852,103 ("the '103 Patent") (collectively, the "Asserted Patents") identified by Plaintiff Universal Connectivity Technologies Inc., ("UCT" or "Plaintiff") in its May 15, 2024 Patent Rule 3-1 Disclosures and July 3, 2024 Notice Regarding Claim Reduction (together, "Infringement Contentions").

The currently Asserted Claims, as reflected in Plaintiff's Infringement Contentions, are:

- '905 Patent: Claims 1, 8, 12, 15, 16, 17, 21, 22, 23, 26
- '307 Patent: Claims 1, 7, 23, 31, 53, 68, 71, 72, 74, 75
- '798 Patent: Claims 11, 19, 20, 21, 22, 23, 24, 25, 26, 28

- ’265 Patent: Claims 1, 8, 11, 14, 16
- ’712 Patent: Claims 1, 2, 5, 7, 9
- ’520 Patent: Claims 1, 4, 8, 9, 12, 13, 15, 18, 19, 25
- ’231 Patent: Claims 1, 2, 10, 11, 12, 13, 14, 16, 17, 18
- ’103 Patent: Claims 1, 2, 3, 11, 12, 13, 17, 20, 21, 22

As detailed further below, the asserted claims of each Asserted Patent are anticipated by and/or obvious in view of one more of the prior art references being produced concurrently pursuant to 35 U.S.C. § 102 and/or § 103, are directed to ineligible subject matter under 35 U.S.C. § 101, and are otherwise invalid under 35 U.S.C. § 112.

I. PRELIMINARY STATEMENT

These invalidity contentions and eligibility contentions are based on Defendant’s current knowledge, understanding, and belief of the Asserted Patents and prior art, of Plaintiff’s infringement theories (inasmuch as they can be inferred from its Infringement Contentions), and of the facts and other information available as of the date of these contentions. Defendant’s investigation, discovery, and analysis of information related to this action is ongoing. Additional discovery, elucidation of Plaintiff’s impermissibly vague infringement contentions, Plaintiff’s expert report(s) regarding infringement, and/or orders of the Court may require Defendant to amend or supplement these invalidity contentions and eligibility contentions, and Defendant expressly reserves the right to do so, subject to the Court’s procedures, as the case proceeds. These contentions represent Defendant’s good-faith effort to provide a comprehensive identification of prior art relevant to this case, but Defendant reserves the right to modify or supplement its prior art list and invalidity contentions and eligibility contentions at a later time with, or based upon, pertinent information that may be subsequently discovered.

Universal Connectivity Technologies Inc.
Ex. 2002 - Page 2

In addition to the prior art that charted in the attached exhibits, additional background references are discussed herein and/or included in the document production accompanying these Invalidity and Eligibility Contentions. Defendant reserves the right to rely on these background references for any purpose including, for example, to demonstrate the knowledge of a POSITA at the relevant time.

To the extent UCT believes anything in these contentions (including the attached claim chart exhibits and accompanying document production) are insufficient, Defendant invites UCT to bring such issues to its attention as soon as reasonably possible.

A. No Waiver

Nothing in these contentions is intended, nor should be construed, as a waiver of any non-infringement position or argument under 35 U.S.C. §§ 101 or 112. To the contrary, as set forth herein, Defendant contends that all Asserted Claims are invalid pursuant to 35 U.S.C. §§ 101 and/or 112. Defendant's statements herein (including the accompanying claim charts) reflect Defendant's present understanding of the purported scope of the claims as alleged by Plaintiff in its Infringement Contentions (as best those contentions can be understood in light of their present deficiencies).

The patent claims have yet to be construed. As a result, Defendant has based these contentions upon its knowledge and understanding of the potential scope of the Asserted Claims at this time, and, in part, upon the apparent interpretations of the Asserted Claims advanced by Plaintiff in its Infringement Contentions. Defendant may disagree with Plaintiff's interpretation of the meaning of many terms and phrases in the Asserted Claims. Defendant has provided these contentions based in part on its present understanding of Plaintiff's apparent constructions and interpretations of the Asserted Claims. These contentions do not represent Defendant's agreement or view as to the proper interpretation of any claim term contained therein. Any similarity between

any apparent claim interpretation in any of Defendant's charts of prior art references and Plaintiff's contentions is not an admission or agreement with Plaintiff about the meaning of any claim term, but rather a reflection of the fact that the subject matter Plaintiff believes is claimed is present in the prior art, or that the claims are otherwise invalid. These contentions are made in the alternative, and should not be interpreted to rely upon, or in any way affect, the non-infringement arguments Defendant may assert in this case. Defendant reserves the right to amend, supplement, or materially modify these contentions as the case proceeds, including to identify additional prior art and/or additional bases for invalidity under 35 U.S.C. §§ 101 and/or 112. Defendant also reserves the right to amend, supplement, or materially modify its contentions based on any infringement and/or additional claim construction positions that Plaintiff may take in this case.

Defendant also reserves the right to amend, supplement, or materially modify its contentions in response to any claim construction or interpretation positions that Plaintiff may take or based on any claim construction the Court may further adopt in this case. Defendant also reserves the right to assert that a claim is indefinite, not enabled, or fails to meet the written description requirement of 35 U.S.C. § 112 based on any claim construction or interpretation position Plaintiff may take in this case or based on any claim construction the Court may further adopt in this case.

B. No Admission

Nothing disclosed herein is an admission or acknowledgement that any product accused of infringement by Plaintiff in its Infringement Contentions (the "Accused Products"), or any of Defendant's other products or services, infringes any of the Asserted Claims.

Defendant further notes that Plaintiff appears to rely upon overly broad interpretations of the Asserted Claims. At the same time, Plaintiff's Infringement Contentions are in most places too general and vague to discern Plaintiff's infringement theories and how exactly Plaintiff

Universal Connectivity Technologies Inc.
Ex. 2002 - Page 4

contends each Accused Product meets or practices each element of the Asserted Claims. As a result, Defendant has been prejudiced in its ability to prepare these contentions. In addition, Plaintiff's Infringement Contentions, in many cases, continue to fail to put Defendant on notice of Plaintiff's interpretation of the Asserted Claims, further prejudicing Defendant's ability to identify relevant prior art.

Patent Rule 3-1(d) required Plaintiff to identify "each element of each asserted claim" that is alleged to be present under the doctrine of equivalents. In its Contentions, Plaintiff identified no specific claim elements as allegedly being present under the doctrine of equivalents. In preparing these invalidity and eligibility contentions, Defendant has relied on Plaintiff's representation that it has no doctrine of equivalents theories for any specific claim element, and any attempt by Plaintiff to present an untimely doctrine of equivalents argument or allegation would be severely prejudicial to Defendant. To the extent Plaintiff attempts to or is later permitted by the Court to amend its contentions to pursue any currently undisclosed doctrine of equivalents theories for additional claim elements, Defendant expressly reserves the right to supplement or amend these contentions to account for such amendments.

To the extent that any of the prior art references discloses the same functionality or feature of any of the Accused Products, Defendant reserves the right to argue that said feature or functionality does not practice any element of any of the Asserted Claims, and to argue, in the alternative, that if said feature or functionality is found to practice any element of any of the Asserted Claims, then the prior art reference demonstrates that the element is not novel, is obvious, and/or is otherwise not patentable.

Attached hereto are representative claim charts that identify examples of where the elements of the Asserted Claims may be found in the prior art and further identifying why those

Universal Connectivity Technologies Inc.
Ex. 2002 - Page 5

claims are not novel or non-obvious. Defendant's claim charts cite particular teachings and disclosures of the prior art as applied to features of the Asserted Claims. Providing citations to portions of the prior art references should not be construed to mean that other portions of the prior art references are not also relevant to the validity of the claims. Other passages of the identified prior art may additionally disclose, either expressly or inherently, information relevant to the validity of one or more elements of the Asserted Claims. And, persons having ordinary skill in the art generally may view an item of prior art in the context of other publications, literature, products, and understanding. As such, Defendant reserves the right to rely on uncited portions of the prior art references and on other publications and expert testimony to establish the state of the art and to aid in understanding and interpreting the cited portions, as providing context thereto, showing reasons to combine references and as additional evidence that the prior art discloses a claim limitation. Defendant further reserves the right to rely on uncited portions of the prior art references, other publications, and testimony to establish bases for combinations of certain cited references that render the asserted claims obvious. Defendant also reserves the right to rely on uncited portions of the prior art references that describe substantially the same functionality or features that are described in cited portions. Further, for any combination, Defendant reserves the right to rely on information generally known to those skilled in the art and/or common sense.

Where Defendant cites a particular drawing or figure in the accompanying charts, the citation encompasses the description of the drawing or figure, as well as any text associated with the drawing or figure (even if the associated text is not itself expressly cited). Similarly, where Defendant cites particular text concerning a drawing or figure in the accompanying charts, the citation encompasses that drawing or figure (even if the associated drawing or figure is not expressly cited).

Universal Connectivity Technologies Inc.
Ex. 2002 - Page 6

The references cited in the attached claim charts may disclose the limitations of the Asserted Claims expressly and/or inherently. Defendant may rely on inherency to demonstrate the invalidity of the Asserted Claims. Moreover, certain prior art references and solutions may inherently anticipate certain features of the asserted claims as construed by Plaintiff. Defendant may rely on cited or uncited portions of the prior art, other documents, and expert testimony to establish the inherency of certain features of the prior art to invalidate the Asserted Claims. Defendant also may rely on any reference identified in these contentions or any other reference to prove that the references are enabling, to provide a reason to combine the references, as an indication of the state of the art, or to explain the meaning of a term used in any reference.

The suggested obviousness combinations may be presented in conjunction with or in the alternative to Defendant's contentions regarding anticipation. These obviousness combinations should not be construed to suggest that any reference included in any combination is not anticipatory in its own right. Further, to the extent that Plaintiff contends that any of the references identified does not constitute prior art under 35 U.S.C. § 102, Defendant reserves the right to rely upon other prior art references in the same patent family with substantially identical disclosures as evidence of invalidity based on the same theories as those disclosed below.

Throughout the charts, Defendant provides exemplary citations indicating where the cited references disclose subject matter recited in the preambles of the claims, without regard to whether the preambles are limitations of the claims. Defendant does not take any position at this time regarding whether or not the preambles of the Asserted Claims are limitations, and will address this topic, if necessary, during the scheduled claim construction process.

With regard to the prior art that Defendant identifies as rendering the Asserted Claims obvious, non-limiting examples of the reasons why the prior art renders the Asserted Claims

Universal Connectivity Technologies Inc.
Ex. 2002 - Page 7

obvious, including exemplary identification of combinations of prior art showing obviousness, are provided below. Generally, for all prior art combinations addressed herein, a reason or motivation to combine the prior art includes that each prior art reference is relevant to data communications and/or power distribution between devices, and one of ordinary skill in the art tasked with implementing such a system would have been motivated to investigate the various existing systems, patents, and other publications, including those identified herein, to address his or her particular needs. The combinations and modifications of the prior art to invalidate the Asserted Claims would have arisen from ordinary innovation, ordinary skill, or common sense and/or would have been obvious to try or otherwise predictable in the field of data communications and/or power distribution between devices.

A person having ordinary skill in the art would have had a reason to combine the prior art based on the nature of the problem to be solved, the teachings of the prior art, and the knowledge of persons having ordinary skill in the art. For example, each of the prior art references is relevant to data communications and/or power distribution between devices, including addressing problems such as symbol alignment, device and signal negotiation, and power provision. In addressing these problems, one of ordinary skill in the art would have been motivated to combine the teachings of research in the field.

Additionally, some prior art references cite or discuss other prior art references, illustrating the close technical relationship among the prior art. To the extent that any prior art reference refers to or discusses another prior art reference, either expressly or inherently, it would have been obvious to combine those references for at least that reason.

As shown in the accompanying claim charts, numerous books and research papers show that methods were being used to streamline packetized data transmission and power distribution

across device-to-device connections. Design incentives and other market forces would have prompted combinations and modifications of the prior art. Design needs and market pressures provide ample reason to combine prior art elements in the manner combined. *See KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 401 (2007). Defendant reserves the right to supplement these discloses if such evidence of design needs and/or market pressures become apparent through discovery in this case.

Moreover, the Asserted Claims are also obvious because they merely arrange old elements, with each performing the same function that had been known, to perform and yield no more than what one having ordinary skill in the art would expect from such an arrangement.

Because there were a finite number of predictable solutions in the art of packetized data transmission and power distribution across device-to-device connections with regard to any claimed feature, it would have been obvious to a person of ordinary skill in the art to pursue the known options. A person skilled in this art would have been familiar with all the claim elements that the patentees used to distinguish the prior art during prosecution. The identified prior art references merely use those familiar elements for their primary or well-known purposes and in a manner well within the ordinary level of skill in the art. Accordingly, common sense and the knowledge of one having ordinary skill in the art render the Asserted Claims invalid.

C. Reservation of Rights

Prior art not currently included in this disclosure may become relevant. Defendant is currently unaware of the extent, if any, to which Plaintiff will contend that limitations of the Asserted Claims are not disclosed in the prior art identified by Defendant or contend that prior art identified by Defendant does not qualify as prior art. Defendant reserves the right to identify other references that would have made the addition of the allegedly missing limitation to the disclosed device or method obvious or show that the allegedly missing limitation would have been known

Universal Connectivity Technologies Inc.
Ex. 2002 - Page 9

or readily apparent to one of ordinary skill in the art at the time of the invention in light of the disclosure of the prior art at issue or other prior art. Defendant further reserves the right to rely on any of the references produced concurrently with these contentions in order to demonstrate the state of art at the alleged times of invention and as evidence of the knowledge of one of ordinary skill in the art[1] in support of any motivations to modify or combine the charted prior art references with other references or knowledge.

To the extent that Defendant charted a published patent application that later issued as a patent that includes substantially the same disclosure, Defendant contends that the issued patent discloses the claims for at least the same reasons explained in connection with the published application and Defendant reserves its rights to rely on the issued patent to establish invalidity. Similarly, to the extent that Defendant charted an issued patent the application for which was previously published (and which contains substantially the same disclosure), Defendant contends that the published application discloses the claims for at least the same reasons explained in connection with the issued patent and Defendant reserves the right to rely on the published application to establish invalidity (including to establish an earlier prior art date).

Consistent with the Patent Rules and Docket Control Order (Dkt. No. 27), and in view of the following, Defendant reserves the right to supplement or otherwise amend these Invalidity and Eligibility Contentions. This is particularly true for all claims for which Plaintiff has not provided adequate infringement contentions.

Plaintiff may also be aware of additional prior art that is not known to Defendant. To the extent Plaintiff produces additional prior art responsive to Defendant's discovery requests after

[1] For purposes of these Invalidity and Eligibility Contentions, the term "general knowledge" includes the common knowledge and common sense of a POSITA as well as the general knowledge of POSITAs in the field.

Universal Connectivity Technologies Inc.
Ex. 2002 - Page 10

these contentions are served, Defendant may supplement its Invalidity and Eligibility Contentions with prior art contained in such production once it has had a fair opportunity to review, analyze, and chart such prior art. Defendant reserves the right to amend its Invalidity and Eligibility Contentions with any additional potential prior art known by Plaintiff but not yet disclosed to Defendant.

Defendant provides these Invalidity and Eligibility Contentions only for the claims that are asserted by Plaintiff, but reserves the right to seek invalidation of all claims in each of the Asserted Patents.

In addition to the prior art that is set forth in the charts, additional prior art references are being produced concurrently with these contentions. This additional art may be relevant to the validity of the Asserted Claims depending on the positions taken by Plaintiff, and in any event may be referenced to show the state and evolution of technology in these fields. Accordingly, for the foregoing reasons, Defendant reserves the right to supplement these Invalidity and Eligibility Contentions with specific claim charts and obviousness combinations with the additional references based on the allegations and contentions asserted by Plaintiff as these allegations and corresponding constructions of the claims become clear.

D. Ongoing Investigation

Defendant's investigation is ongoing, and Defendant expressly reserves the right to amend its disclosures and document production to account for evidence uncovered as its investigation continues. Such amendments include identifying and relying on additional references that may result from Defendant's further search and analysis. Defendant reserves the right to supplement these contentions in light of any additional prior art of which Plaintiff is aware and has not disclosed to Defendant in discovery, or that might be subsequently disclosed by Plaintiff in response to Defendant's discovery requests. An example of such prior art may include products

sold and offered for sale by Silicon Image and/or Lattice Semiconductor. Such products may have been offered for sale or publicly shown in a manner that renders them prior art under 35 U.S.C. §§ 102 and 103. Defendant reserves the right to supplement these contentions to rely on those and other products from Plaintiff. Defendant's investigation into prior art systems and products is ongoing, and Defendant reserves the right to amend these contentions to reflect the identification of and information about such systems and products learned through discovery.

Defendant anticipates issuing subpoenas to third parties believed to have knowledge, documentation and/or corroborating evidence concerning some of the prior art listed herein and/or additional prior art. These third parties include, but are not limited to, the authors, employers of authors, inventors, assignees, or former or current employees of assignees or previous assignees, of the references identified in these Invalidity and Eligibility Contentions. For example, Defendant anticipates issuing subpoenas to potential prior artists including but not limited to individuals and entities responsible for the development of prior art systems identified in these Invalidity and Eligibility Contentions. Defendant reserves the right to supplement these contentions in light of any newly discovered information produced by these or other companies from which Defendant may seek discovery.

E. Incorporation by Reference

Defendant hereby incorporates by reference all arguments made in all IPR petitions filed by any party or third party (and all Institution Decisions instituting IPR relating to such petitions) challenging any Asserted Claim, including any IPR petition or decision instituting IPR that is filed after the date of these contentions. Defendant further incorporates by reference the file history of each Asserted Patent and the prior art cited therein.

Universal Connectivity Technologies Inc.
Ex. 2002 - Page 12

II. PRIORITY DATE OF THE ASSERTED PATENT CLAIMS

Plaintiff asserted the following priority dates for the Asserted Claims in its Infringement Contentions:

Asserted Patent	"Priority Date"
'905 Patent	"at least a priority date of November 22, 2000"
'307 Patent	"at least a priority date of June 12, 2003"
'798 Patent	"at least a priority date of November 22, 2000"
'265 Patent	"at least a priority date of March 31, 2011"
'712 Patent	"at least a priority date of December 11, 2008"
'520 Patent	"at least a priority date of January 4, 2008"
'231 Patent	"at least a priority date of January 4, 2008"
'103 Patent	"at least a priority date of April 14, 2014"

It is Plaintiff's burden to show entitlement to its asserted priority dates, and Defendant asserts that Plaintiff has failed to meet that burden, at least because it discloses ambiguous "priority dates" qualified with the phrase "at least as early as." To the extent Plaintiff attempts to rely on a different date from the specifically enumerated "Priority Dates" listed above (i.e., under the "at least" language), Defendant reserves the right to supplement these contentions. Moreover, because discovery has not yet been completed, Defendant has relied on Plaintiff's assertions for the purposes of identifying relevant prior art in these Invalidity and Eligibility Contentions, but Defendant does not concede that those priority dates are accurate. Defendant reserves the right to supplement these Invalidity and Eligibility Contentions should discovery reveal that Plaintiff is not entitled to the asserted priority date. In addition, Defendant also reserves the right to supplement these contentions should Plaintiff subsequently attempt to amend its infringement

contentions in any way, or to otherwise modify its infringement allegations or seek to establish an earlier date of invention.

III. THE ASSERTED CLAIMS OF THE '905 PATENT ARE INVALID

A. Identification of Prior Art

Defendant contends that the prior art references charted in Exhibits A-1 through A-6, discussed in any incorporated IPR petitions and institution decisions, and/or described below anticipate and/or render obvious, alone or in combination, one or more of the Asserted Claims of the '905 Patent.

1. The '905 Patent Claims Are Anticipated by the Prior Art

Some or all of the Asserted Claims of the '905 Patent are invalid as anticipated under 35 U.S.C. § 102 in view of each of the prior art references identified in the claim charts included in Exhibit A-1 through A-6 and/or identified in any incorporated IPR petitions and institution decisions, which identify specific examples of where each limitation of the Asserted Claims is found in the prior art references. As explained above, the cited portions of prior art references identified in the attached claim charts are exemplary in nature and representative of the content and teaching of the prior art references, and should be understood in the context of the reference as a whole and as they would be understood by a person of ordinary skill in the art ("POSITA").

In addition to references identified in any incorporated IPR petitions and institution decisions, Defendant identifies the following references as anticipating one or more of the Asserted Claims of the '905 Patent under 35 U.S.C. § 102. The table of anticipating references below is exemplary, and it does not constitute an admission that any reference not listed below does not also anticipate the claims of the '905 Patent. Further, Defendant contends that any prior art reference in the attached charts that is charted for each limitation of any given claim, anticipates that claim, regardless of whether that prior art reference is listed in the following tables.

Contentions, Defendant asserts that the Asserted Claims of the ’307 Patent are invalid for reciting at least the following claim term:

- “Multiple levels of communication protocol functionality” (claims 23 and 68)
- “Indictive of” (all asserted claims)

These claim terms, as apparently construed by Plaintiff, violate the definiteness requirements of 35 U.S.C. § 112 at least because they fail to inform, with reasonable certainty, those skilled in the art about the scope of the invention. Based on Defendant’s present understanding of Plaintiff’s Infringement Contentions, at least these claim terms are indefinite because they are inconsistent with and/or broader than the alleged invention disclosed in the specification and during prosecution, and given Plaintiff’s apparent constructions of the claims, a POSITA at the time of the invention would not understand what is claimed with reasonable certainty, even when the claims are read in light of the specification and prosecution history.

V. THE ASSERTED CLAIMS OF THE ’798 PATENT ARE INVALID

A. Identification of Prior Art

Defendant contends that the prior art references charted in Exhibits C-1 through C-8, discussed in any incorporated IPR petitions and institution decisions, and/or described below anticipate and/or render obvious, alone or in combination, one or more of the Asserted Claims of the ’798 Patent.

1. The ’798 Patent Claims Are Anticipated by the Prior Art

Some or all of the Asserted Claims of the ’798 Patent are invalid as anticipated under 35 U.S.C. § 102 in view of each of the prior art references identified in the claim charts included in Exhibit C-1 through C-8 and/or identified in any incorporated IPR petitions and institution decisions, which identify specific examples of where each limitation of the Asserted Claims is found in the prior art references. As explained above, the cited portions of prior art references

identified in the attached claim charts are exemplary in nature and representative of the content and teaching of the prior art references, and should be understood in the context of the reference as a whole and as they would be understood by a person of ordinary skill in the art.

In addition to references identified in any incorporated IPR petitions and institution decisions, Defendant identifies the following references as anticipating one or more of the Asserted Claims of the '798 Patent under 35 U.S.C. § 102. The table of anticipating references below is exemplary, and it does not constitute an admission that any reference not listed below does not also anticipate the claims of the '798 Patent. Further, Defendant contends that any prior art reference in the attached charts that is charted for each limitation of any given claim, anticipates that claim, regardless of whether that prior art reference is listed in the following tables.

a) Prior Art Patents and Applications

Number	Published/Issued Date	Short Name	Exhibit Number
U.S. Patent No. 5,592,487	Filed: Mar. 20, 1995 Issued: Jan. 7, 1997	Knecht	C-1
U.S. Patent No. 5,686,965	Filed: Sep. 18, 1995 Issued: Nov. 11, 1997	Auld	C-2
European Patent Application No. 1,137,278	Filed: Mar. 20, 2001 Pub'd: Sep. 26, 2001	Kotani	C-3
U.S. Patent No. 5,790,607	Filed: Nov. 28, 1995 Issued: Aug. 4, 1998	Burke	C-4
U.S. Patent No. 6,609,167	Filed: Mar. 15, 2000 Issued: Aug. 19, 2003	Bastiani	C-5
U.S. Patent No. 7,251,256	Filed: May 18, 2000 Issued: July 31, 2007	Barry	C-6
WIPO Patent Pub. No. 1999/023776	Filed: Oct. 31, 1997 Pub'd: May 14, 1999	Hui	C-7
U.S. Patent No. 5,917,830 U.S. Patent No. 6,430,230	Filed: Oct. 18, 1996 Issued: June 29, 1999 Filed: June 30, 1999 Issued: Aug. 6, 2002	Chen Cunningham	C-8

U.S. Patent No. 5,801,781	Filed: June 19, 1996 Issued: Sep. 1, 1998	Hiroshima	
WIPO Patent Pub. No. 2000/011831	Filed: Aug. 17, 1999 Pub'd: Mar. 2, 2000	Haartsen	
U.S. Patent No. 5,974,464	Filed: Sep. 30, 1996 Issued: Oct. 26, 1999	Shin	
U.S. Patent No. 6,498,568	Filed. Sep. 24, 1997 Issued: Dec. 24, 2002	Austin	

b) Prior Art Systems

On information and belief, systems implementing the protocols identified below were in public use, on sale, or otherwise available to the public before the priority date of the '798 Patent. Defendant's further investigation and/or subsequent discovery from Plaintiff or third parties with knowledge regarding prior art systems may reveal additional relevant prior art systems and/or further information regarding the systems identified below, or other relevant prior art systems. Defendant reserves the right to supplement these Invalidity and Eligibility Contentions, based on subsequent investigation and discovery, including from third parties.

- MPEG-2 Transport Stream
- Lightweight Medium Access Protocol ("LMAC")

Defendant additionally identifies and relies on any system, product, or public knowledge or use that embodies or otherwise incorporates any of the prior art patents and publications listed above. Defendant reserves the right to identify and rely on systems that represent different versions or are otherwise related variations of the identified systems. Defendant also incorporates by reference any and all other prior art systems identified in any other case brought by Plaintiff and/or involving the Asserted Patents.

To the extent any item of prior art cited above is deemed not to disclose, explicitly or inherently, any limitation of an asserted claim of the '798 Patent, Defendant contends that any difference between that prior art and the corresponding patent claim would have been either inherent in the art or obvious to a person of ordinary skill in the art, even if Defendant has not specifically denoted that the art is to be combined with the knowledge of a person of ordinary skill in the art.

2. The '798 Patent Claims Are Obvious Over the Prior Art

To the extent Plaintiff argues that any limitation of the asserted claims of the '798 Patent are not explicitly disclosed by an item of prior art identified above, in Exhibits C-1 through C-8, or discussed in any incorporated IPR petitions and institution decisions, any purported differences are such that the claimed subject matter as a whole would have been obvious to one skilled in the art at the time of the alleged invention, in view of the state of the art and knowledge of those skilled in the art. The item of prior art would, therefore, render the relevant claims invalid as obvious under 35 U.S.C. § 103.

Furthermore, it would have been obvious to combine any of the prior art in Exhibits C-1 through C-8 with each other, at least because all of these references relate to communication of packets between a transmitting device and a receiving device, and synchronization of such communication.

For example, U.S. Patent No. 5,592,487 ("Knecht") teaches that "[a] communications protocol is established with data packets, control packets, checksum packets, and synchronization symbols. One of the control packets is transmitted after a synchronization symbol is transmitted. One of the synchronization symbols, data packets, or checksum packets is transmitted after one of the control packets is transmitted. One of the synchronization symbols is transmitted after one of the checksum packets is transmitted, and one of the synchronization symbols or another of the data

packets is transmitted after one of the data packets is transmitted." Knecht, 2:3–13. Knecht further teaches that the "[c]ommunication protocol is established and maintained with four signals, namely: sync symbol, data packet, control packet, and checksum packet. In the preferred embodiment, control, data and checksum packets comprise 16 bits of information before encoding and 20 bits after encoding." *Id.* at 3:28–32.

Furthermore, U.S. Patent No. 5,686,965 ("Auld") teaches that "[a]n MPEG ISO system bitstream typically includes a sequence of packs 90. A typical pack 90, in turn, includes a sequence pack layer headers 92, a system header packet 94, a sequence of packets 96, 98, and 100, and an end code 102. The pack layer header 92 contains a pack start code, or sync code, used for synchronization purposes, and a system clock value. The system header packet 94 contains a variety of information commonly referred to as system housekeeping data, and a system stream ID which is used to differentiate among multiple system streams. A sequence of one or more packets typically contains either encoded audio or encoded video stream data. The final element of a pack 90 is an end code 102. It should be noted that any one packet 96, 98, 100 may carry either audio or video data, but not both simultaneously. With respect to a packet containing video data, typically a video packet header and one or more video frames is contained within the packet." Auld, 10:20–36.

Furthermore, European Patent Application No. 1,137,278 ("Kotani") teaches "the structure of a packet generated by a VCR. A sync pattern of about 2 bytes, which indicates the head of the packet, and an ID of about 3 bytes, which includes address information on data, are added to the packet." Kotani, ¶ 4. Kotani further teaches that "packets for commercialized household digital VCRs can be classified into two types, i.e., a 12-byte sub-code packet including a 2-byte sync pattern, and a 90-byte AV data packet including a 2-byte sync pattern." *Id.* ¶ 5. Kotani further

teaches that "[w]hen there are two types of time-division multiplexed HD data (for example, when there are a sub-code and AV data), it is necessary to add different sync patterns to the two types of data in order to discriminate between each type of data." *Id.* ¶ 35.

Furthermore, U.S. Patent No. 5,790,607 ("Burke") teaches that "for asynchronous data transmission and reception, it is highly desireable to organize the data into predetermined formats or packets for accurate and reliable data reception. In addition to providing the actual data, which is highly variable and essentially random, the data packet should also provide appropriate timing and synchronizing information needed for accurate data reception, without creating excessive overhead and without decreasing data throughput. In addition, for asynchronous data transmission and reception, an apparatus and method is also necessary to quickly, reliably and accurately detect such timing and synchronizing information prior to the reception of any actual data, without excessive or unnecessary processor complexity, speed, and cost." Burke, 2:38–51.

Furthermore, U.S. Patent No. 6,609,167 ("Bastiani") teaches that "a method for communicating serial data between a host computer and a peripheral device is disclosed. The packet for communication between the host computer and the peripheral device includes: (a) a synchronization field; (b) a packet type (PT) field following the synchronization field, the packet type field defining an error correction type for the packet; (c) a byte count (BC) field for defining a length of data for the packet; (d) a data type (DT) field for defining whether the data is one of link control, device control, and device data; (e) a data field; and (f) a CRC field for error checking the data in the data field." Bastiani, 2:29–39.

Furthermore, U.S. Patent No. 7,251,256 ("Barry") teaches that "[a]n LMAC-compatible link has the following characteristics: (a) the two devices at the endpoints of the link are not synchronized, e.g. the link is asynchronous; (b) the data transported on the link is in the form of

packets, cells, or frames; and (c) the MAC layer encoding of the data is such that devices called LMACs can insert synchronization symbols anywhere into the data stream at the ingress end of the link and remove them at the egress end so that the synchronization symbols are not visible to higher-layer components that process data packets." Barry, 3:61–4:3. Barry further teaches that "[i]n the LMAC protocol, the MAC layer encoding of the data is such that devices called LMACs can insert synchronization symbols anywhere into the data stream at the ingress end of an LMAC-compatible link and remove them at the egress end so that the synchronization symbols are not visible to higher-layer components that process data packets." Barry, 4:65–5:4.

Furthermore, WIPO Patent Publication No. 1999/023776 ("Hui") teaches that "[i]n data retrieval or communication systems, digital data or digitized multimedia information which can be, potentially, large or variable in size is normally packetized before storage or transmission. This process breaks the data into smaller packets where each packet contains one or more detectable synchronization patterns called sync-word, necessary information about the packet, and a payload of a smaller partition of the original data. The information about the packet may consist of a packet identification (ID), size of the payload contained in the current packet, timing information, and any necessary side information." Hui, 1:13–19.

Furthermore, U.S. Patent No. 5,917,830 ("Chen") teaches "the insertion of digital video messages such as commercials into a pre-existing compressed packetized data stream." Chen, 1:7–9. Chen further teaches that "a transport stream, or transport multiplex is made up of a contiguous set of fixed length packets. Each packet is 188 total bytes in length, with the first four of those bytes being defined as the packet header." *Id.* at 14:47–50. Chen further discloses that "[v]arious timing and identification information is provided in different portions of the transport stream. These include a packet identifier (PID) found in the transport header of each transport

packet" and "a program clock reference (PCR) which effectively represents a sample of the system time clock (STC) time base that underlies the service composed of the PIDs referenced in the service map." *Id.* at 14:57–59, 15:5–8.

Furthermore, U.S. Patent No. 6,430,230 ("Cunningham") teaches "a method of encoding payload bits for transmission over a communications link such that the running disparity of the transmitted symbol stream is constrained within a predetermined limit, and said payload bits comprising either data or control information." Cunningham, 1:58–63. Cunningham further teaches that "a data frame intended to be communicated over an optical fibre channel 20 typically comprises a binary digital message (payload) to be transferred between stations on the network; this message may comprise user data, or control information for managing operation of the network and its components." *Id.* at 2:65–3:3. Cunningham further teaches that "[c]ontrol information is conveyed by using selected n-bit data block values, together with a distinctive value of 10 for the pair of label bits a and b to indicate that the associated data block represents control information rather than actual data." *Id.* at 4:23–27.

Furthermore, U.S. Patent No. 5,801,781 ("Hiroshima") teaches "the details of the MPEG2-TS. Each packet of the MPEG2-TS has a prefix length of 188 bytes. The header 194 consists of four bytes and the payload 196 consists of 184 bytes. The header 194 is constructed by sync bytes 198, an error indicator 200, a unit start indicator 202, a transport priority 204, a packet ID 206, a scramble control 208, an adaptation field control 210, a continuity counter 212, and an adaptation field 214." Hiroshima, 10:51–58. Hiroshima further teaches "a process for forming the MPEG2-TS from the PES packet 82 by a multiplexing process." *Id.* at 11:13–14.

Furthermore, WIPO Patent Publication No. 2000/011831 ("Haartsen") teaches "a multi-part digital preamble for use in transmitting digital data packets. Advantageously, a preamble

according to the invention substantially reduces the overall length of DC-free sequences that are transmitted with each data packet in a digital FM system." Haartsen, 5:3–6. More specifically, Haartsen teaches that "a digital preamble can be strategically divided into three or more parts so that synchronization and other information can be included within the preamble while still enabling accurate DC estimation." *Id.* at 12:13–16.

Furthermore, U.S. Patent No. 5,974,464 ("Shin") teaches "a high-speed digital video signal transmission system in which high-speed video transmission is effected between a host computer system and a system display." Shin, 4:56–59. Shin further teaches that "various synchronization and other special characters are defined using the high-transition and low-transition sets of out-of-band characters. . . . The difference in the number of transitions between in-band and out-of-band characters allows selected out-of-band characters to serve as control characters, and to be readily distinguished from in-band characters within the transmitted data stream." *Id.* at 9:64–10:8.

Furthermore, U.S. Patent No. 6,498,568 ("Austin") teaches a "system [that] uses up to 5 pre-defined data packet sizes and up to 16 user defined data message types within each packet. Each packet has Forward Error Connection (FEC) and bit interleaving applied to it on transmission. Error checking is performed using a standard CRC. The system uses standard run in and byte synchronisation techniques. The byte synchronisation word used is 32 bits long and there are 5 different words used, one to identify each of the different packet types." Austin, 4:37–45.

Accordingly, because each of the identified references are directed to similar devices in the similar field of endeavor, a person of ordinary skill in the art would be motivated to combine the teachings of two or more references together because the references disclose similar systems that solve the same problems in similar ways. Thus, it would have been obvious to one of ordinary

skill in the art to combine any of the foregoing references and one of ordinary skill would have expected such a combination to be successful.

For example, to the extent that it is determined that any of the above references do not disclose a "different synchronization symbol" for each "packet type," it would further have been obvious to modify the teachings of that reference to include this feature in view of the following teachings: *See, e.g.*, Auld, 10:53–61 ("The packet data field of the packet layer 96 may contain emulations of the sync codes of other bitstream data, such as those comprising audio packets. The system synchronizer 62, however, analyzes the bitstream to determine whether a legitimate video packet sync code has been located, as at 126. If either the video sync code has not been located or a sync code emulation is located, the system synchronizer 62 will continue to predict the occurrence of a legitimate video sync code in the bitstream."); Kotani, ¶ 35 ("When there are two types of time-division multiplexed HD data (for example, when there are a sub-code and AV data), it is necessary to add different sync patterns to the two types of data in order to discriminate between each type of data."); Bastiani, 24:58–63 ("A packet type (PT) immediately follows the SYNC field of every ASP packet. A PT consists of a four bit packet type field followed by a four-bit check field as shown in FIG. 18. The PT indicates the type of packet and, by inference, the format of the packet and the type of error detection applied to the packet."); Hui, 13:6–7 ("The sync-word is defined based on applications or packet types, and is normally a digital pattern not easily found within the payload."); Chen, 14:56–15:3 ("Various timing and identification information is provided in different portions of the transport stream. These include a packet identifier (PID) found in the transport header of each transport packet to provide a reference number for identifying the transport packets carrying a specific service component. . . . Each type of packet will have a different PID identifying the packet type."); Cunningham, 4:23–27 ("Control

Universal Connectivity Technologies Inc.
Ex. 2002 - Page 24

information is conveyed by using selected n-bit data block values, together with a distinctive value of 10 for the pair of label bits a and b to indicate that the associated data block represents control information rather than actual data."); Hiroshima, 11:16–22 ("The TS header 194 is formed in a manner such that the parameters such as error indicator 200 indicative of the presence or absence of the bit error in the packet, unit start indicator 202 indicating whether a new PES packet appears in the TS payload or not, packet ID 206 indicating whether the packet in the payload is video or audio . . . ."); Haartsen, 12:19–13:2 ("Thereafter, a synchronization part of the preamble, which is not necessarily DC free, but which is protected by code to prevent detection errors, can be used not only to synchronize detection of a data packet, but also to carry timing and/or other useful information (e.g., channel identification, destination address, time of day, user identification, etc.)."); Austin, 4:42–45 ("The byte synchronisation word used is 32 _bits long and there are 5_different words used, one to identify each of the different packet types."). Therefore, a person of ordinary skill in the art would be motivated to make such a combination.

Additionally, to the extent that it is determined that any of the above references do not disclose "synchronization information," it would further have been obvious to modify the teachings of that reference to include this feature in view of the following teachings: *See, e.g.*, Burke, 3:41–48 ("In the preferred embodiment of the invention, data for transmission is organized into packets beginning with a 'preamble' 101 comprising a known, predetermined bit sequence, followed by a transition 102 comprising an indicator that data will follow, such as synchronizing information (a SYNC word preferrably having a high autocorrelation), followed by a data stream 103 having a variable length . . . ."); Hui, 13:6–9 ("The sync-word is defined based on applications or packet types, and is normally a digital pattern not easily found within the payload. After detection of the sync-word, the process checks at step 503 for the correct packet ID which may be

used to differentiate packets from different data sources, types, or versions."); Haartsen, 13:16–19 ("The synchronization word 520, which can include timing and/or other information and which is coded to protect against detection errors, is in turn followed by a trailing DC-free sequence 530."). Therefore, a person of ordinary skill in the art would be motivated to make such a combination.

Additionally, to the extent that it is determined that any of the above references do not disclose packets containing "in-band symbols" that are "transition optimized" and synchronization symbols that are "out-of-band symbols" and "not transition optimized," it would further have been obvious to modify the teachings of that reference to include this feature in view of the following teachings: *See, e.g.*, Auld, 11:14–19 ("A typical sync word may be represented by a 24-bit or 32-bit word made up of consecutive zeros and ending with a '1' bit. Alignment involves the recognition of word or byte boundaries within a system bitstream in accordance with the syntax of the selected compression algorithm."); Knecht, 4:46–49 ("The 5-bit codes, with the exception of the sync symbol, are formatted in a 40/60 percent ratio of 0 and 1 bits, with either end of the 5-bit code being made up of no more than two logical '0' or '1' bits."), 5:37–38 ("Two representations of sync symbols namely 00000 and 11111 are necessary."); Kotani, ¶ 6 ("Ten l's or ten 0's are repeated in the respective [sync] patterns. In this way, erroneous detection of data bits and sync patterns is prevented since the number of consecutive 0's is restricted in the data bits."); Shin, 9:66–10:8 ("Each character corresponding to one of these 'out-of-band' characters associated with the high-transition set includes less than four logical transitions, and each of the out-of-band characters associated with the low-transition set of code characters includes more than four logical transitions. The difference in the number of transitions between in-band and out-of-band characters allows selected out-of-band characters to serve as control characters, and to be readily

distinguished from in-band characters within the transmitted data stream."). Therefore, a person of ordinary skill in the art would be motivated to make such a combination.

Additionally, to the extent that it is determined that any of the above references do not disclose packet types that include a "control packet," it would further have been obvious to modify the teachings of that reference to include this feature in view of the following teachings: *See, e.g.*, Knecht, 3:28–32 ("Communication protocol is established and maintained with four signals, namely: sync symbol, data packet, control packet, and checksum packet. In the preferred embodiment, control, data and checksum packets comprise 16 bits of information before encoding and 20 bits after encoding."); Kotani, ¶ 5 ("For example, packets for commercialized household digital VCRs can be classified into two types, i.e., a 12-byte sub-code packet including a 2-byte sync pattern, and a 90-byte AV data packet including a 2-byte sync pattern."); Bastiani, 26:48–50 ("Data type 1 will be used for control and status of the device, and Data type 2 will be used to transfer data to the device."); Chen, 14:65–15:3 ("In the case of video, audio or isochronous control data, the stream of packets labeled with a single PID represents a single video, audio or isochronous data service elementary stream, respectively. Each type of packet will have a different PID identifying the packet type."); Cunningham, 4:23–27 ("Control information is conveyed by using selected n-bit data block values, together with a distinctive value of 10 for the pair of label bits a and b to indicate that the associated data block represents control information rather than actual data."). Therefore, a person of ordinary skill in the art would be motivated to make such a combination.

Additionally, to the extent that it is determined that any of the above references do not disclose transmitting symbols to a "switch network," it would further have been obvious to modify the teachings of that reference to include this feature in view of the following teachings: *See, e.g.*,

Burke, 1:31–36 ("[D]igital data may be transmitted both in the downstream direction, from the head end (connected to a network) to the individual user (subscriber access unit), and in the upstream direction, from the individual user to the head end (and to the network)."); Bastiani, 6:3–6 ("ASP is a star network that supports data exchange between a host computer and storage devices such as disks, tapes and CD's. The attached peripherals share ASP bandwidth through a host scheduled polling based protocol."), 6:20–23 ("In this example, the host adapter 104 is configured to include ASP circuitry 104 a for processing ASP packet data and communicating between the host computer 102 and the ASP device 106."); Barry, 12:17–23 ("The output of packet processor 54 enters the switch fabric 50. One suitable packet switch is the MMC Networks model nP5400 Packet Switch Module. In one embodiment, four such switches are connected in each switching card for faster throughput. This embodiment is used on the switching card. The capabilities of this switch are not critical to this specification."); Chen, 4:63–67 ("A compressed digital packetized data stream which includes the inserted message combined with the main program is output from the DAIM 130 to an ATM transmitter (Tx) interface 126, and then provided to an access network (e.g., cable system headend) 140."); Cunningham, 2:65–3:1 ("Referring to FIG. 1, a data frame intended to be communicated over an optical fibre channel 20 typically comprises a binary digital message (payload) to be transferred between stations on the network. . . ."); Hiroshima, 5:40–43 ("The video-on-demand system corresponds to MPEG2 having a transmission rate of 6.144 Mbps and transmits data by a prefix bit rate of the MPEG2-TS through the network 22."). Therefore, a person of ordinary skill in the art would be motivated to make such a combination.

Additionally, to the extent that it is determined that any of the above references do not disclose a "synchronization primitive" made up of "a sequence of bit-string synchronization

symbols," it would further have been obvious to modify the teachings of that reference to include this feature in view of the following teachings: *See, e.g.*, Bastiani, 24:43–46 ("A minimum of two and maximum of four SYNC characters must be sent to assure byte synchronization. It is used by the receive circuitry to align incoming data bytes and is defined to be eight bits in length."); Cunningham, 7:5–8 ("There are many potential ways to achieve code block synchronization (i.e. correct alignment in the receiver of the label and payload fields). For example, one technique would be to transmit a 'comma sequence' during start-up."); Shin, 15:20–26 ("During each preambling period, three repetitions of the same synchronization character are produced by the encoder 18. As is described herein, by processing the 21 bits most recently produced by the encoder 14, synchronization module 114 is able to detect at least one of the three repetitions of the synchronization character transmitted during a given preambling period."). Therefore, a person of ordinary skill in the art would be motivated to make such a combination.

A person skilled in the art would have been motivated to combine any of the above references. In particular, the combinations identified herein would merely have been (a) a combination of prior art elements according to known methods to yield predictable results; (b) a simple substitution of one known element for another to obtain predictable results; (c) a use of known technique to improve similar devices or methods in the same way; (d) application of a known technique to a known device or method ready for improvement to yield predictable resIs; (e) obvious to try; and/or (f) known work in one field of endeavor prompting variations of it for use in either the same field or a different one based on design incentives or other market forces since the variations are predictable to one of ordinary skill in the art. Moreover, a person of ordinary skill in the art had a reason to combine these references because—at a minimum—they

all pertain to serial communication between a transmitting device and a receiving device using packets.

As discussed above, the motivation to combine the teachings of the patents and other prior art referenced above is found in the art as specified in the charts and in: (1) the nature of the problem being solved, (2) the express, implied and inherent teachings of the prior art, (3) the knowledge of persons of ordinary skill in the art; and (4) the fact that the prior art is generally directed towards communication between a transmitting device and a receiving device using packets, and (5) the predictable results obtained in combining the different elements of the prior art. The Supreme Court has held that a showing of a motivation to combine within the references themselves is not required to prove obviousness. *See KSR*, 550 U.S. at 401 ("Any need or problem known in the field of endeavor at the time of invention may provide the motivation to combine.").

For example, a person skilled in the art would have been motivated to combine Auld with Haartsen, Kotani, Shin, and/or additional references discussed above and disclosed in Exhibits C-1 through C-8 for the reasons provided above. Moreover, the combination of references is no more than a combination of systems all directed to serial transmission of data using packets. Additionally, the combination reflects no more than a simple substitution of known elements for one another to obtain predictable results. For example, a simple substitution of Haartsen's digital preamble for Auld's sync code would achieve the predictable result of incorporating synchronization information into a synchronization symbol. As another example, a person of ordinary skill in the art would have been motivated to implement Kotani's disclosure of a unique synchronization code for subcode (i.e., control) packets into Auld, which discloses a need to distinguish between packet types. Additionally, the straightforward addition of Shin's preambling

Universal Connectivity Technologies Inc.
Ex. 2002 - Page 30

sequence to Auld would enhance the capability of Auld's system to synchronize and align the transmitted data packets.

Furthermore, a person skilled in the art would have been motivated to combine Bastiani with Haartsen, Kotani, Knecht, Shin, and/or additional references discussed above and disclosed in Exhibits C-1 through C-8 for the reasons provided above. Moreover, the combination of references is no more than a combination of systems all directed to serial transmission of data using packets. Additionally, the combination reflects no more than a simple substitution of known elements for one another to obtain predictable results. For example, a simple substitution of Haartsen's digital preamble for the non-data fields of the packets in Bastiani would achieve the predictable result of incorporating synchronization information into a synchronization symbol. As another example, a person of ordinary skill in the art would have been motivated to implement the out-of-band synchronization symbols in Kotani and Knecht in order to avoid the erroneous detection of packets and synchronization symbols. Additionally, the straightforward addition of Shin's preambling sequence to Bastiani would enhance the capability of Bastiani's system to synchronize and align the transmitted data packets.

Thus, in addition to the combinations discussed above, at least the following exemplary combinations render obvious some or all of the Asserted Claims of the '798 Patent:

- Auld in view of Haartsen
- Auld in view of Haartsen and Shin
- Auld in view of Haartsen and Kotani
- Bastiani in view of Haartsen
- Bastiani in view of Haartsen and Kotani
- Bastiani in view of Haartsen and Knecht

- Bastiani in view of Haartsen and Shin

B. Invalidity of the ’798 Patent Based on 35 U.S.C. § 112, ¶ 1 (Enablement and Written Description)

Based on Defendant’s present understanding of Plaintiff’s asserted scope of the claims of the ’798 Patent, to the extent such positions can be understood from Plaintiff’s Infringement Contentions, Defendant asserts that the Asserted Claims of the ’798 Patent are invalid for failure to provide a written description that enables the full scope of the Asserted Claims based on at least the following claim terms, phrases, or limitations:

- “the out-of-band symbols are not transition optimized.” (claims 3, 13, 21)
- “synchronization information” (claim 19)
- “bit-string synchronization symbols” (claim 28)

Based on Defendant’s present understanding of Plaintiff’s Infringement Contentions and apparent interpretation of the scope of the Asserted Claims, at least one or more of these claim terms, phrases, and limitations are not described in the specification of the Asserted Patent and do not enable a person of ordinary skill in the art to practice the claimed invention without undue experimentation or to distinguish all operative embodiments from all nonoperative embodiments without undue experimentation.

C. Invalidity of the ’798 Patent Based on 35 U.S.C. § 112, ¶ 2 (Indefiniteness)

Based on Defendant’s present understanding of Plaintiff’s asserted scope of the claims of the ’798 Patent, to the extent such positions can be understood from Plaintiff’s Infringement Contentions, the Asserted Claims of the ’798 Patent are invalid at least because they fail to inform, with reasonable certainty, those skilled in the art about the scope of the invention. At least the following claim language is indefinite:

- “the synchronization symbols are out-of-band symbols” (claims 2, 12, 20)

- "the symbols of the packets include in-band symbols" (claims 2, 12, 20)
- "the synchronization symbol permitting properly alignment upon receiving with a synchronization primitive to be correctly aligned on a symbol boundary (claim 11)
- "synchronization information" (claim 19)
- "the transmitted synchronization symbol . . . permitting an external receiving node to properly align with a synchronization primitive to be correctly aligned on a symbol boundary" (claim 19)
- "transition optimized" (claim 21)

These claim terms, phrases, limitations, and all dependent claims thereof, as apparently construed by Plaintiff, violate the definiteness requirements of 35 U.S.C. § 112 at least because they fail to inform, with reasonable certainty, those skilled in the art about the scope of the invention.

Based on Defendant's present understanding of Plaintiff's Infringement Contentions, one or more of these claim terms, phrases, and limitations are indefinite because they are inconsistent with and broader than the alleged invention disclosed in the specification and during prosecution, and given Plaintiff's apparent constructions of the claims, a person of ordinary skill in the art at the time of the invention would not understand what is claimed with reasonable certainty, even when the claims are read in light of the specification and prosecution history.

Additionally, at least claim 19 (and its dependent claims) of the '798 Patent recites functions without sufficient structure for performing the function, thus invoking 35 U.S.C. § 112, ¶ 6. Because the specification of the '798 Patent also does not recite structure for performing the function, the following terms are invalid as indefinite under 35 U.S.C. §§ 112, ¶ 2:

- "an identification component that identifies a packet type of a packet of symbols" (claim 19)
- "a transmission component that transmits a synchronization symbol that corresponds to the identified packet type . . . and that transmits the symbols of the packet" (claim 19)

The indefiniteness issues identified above apply to all claims that include the identified limitations, either explicitly or through claim dependency, and are not limited to the exemplary claims listed above.

VI. THE ASSERTED CLAIMS OF THE ’265 PATENT ARE INVALID

A. Identification of Prior Art

Defendant contends that the prior art references charted in Exhibits D-1 through D-8, discussed in any incorporated IPR petitions and institution decisions, discussed or cited in the prosecution history and/or any reexamination proceedings, and/or described below anticipate and/or render obvious, alone or in combination, one or more of the Asserted Claims of the ’265 Patent.

1. The ’265 Patent Claims Are Anticipated by the Prior Art

Some or all of the Asserted Claims of the ’265 Patent are invalid as anticipated under 35 U.S.C. § 102 in view of each of the prior art references identified in the claim charts included in Exhibit D-1 through D-8 and/or identified in any incorporated IPR petitions and institution decisions, which identify specific examples of where each limitation of the Asserted Claims is found in the prior art references. As explained above, the cited portions of prior art references identified in the attached claim charts are exemplary in nature and representative of the content and teaching of the prior art references, and should be understood in the context of the reference as a whole and as they would be understood by a person of ordinary skill in the art.

In addition to references identified in any incorporated IPR petitions and institution decisions, Defendant identifies the following references as anticipating one or more of the Asserted Claims of the ’265 Patent under 35 U.S.C. § 102. The table of anticipating references below is exemplary, and it does not constitute an admission that any reference not listed below does not also anticipate the claims of the ’265 Patent. Further, Defendant contends that any prior art

Universal Connectivity Technologies Inc.
Ex. 2002 - Page 34

Dated: July 24, 2024

Respectfully Submitted,

/s/ Stephen Marshall
Indranil Mukerji
MA Bar No. 644059
imukerji@willkie.com
Stephen Marshall
DC Bar No. 1012870
smarshall@willkie.com
Aliza George Carrano
DC Bar No. 995661
acarrano@willkie.com
Dane Sowers
TX Bar No. 24126820
dsowers@willkie.com
WILLKIE FARR & GALLAGHER LLP
1875 K Street, N.W.
Washington, DC 20006-1238
Telephone: (202) 303-1000
Facsimile: (202) 303-2000

Melissa R. Smith
Texas State Bar No. 24001351
melissa@gillamsmithlaw.com
GILLAM & SMITH, LLP
303 South Washington Avenue
Marshall, Texas 75670
Telephone: (903) 934-8450
Facsimile: (903) 934-9257

Attorneys for Defendant Lenovo Group Limited

CERTIFICATE OF SERVICE

The undersigned certifies that a true and correct copy of the above and foregoing document has been served on July 24, 2024 to all counsel of record via electronic mail.

/s/ Debbie Kakanas__

Universal Connectivity Technologies Inc.
Ex. 2002 - Page 36

Exhibit C-2

Invalidity of the ’798 Patent in View of U.S. Patent No. 5,686,965 (“Auld”)

This chart provides examples of where the prior art reference discloses, either expressly or inherently, the claim elements of asserted U.S. Patent No. 7,746,798 (“the ’798 patent”). The citations herein are exemplary and are not exhaustive; Defendant has cited the most relevant portions of the identified prior art. For each citation, Defendant hereby incorporates by reference the entire cited prior art reference, and reserves the right to rely on uncited portions of each identified prior art reference. To provide context or identify the state of the art at the time of the alleged invention, Defendant reserves the right to rely on: (1) uncited portions of the identified prior art; (2) other prior art not identified herein; (3) references that show the state of the art (irrespective of whether such references themselves qualify as prior art); (4) factual testimony from the inventors or authors of the prior art references, or purveyors of prior art devices; and (5) expert testimony.

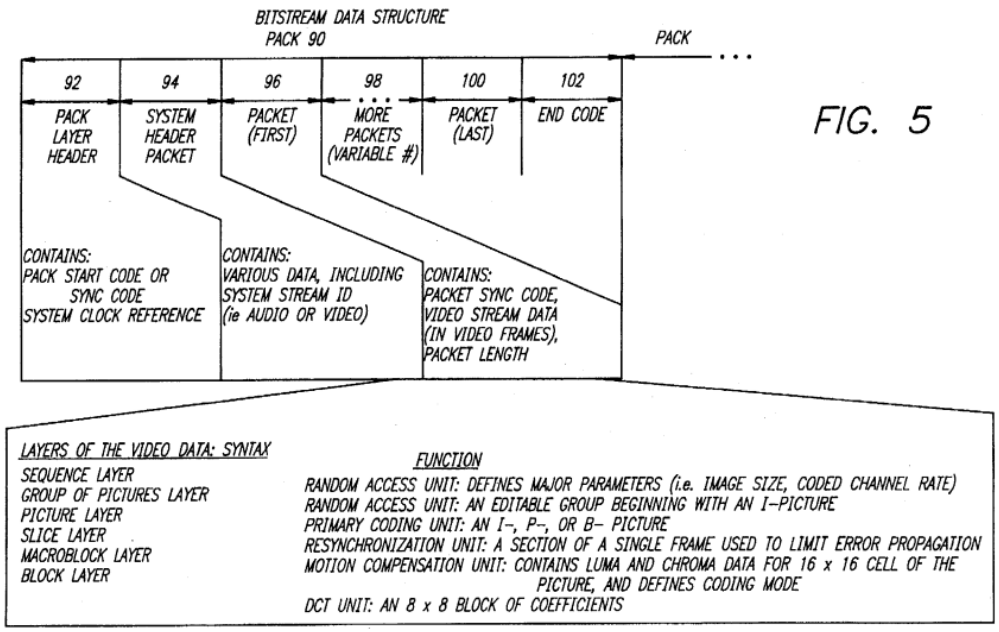
Where Defendant cites a figure, the citation includes any text associated with the figure. Where Defendant cites passages relating to a figure, the citation includes the associated figure any other text associated with that figure. Where Defendant cites particular prior art as disclosing a claim element, that citation equally applies to all identical or similar claim elements in this chart. Defendant reserves the right to further assert that a claim element is inherent, the element would have been obvious to a person of ordinary skill in the art, or the element is disclosed in other prior art that may be combined with the cited prior art. To establish the inherency of certain features of the prior art to invalidate the asserted claims, Defendant may rely on cited or uncited portions of the prior art, other documents, factual testimony, and expert testimony.

This invalidity chart is based on information presently available to Defendant, and the information herein is provided as a non-limiting example.

Auld was filed on September 18, 1995, and published on November 11, 1997; therefore, Auld is prior art under at least 35 U.S.C. § 102(b).

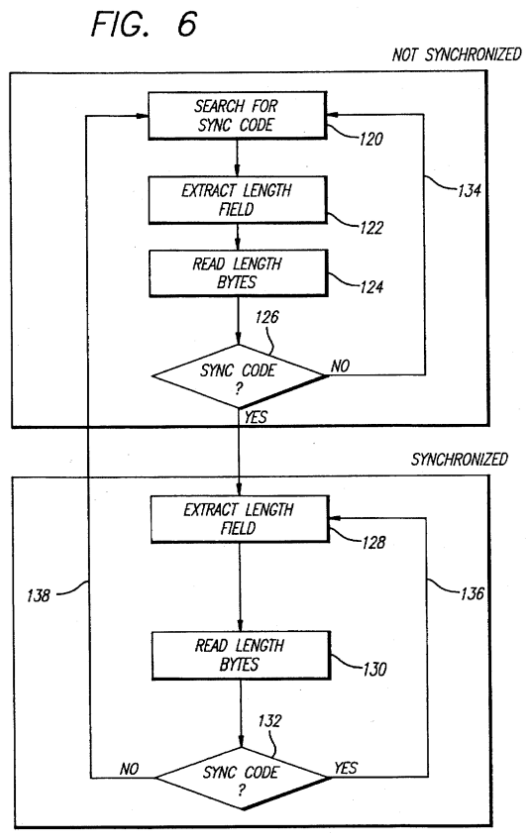
Exhibit C-2

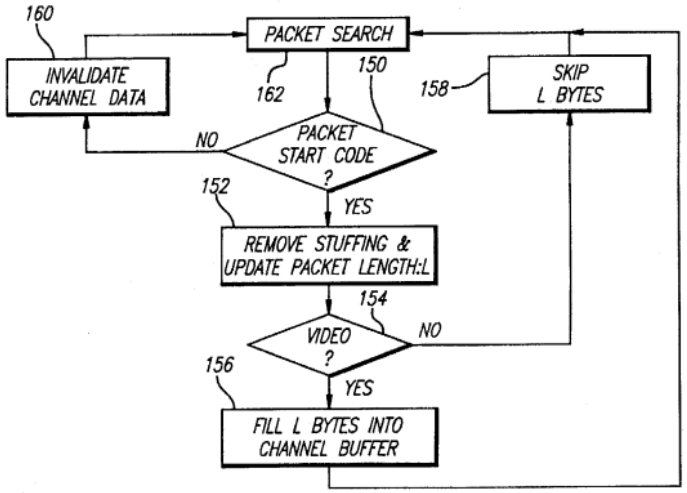
Claim Limitation		Auld
Claim 11		
11[p]	A method for identifying packet types of packets of symbols, the method comprising:	To the extent the preamble is limiting, Auld discloses this element and/or renders it obvious either alone or in combination with other references and/or the knowledge of one of ordinary skill in the art. *See, e.g.*: "The logical structure of a typical MPEG system bitstream is illustrated in FIG. 5, which represents the highest level MPEG data structure, and is often termed an ISO (International Standards Organization) system stream. An MPEG ISO system bitstream typically includes a sequence of packs 90. A typical pack 90, in turn, includes a sequence pack layer headers 92, a system header packet 94, a sequence of packets 96, 98, and 100, and an end code 102. The pack layer header 92 contains a pack start code, or sync code, used for synchronization purposes, and a system clock value. The system header packet 94 contains a variety of information commonly referred to as system housekeeping data, and a system stream ID which is used to differentiate among multiple system streams. A sequence of one or more packets typically contains either encoded audio or encoded video stream data. The final element of a pack 90 is an end code 102. It should be noted that any one packet 96, 98, 100 may carry either audio or video data, but not both simultaneously. With respect to a packet containing video data, typically a video packet header and one or more video frames is contained within the packet. Further illustrated in FIG. 5 are the hierarchical layers comprising a video packet 96. The various layers comprising the video data stream include a sequence layer, a group of pictures layer, a picture layer, a slice layer, a macroblock layer, and a block layer. The functions associated with each of these layers is also provided in FIG. 5. It should be noted that each packet or layer may contain one or more synchronization codes, referred to in the art as sync code, which are necessary to properly synchronize and align the encoded bitstream." Auld, 10:16–45.



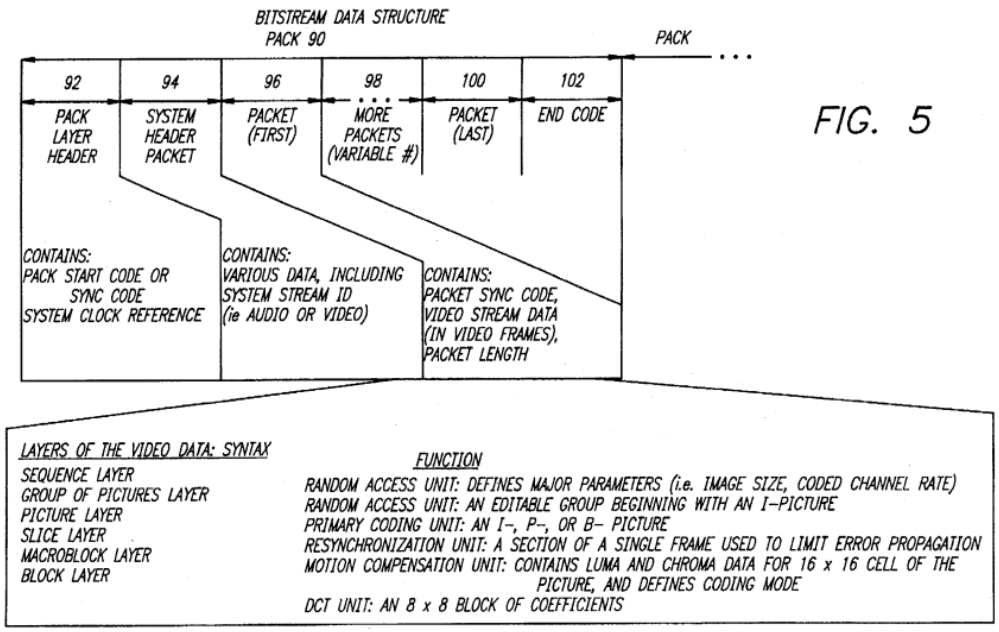
Auld, Fig. 5.

"A state diagram showing the system synchronization process is shown in FIG. 6. Assuming, for example, that the system synchronizer 62 is not synchronized to the bitstream, the system synchronizer 62 searches for a sync code, or start code, comprising the packet layer 96, as at 120. The length field of the packet layer 96 is then extracted, and the length of the packet data field is read, as at 122 and 124 respectively. The packet data field of the packet layer 96 may contain emulations of the sync codes of other bitstream data, such as those comprising audio packets. The system synchronizer 62, however, analyzes the bitstream to determine whether a legitimate video packet sync code has been located, as at 126. If either the video sync code has not been located or a sync code emulation is located, the system synchronizer 62 will continue to predict the occurrence of a legitimate video sync code in the bitstream. This prediction is based on the length field of the previously occurring header. The system synchronizer 62 then proceeds to repeat the sync code searching process, as at 134. If a sync code is located within the bitstream, as at 126 and 132, the

		length of the field in bytes is extracted and read, as at 128 and 130 respectively. When the sync code is located, as at 126 and 132, the bitstream is considered synchronized. The system synchronizer 62 will then attempt to locate the sync code of the next packet, as at 136. If no sync code can be located for the next packet 98, the system synchronizer 62 begins to search for a sync code and the bitstream transitions from a synchronized state to a non-synchronized state, as at 138." Auld, 10:46–11:6.  Auld, Fig. 6.

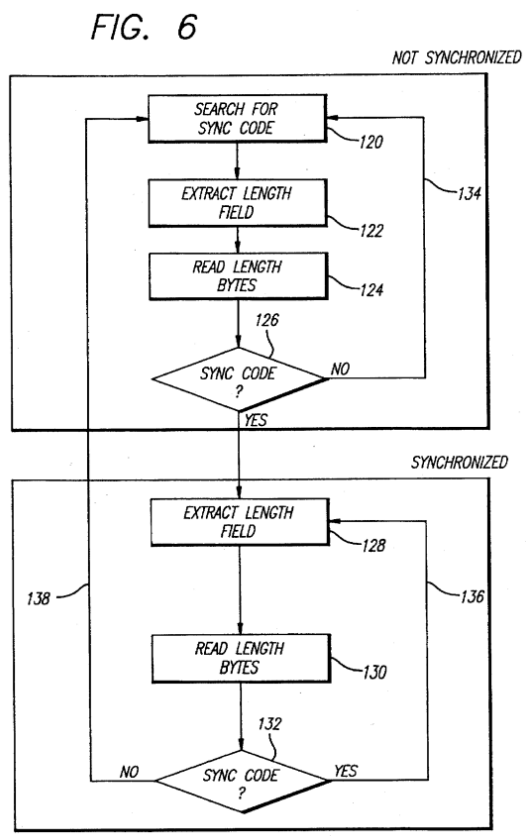
		"Turning now to FIG. 7, there is shown a flow diagram illustrating the process of extracting the video bitstream component of a multiplexed system bitstream that is performed by the system stream pre-parser 63. The system stream pre-parser 63 is intended to pass only the video bitstream component of a multiplexed system bitstream. Any non-video information components, such as the audio portion of the system bitstream, are discarded by the system stream pre-parser 63. Essentially, the packet sync code, stream ID, and packet length information at the packet layer are used to identify the type of packet being analyzed, and to skip any non-video packets. When a packet sync code has been located, as at 150, so-called stuffing data is removed, and the length of the packet data field is updated to reflect the decrease in length of the packet word resulting from the removal of the stuffing data, as at 152. Stuffing data, also referred to as stuffing bits or stuffing bytes, are typically code words that may be inserted into a compressed bitstream to increase the bit rate of the stream to that required by the digital storage or transmission medium. The stuffing data may later be discarded by the decoder during the decoding process." Auld, 11:31–52.  FIG. 7 Auld, Fig. 7.

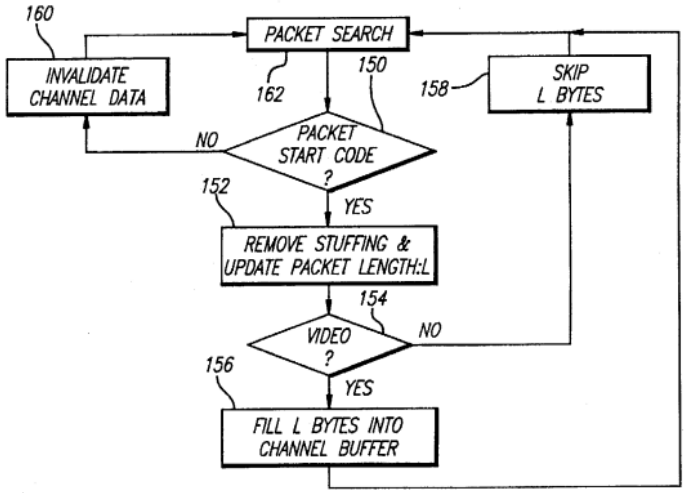
		To the extent that Plaintiff contends that Auld does not explicitly or inherently disclose this element, or render obvious this element, this element is obvious in combination with one or more references and/or with the knowledge of a person of ordinary skill in the art, as set forth in Defendant's Invalidity Contentions.
11(a)	receiving a synchronization symbol by a receiver indicating a packet type, each packet type having a different synchronization symbol;	Auld discloses and/or renders obvious this limitation. *See, e.g.*: "The logical structure of a typical MPEG system bitstream is illustrated in FIG. 5, which represents the highest level MPEG data structure, and is often termed an ISO (International Standards Organization) system stream. An MPEG ISO system bitstream typically includes a sequence of packs 90. A typical pack 90, in turn, includes a sequence pack layer headers 92, a system header packet 94, a sequence of packets 96, 98, and 100, and an end code 102. The pack layer header 92 contains a pack start code, or sync code, used for synchronization purposes, and a system clock value. The system header packet 94 contains a variety of information commonly referred to as system housekeeping data, and a system stream ID which is used to differentiate among multiple system streams. A sequence of one or more packets typically contains either encoded audio or encoded video stream data. The final element of a pack 90 is an end code 102. It should be noted that any one packet 96, 98, 100 may carry either audio or video data, but not both simultaneously. With respect to a packet containing video data, typically a video packet header and one or more video frames is contained within the packet. Further illustrated in FIG. 5 are the hierarchical layers comprising a video packet 96. The various layers comprising the video data stream include a sequence layer, a group of pictures layer, a picture layer, a slice layer, a macroblock layer, and a block layer. The functions associated with each of these layers is also provided in FIG. 5. It should be noted that each packet or layer may contain one or more synchronization codes, referred to in the art as sync code, which are necessary to properly synchronize and align the encoded bitstream." Auld, 10:16–45.



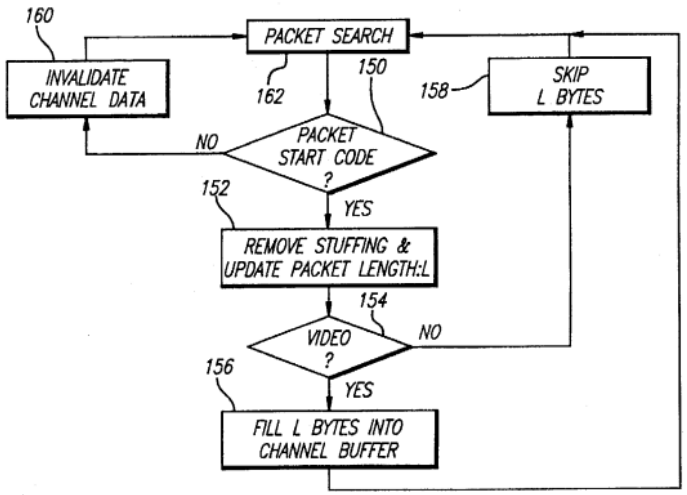
Auld, Fig. 5.

"A state diagram showing the system synchronization process is shown in FIG. 6. Assuming, for example, that the system synchronizer 62 is not synchronized to the bitstream, the system synchronizer 62 searches for a sync code, or start code, comprising the packet layer 96, as at 120. The length field of the packet layer 96 is then extracted, and the length of the packet data field is read, as at 122 and 124 respectively. The packet data field of the packet layer 96 may contain emulations of the sync codes of other bitstream data, such as those comprising audio packets. The system synchronizer 62, however, analyzes the bitstream to determine whether a legitimate video packet sync code has been located, as at 126. If either the video sync code has not been located or a sync code emulation is located, the system synchronizer 62 will continue to predict the occurrence of a legitimate video sync code in the bitstream. This prediction is based on the length field of the previously occurring header. The system synchronizer 62 then proceeds to repeat the sync code searching process, as at 134. If a sync code is located within the bitstream, as at 126 and 132, the

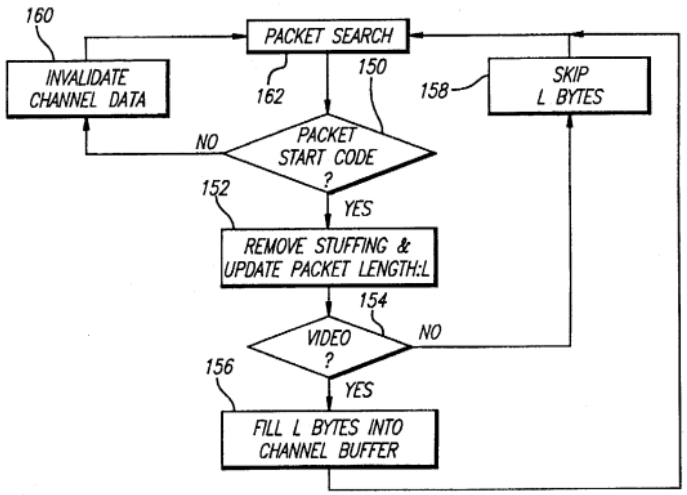
		length of the field in bytes is extracted and read, as at 128 and 130 respectively. When the sync code is located, as at 126 and 132, the bitstream is considered synchronized. The system synchronizer 62 will then attempt to locate the sync code of the next packet, as at 136. If no sync code can be located for the next packet 98, the system synchronizer 62 begins to search for a sync code and the bitstream transitions from a synchronized state to a non-synchronized state, as at 138." Auld, 10:46–11:6.  Auld, Fig. 6.

		"Turning now to FIG. 7, there is shown a flow diagram illustrating the process of extracting the video bitstream component of a multiplexed system bitstream that is performed by the system stream pre-parser 63. The system stream pre-parser 63 is intended to pass only the video bitstream component of a multiplexed system bitstream. Any non-video information components, such as the audio portion of the system bitstream, are discarded by the system stream pre-parser 63. Essentially, the packet sync code, stream ID, and packet length information at the packet layer are used to identify the type of packet being analyzed, and to skip any non-video packets. When a packet sync code has been located, as at 150, so-called stuffing data is removed, and the length of the packet data field is updated to reflect the decrease in length of the packet word resulting from the removal of the stuffing data, as at 152. Stuffing data, also referred to as stuffing bits or stuffing bytes, are typically code words that may be inserted into a compressed bitstream to increase the bit rate of the stream to that required by the digital storage or transmission medium. The stuffing data may later be discarded by the decoder during the decoding process." Auld, 11:31–52.  FIG. 7 Auld, Fig. 7.

		"Error propagation due to the emulation of non-video sync codes is generally not a concern with respect to the video bitstream input to the video stream synchronizer 65 and video stream pre-parser 66. Once the system stream pre-parser 63 parses the video bitstream from the multiplexed system bitstream, non-video sync codes generally are not present in the video bitstream. The emulation of non-video sync codes associated with conventional decoding schemes results from mis-interpreting audio and other non-video bitstream sync codes as legitimate video bitstream sync codes. Since these extraneous non-video bitstreams have, in accordance with the present invention, been stripped from the system bitstream by the system stream pre-parser 63, it is highly unlikely that such errant sync codes could remain within the video bitstream. Moreover, in the event that the video stream synchronizer 65 locates an unexpected legitimate sync code within the video bitstream, the video stream synchronizer 65 will synchronize to this unexpected sync code. Thus, synchronization of the video bitstream would simply be established earlier than expected, and would not result in the propagation of a synchronization error into the post-parser 56." Auld, 12:32–53. To the extent that Plaintiff contends that Auld does not explicitly or inherently disclose this element, or render obvious this element, this element is obvious in combination with one or more references and/or with the knowledge of a person of ordinary skill in the art, as set forth in Defendant's Invalidity Contentions.
11(b)	receiving a packet of symbols;	Auld discloses and/or renders obvious this limitation. *See, e.g.*: "Turning now to FIG. 7, there is shown a flow diagram illustrating the process of extracting the video bitstream component of a multiplexed system bitstream that is performed by the system stream pre-parser 63. The system stream pre-parser 63 is intended to pass only the video bitstream component of a multiplexed system bitstream. Any non-video information components, such as the audio portion of the system bitstream, are discarded by the system stream pre-parser 63. Essentially, the packet sync code, stream ID, and packet length information at the packet layer are used to identify the type of packet being analyzed, and to skip any non-video packets. When a packet sync code has been located, as at 150, so-called stuffing data is removed, and the length of the packet data field is updated to reflect the decrease in length of the packet word resulting from the removal

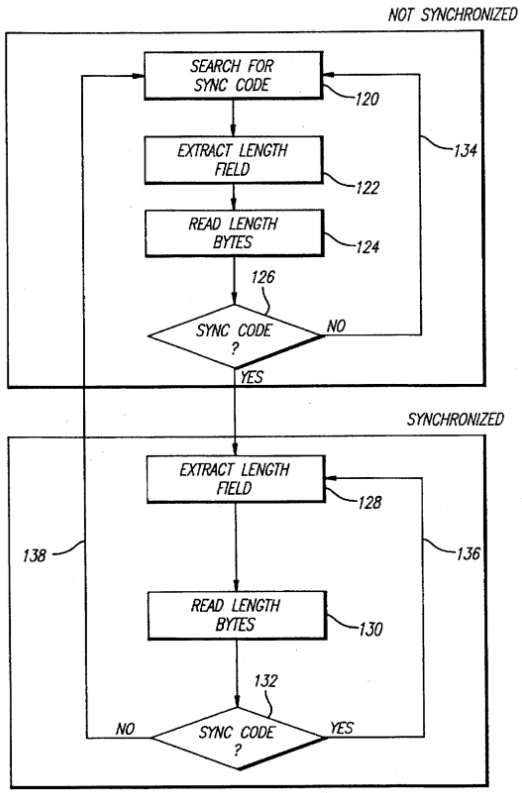
		of the stuffing data, as at 152. Stuffing data, also referred to as stuffing bits or stuffing bytes, are typically code words that may be inserted into a compressed bitstream to increase the bit rate of the stream to that required by the digital storage or transmission medium. The stuffing data may later be discarded by the decoder during the decoding process." Auld, 11:31–52.  FIG. 7 Auld, Fig. 7. "After removal of the stuffing data and updating the length of the packet data field, the stream ID of the packet is read to determine whether the bitstream is a video bitstream, as at 154. If the system ID indicates that the bitstream does indeed contain video data, the data representative of the video information is transferred to the channel buffer 54, as shown at 156. If the system ID indicates that the packet does not contain video data, the number of bytes comprising the non-video data are skipped, as at 158, and subsequent packets are searched, as at 162. If a packet sync code cannot be located, the data in the fixed rate channel 50 is considered invalid and is purged, as at 160, and subsequent packet searching continues, as at 162.

		While the system synchronizer 62 may initially synchronize incorrectly to a video bitstream, the probability of incorrect synchronization occurring is very low. Incorrect synchronization may occur as a result of data emulating video start codes, or sync codes, in the packet data field of a packet 96. In order to mis-synchronize, the system synchronizer 62 must find a second sync code after discarding the "length" bits of the packet data field, as at 158 of FIG. 7. Preferably, the "length" field is represented by a 16-bit symbol, and the sync code is represented by a 32-bit symbol. Assuming that the bits in the data field of the packet layer 96 arrive in random order, that is, "0" and "1" bits are equally probable, the mis-synchronization sequence just described has a probability of 0.548 =3.55×10-15. With a bitstream rate of 15 megabytes per second, a mis-synchronization failure would occur every 18.7×106 seconds, which is approximately 217 days. Note that any mis-synchronization failures can occur only if the system synchronizer 62 is not already synchronized to the bitstream. If such a failure is encountered, the system synchronizer 62 can subsequently reacquire synchronization within a period roughly equivalent to the time required to analyze two packets of data." Auld, 11:53–12:30. To the extent that Plaintiff contends that Auld does not explicitly or inherently disclose this element, or render obvious this element, this element is obvious in combination with one or more references and/or with the knowledge of a person of ordinary skill in the art, as set forth in Defendant's Invalidity Contentions.
11(c)	indicating that the received packet of symbols has the packet type of the received synchronization symbol; and	Auld discloses and/or renders obvious this limitation. *See, e.g.*: "It should be noted that any one packet 96, 98, 100 may carry either audio or video data, but not both simultaneously. With respect to a packet containing video data, typically a video packet header and one or more video frames is contained within the packet. Further illustrated in FIG. 5 are the hierarchical layers comprising a video packet 96. The various layers comprising the video data stream include a sequence layer, a group of pictures layer, a picture layer, a slice layer, a macroblock layer, and a block layer. The functions associated with each of these layers is also provided in FIG. 5. It should be noted that each packet or layer may contain one or more synchronization codes, referred to in the art as sync code, which are necessary to properly synchronize and align the encoded bitstream."

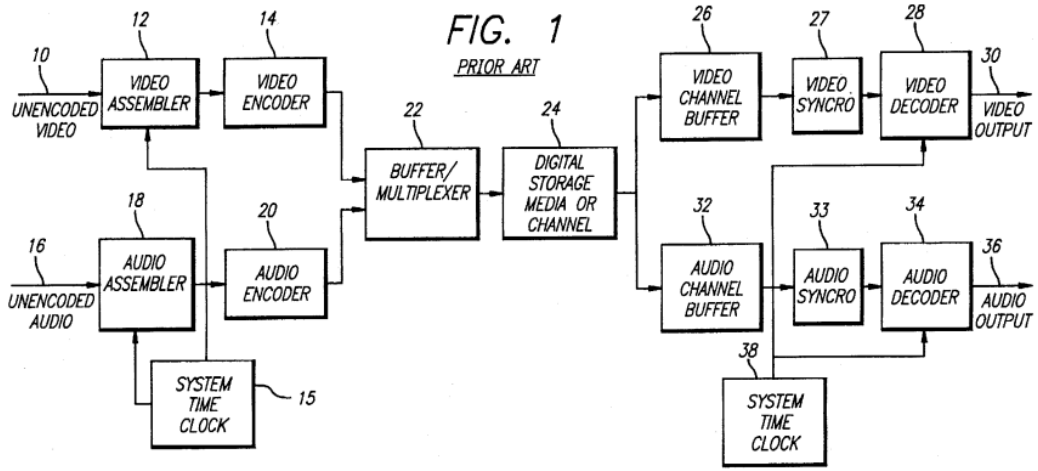
		Auld, 10:32–45. "Turning now to FIG. 7, there is shown a flow diagram illustrating the process of extracting the video bitstream component of a multiplexed system bitstream that is performed by the system stream pre-parser 63. The system stream pre-parser 63 is intended to pass only the video bitstream component of a multiplexed system bitstream. Any non-video information components, such as the audio portion of the system bitstream, are discarded by the system stream pre-parser 63. Essentially, the packet sync code, stream ID, and packet length information at the packet layer are used to identify the type of packet being analyzed, and to skip any non-video packets. When a packet sync code has been located, as at 150, so-called stuffing data is removed, and the length of the packet data field is updated to reflect the decrease in length of the packet word resulting from the removal of the stuffing data, as at 152. Stuffing data, also referred to as stuffing bits or stuffing bytes, are typically code words that may be inserted into a compressed bitstream to increase the bit rate of the stream to that required by the digital storage or transmission medium. The stuffing data may later be discarded by the decoder during the decoding process." Auld, 11:31–52.  FIG. 7

		Auld, Fig. 7. "After removal of the stuffing data and updating the length of the packet data field, the stream ID of the packet is read to determine whether the bitstream is a video bitstream, as at 154. If the system ID indicates that the bitstream does indeed contain video data, the data representative of the video information is transferred to the channel buffer 54, as shown at 156. If the system ID indicates that the packet does not contain video data, the number of bytes comprising the non-video data are skipped, as at 158, and subsequent packets are searched, as at 162. If a packet sync code cannot be located, the data in the fixed rate channel 50 is considered invalid and is purged, as at 160, and subsequent packet searching continues, as at 162." Auld, 11:53–65. "Error propagation due to the emulation of non-video sync codes is generally not a concern with respect to the video bitstream input to the video stream synchronizer 65 and video stream pre-parser 66. Once the system stream pre-parser 63 parses the video bitstream from the multiplexed system bitstream, non-video sync codes generally are not present in the video bitstream. The emulation of non-video sync codes associated with conventional decoding schemes results from mis-interpreting audio and other non-video bitstream sync codes as legitimate video bitstream sync codes. Since these extraneous non-video bitstreams have, in accordance with the present invention, been stripped from the system bitstream by the system stream pre-parser 63, it is highly unlikely that such errant sync codes could remain within the video bitstream. Moreover, in the event that the video stream synchronizer 65 locates an unexpected legitimate sync code within the video bitstream, the video stream synchronizer 65 will synchronize to this unexpected sync code. Thus, synchronization of the video bitstream would simply be established earlier than expected, and would not result in the propagation of a synchronization error into the post-parser 56." Auld, 12:32–53. To the extent that Plaintiff contends that Auld does not explicitly or inherently disclose this element, or render obvious this element, this element is obvious in combination with one or more references

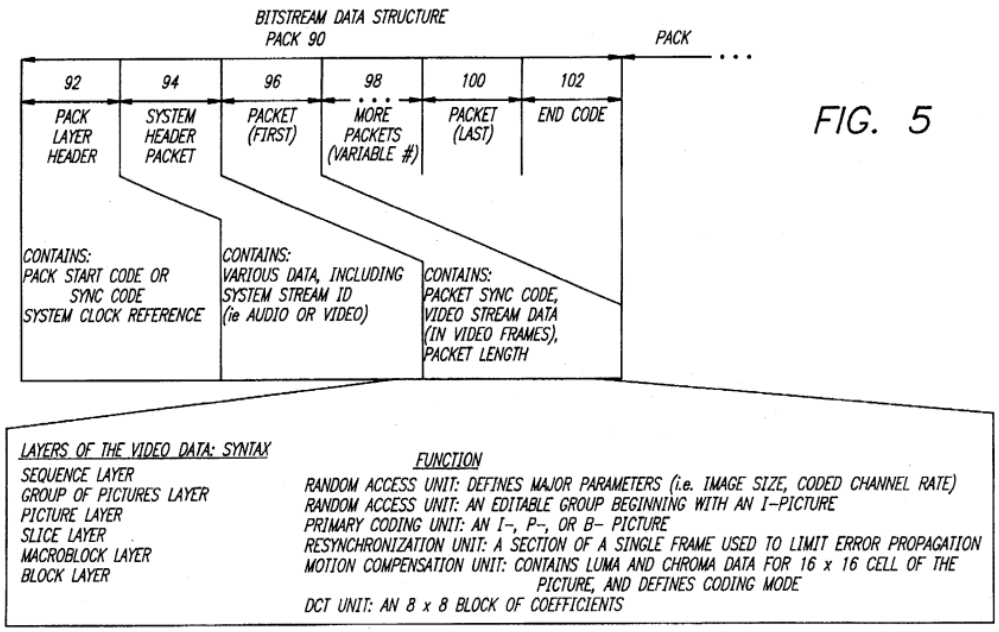
		and/or with the knowledge of a person of ordinary skill in the art, as set forth in Defendant's Invalidity Contentions.
11(d)	the synchronization symbol permitting properly alignment upon receiving with a synchronization primitive to be correctly aligned on a symbol boundary.	Auld discloses and/or renders obvious this limitation. *See, e.g.*: "It should be noted that any one packet 96, 98, 100 may carry either audio or video data, but not both simultaneously. With respect to a packet containing video data, typically a video packet header and one or more video frames is contained within the packet. Further illustrated in FIG. 5 are the hierarchical layers comprising a video packet 96. The various layers comprising the video data stream include a sequence layer, a group of pictures layer, a picture layer, a slice layer, a macroblock layer, and a block layer. The functions associated with each of these layers is also provided in FIG. 5. It should be noted that each packet or layer may contain one or more synchronization codes, referred to in the art as sync code, which are necessary to properly synchronize and align the encoded bitstream." Auld, 10:32–45. "A state diagram showing the system synchronization process is shown in FIG. 6. Assuming, for example, that the system synchronizer 62 is not synchronized to the bitstream, the system synchronizer 62 searches for a sync code, or start code, comprising the packet layer 96, as at 120. The length field of the packet layer 96 is then extracted, and the length of the packet data field is read, as at 122 and 124 respectively. The packet data field of the packet layer 96 may contain emulations of the sync codes of other bitstream data, such as those comprising audio packets. The system synchronizer 62, however, analyzes the bitstream to determine whether a legitimate video packet sync code has been located, as at 126. If either the video sync code has not been located or a sync code emulation is located, the system synchronizer 62 will continue to predict the occurrence of a legitimate video sync code in the bitstream. This prediction is based on the length field of the previously occurring header. The system synchronizer 62 then proceeds to repeat the sync code searching process, as at 134. If a sync code is located within the bitstream, as at 126 and 132, the length of the field in bytes is extracted and read, as at 128 and 130 respectively. When the sync code is located, as at 126 and 132, the bitstream is considered synchronized. The system synchronizer 62 will then attempt to locate the sync code of the next packet, as at 136. If no sync

		code can be located for the next packet 98, the system synchronizer 62 begins to search for a sync code and the bitstream transitions from a synchronized state to a non-synchronized state, as at 138." Auld, 10:46–11:6. FIG. 6  Auld, Fig. 6. "One important advantage of the present invention concerns the alignment and synchronization of a multiplexed system bitstream received from a fixed rate channel 50 by the pre-parser 52. As

		shown in FIG. 4, the system stream synchronizer 62 aligns the multiplexed system bitstream with respect to sync words embedded in the system bitstream data. Synchronization involves the alignment of data within a packet relative to the position of a sync word within the bitstream. A typical sync word may be represented by a 24-bit or 32-bit word made up of consecutive zeros and ending with a "1" bit. Alignment involves the recognition of word or byte boundaries within a system bitstream in accordance with the syntax of the selected compression algorithm. The location of sync words distributed within the system bitstream delineates the boundaries between informational word groupings within the packet. These word groupings conform to the particular grammar and syntax rules dictated by the compression algorithm employed for data encoding. Once aligned, the organization of the data structure comprising a particular informational word grouping following a sync code is thus of an established format. In a preferred embodiment, the system stream synchronizer 62 performs synchronization and alignment in a serial fashion, one bit at a time. The synchronized system bitstream is then processed by the system stream pre-parser 63." Auld, 11:7–30. To the extent that Plaintiff contends that Auld does not explicitly or inherently disclose this element, or render obvious this element, this element is obvious in combination with one or more references and/or with the knowledge of a person of ordinary skill in the art, as set forth in Defendant's Invalidity Contentions.
Claim 19		
19[p]	A communications device for transmitting packet types of packets, comprising:	To the extent the preamble is limiting, Auld discloses this element and/or renders it obvious either alone or in combination with other references and/or the knowledge of one of ordinary skill in the art. *See, e.g.*: "A typical and conventional video encoding and decoding scheme is shown in FIG. 1. An initial video signal 10 is input to a video assembler 12 wherein pictures or frames are assembled. The assembled pictures are then input into a video encoder 14 which compresses the video data in accordance with a selected compression algorithm. Similarly, an initial audio signal 16 is input into an audio assembler 18 wherein audio frames are assembled. The audio frames are then input to an audio encoder 20 for audio compression. A system clock 15 provides time stamp information which

		is incorporated into the encoded audio and video signals for synchronization purposes. The encoded video signal output from video encoder 14 and the encoded audio signal output from audio encoder 20 are then input to a buffer/multiplexer 22 to produce a multiplexed system bitstream. This multiplexed system bitstream may then be stored in a digital storage media 24 or transmitted along an appropriate transmission channel 24 for later reception, decoding, and display." Auld, 5:51–6:2.  Auld, Fig. 1. "In accordance with one aspect of the present invention, a multiplexed system bitstream, comprising at least an audio bitstream portion and a video bitstream portion, is input to a pre-parser through a fixed rate channel. A system bitstream synchronizer then initially aligns and synchronizes to the received multiplexed system bitstream. The system stream pre-parser also extracts the video bitstream portion from the multiplexed system bitstream. A video stream synchronizer then analyzes the video bitstream for the presence of a video sync code. The various layers comprising the video bitstream are subsequently synchronized and aligned by a video stream pre-parser.

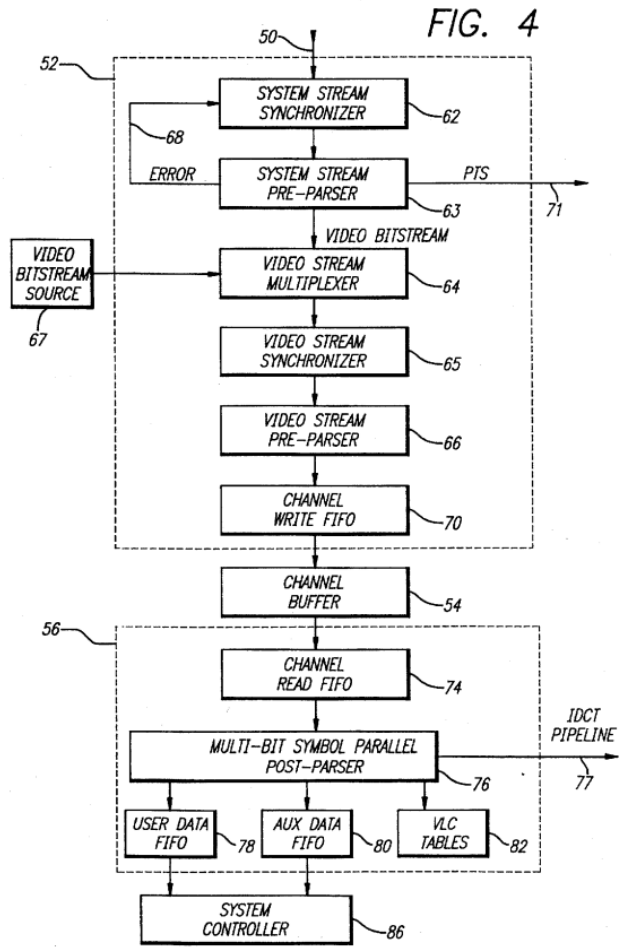
		Preferably, the parsed video bitstream is then bit-serially written to a channel write FIFO (First-in-First-Out) buffer prior to being transferred to a channel buffer." Auld, 7:35–49. "A typical pack 90, in turn, includes a sequence pack layer headers 92, a system header packet 94, a sequence of packets 96, 98, and 100, and an end code 102. The pack layer header 92 contains a pack start code, or sync code, used for synchronization purposes, and a system clock value. The system header packet 94 contains a variety of information commonly referred to as system housekeeping data, and a system stream ID which is used to differentiate among multiple system streams. A sequence of one or more packets typically contains either encoded audio or encoded video stream data. The final element of a pack 90 is an end code 102. It should be noted that any one packet 96, 98, 100 may carry either audio or video data, but not both simultaneously. With respect to a packet containing video data, typically a video packet header and one or more video frames is contained within the packet. Further illustrated in FIG. 5 are the hierarchical layers comprising a video packet 96. The various layers comprising the video data stream include a sequence layer, a group of pictures layer, a picture layer, a slice layer, a macroblock layer, and a block layer. The functions associated with each of these layers is also provided in FIG. 5. It should be noted that each packet or layer may contain one or more synchronization codes, referred to in the art as sync code, which are necessary to properly synchronize and align the encoded bitstream." Auld, 10:21–45.



Auld, Fig. 5.

"1. A Motion Picture Experts Group (MPEG) data transmission system, comprising:
a transmitter including an encoder for encoding data in accordance with an MPEG standard for producing and transmitting a multiplexed bitstream, said multiplexed bitstream including:
video bitstream data; and
non-video bitstream data;
a communications channel for carrying the multiplexed bitstream from the transmitter; and
a receiver for receiving the bitstream from the communications channel, the receiver including:
a channel buffer;
a pre-parser for parsing the multiplexed bitstream, separating the video bitstream data from the non-video bitstream data, and storing the separated video bitstream data in the channel buffer; and
a post-parser for reading the video bitstream data from the channel buffer, and parsing and translating the video bitstream data to produce video symbols."

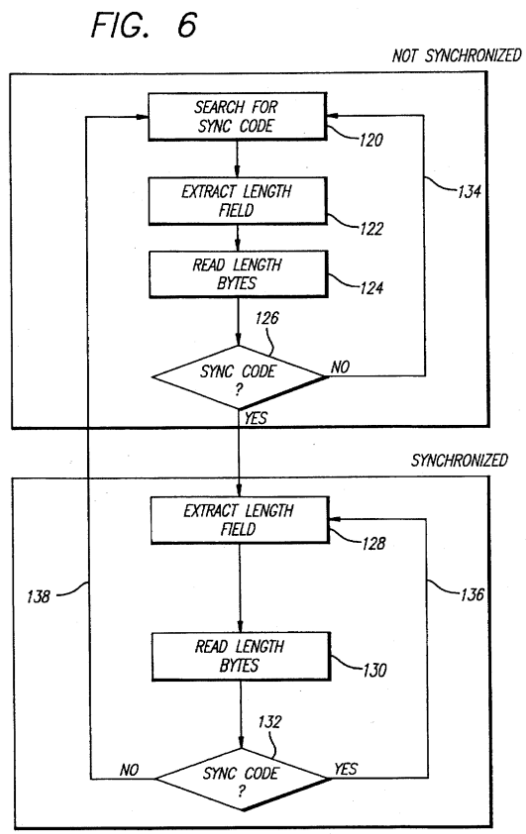
		Auld, claim 1. To the extent that Plaintiff contends that Auld does not explicitly or inherently disclose this element, or render obvious this element, this element is obvious in combination with one or more references and/or with the knowledge of a person of ordinary skill in the art, as set forth in Defendant's Invalidity Contentions.
19(a)	an identification component that identifies a packet type of a packet of symbols; and	Auld discloses and/or renders obvious this limitation. *See, e.g.*:



Auld, Fig. 4.

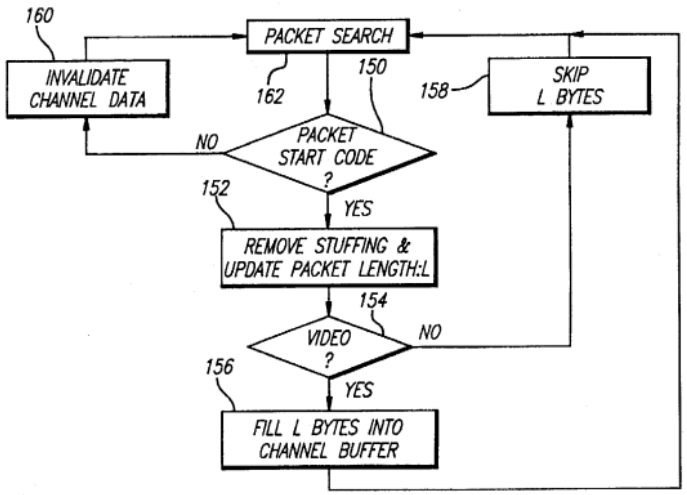
"A state diagram showing the system synchronization process is shown in FIG. 6. Assuming, for example, that the system synchronizer 62 is not synchronized to the bitstream, the system synchronizer 62 searches for a sync code, or start code, comprising the packet layer 96, as at 120.

		The length field of the packet layer 96 is then extracted, and the length of the packet data field is read, as at 122 and 124 respectively. The packet data field of the packet layer 96 may contain emulations of the sync codes of other bitstream data, such as those comprising audio packets. The system synchronizer 62, however, analyzes the bitstream to determine whether a legitimate video packet sync code has been located, as at 126. If either the video sync code has not been located or a sync code emulation is located, the system synchronizer 62 will continue to predict the occurrence of a legitimate video sync code in the bitstream. This prediction is based on the length field of the previously occurring header. The system synchronizer 62 then proceeds to repeat the sync code searching process, as at 134. If a sync code is located within the bitstream, as at 126 and 132, the length of the field in bytes is extracted and read, as at 128 and 130 respectively. When the sync code is located, as at 126 and 132, the bitstream is considered synchronized. The system synchronizer 62 will then attempt to locate the sync code of the next packet, as at 136. If no sync code can be located for the next packet 98, the system synchronizer 62 begins to search for a sync code and the bitstream transitions from a synchronized state to a non-synchronized state, as at 138." Auld, 10:46–11:6.

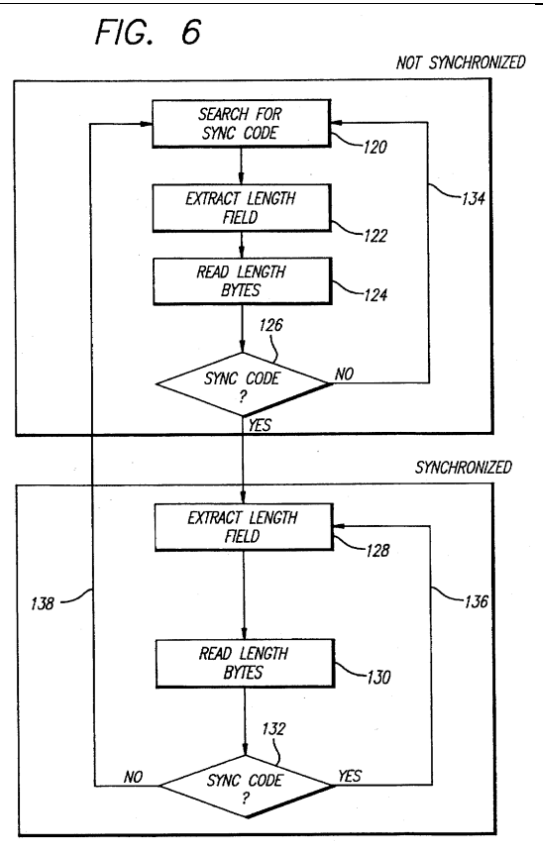


Auld, Fig. 6.

"Turning now to FIG. 7, there is shown a flow diagram illustrating the process of extracting the video bitstream component of a multiplexed system bitstream that is performed by the system stream pre-parser 63. The system stream pre-parser 63 is intended to pass only the video bitstream component of a multiplexed system bitstream. Any non-video information components, such as the audio portion of the system bitstream, are discarded by the system stream pre-parser 63. Essentially, the packet sync code, stream ID, and packet length information at the packet layer are used to identify the type of packet being analyzed, and to skip any non-video packets. When a packet sync

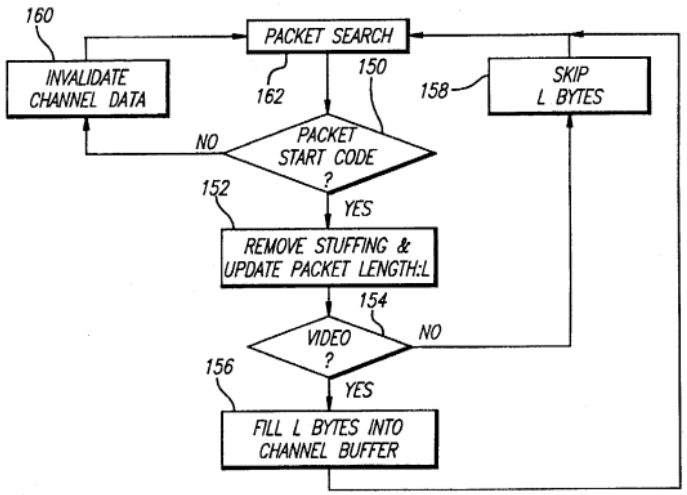
		code has been located, as at 150, so-called stuffing data is removed, and the length of the packet data field is updated to reflect the decrease in length of the packet word resulting from the removal of the stuffing data, as at 152. Stuffing data, also referred to as stuffing bits or stuffing bytes, are typically code words that may be inserted into a compressed bitstream to increase the bit rate of the stream to that required by the digital storage or transmission medium. The stuffing data may later be discarded by the decoder during the decoding process." Auld, 11:31–52.  FIG. 7 Auld, Fig. 7. To the extent that Plaintiff contends that Auld does not explicitly or inherently disclose this element, or render obvious this element, this element is obvious in combination with one or more references and/or with the knowledge of a person of ordinary skill in the art, as set forth in Defendant's Invalidity Contentions.
19(b)	a transmission component that transmits a synchronization symbol	Auld discloses and/or renders obvious this limitation. *See, e.g.*:

	that corresponds to the identified packet type, the transmitted synchronization symbol providing synchronization information and each packet type having a different synchronization symbol and permitting an external receiving node to properly align with a synchronization primitive to be correctly aligned on a symbol boundary, and that transmits the symbols of the packet.	"A state diagram showing the system synchronization process is shown in FIG. 6. Assuming, for example, that the system synchronizer 62 is not synchronized to the bitstream, the system synchronizer 62 searches for a sync code, or start code, comprising the packet layer 96, as at 120. The length field of the packet layer 96 is then extracted, and the length of the packet data field is read, as at 122 and 124 respectively. The packet data field of the packet layer 96 may contain emulations of the sync codes of other bitstream data, such as those comprising audio packets. The system synchronizer 62, however, analyzes the bitstream to determine whether a legitimate video packet sync code has been located, as at 126. If either the video sync code has not been located or a sync code emulation is located, the system synchronizer 62 will continue to predict the occurrence of a legitimate video sync code in the bitstream. This prediction is based on the length field of the previously occurring header. The system synchronizer 62 then proceeds to repeat the sync code searching process, as at 134. If a sync code is located within the bitstream, as at 126 and 132, the length of the field in bytes is extracted and read, as at 128 and 130 respectively. When the sync code is located, as at 126 and 132, the bitstream is considered synchronized. The system synchronizer 62 will then attempt to locate the sync code of the next packet, as at 136. If no sync code can be located for the next packet 98, the system synchronizer 62 begins to search for a sync code and the bitstream transitions from a synchronized state to a non-synchronized state, as at 138." Auld, 10:46–11:6.



Auld, Fig. 6.

"Turning now to FIG. 7, there is shown a flow diagram illustrating the process of extracting the video bitstream component of a multiplexed system bitstream that is performed by the system stream pre-parser 63. The system stream pre-parser 63 is intended to pass only the video bitstream component of a multiplexed system bitstream. Any non-video information components, such as the audio portion of the system bitstream, are discarded by the system stream pre-parser 63. Essentially, the packet sync code, stream ID, and packet length information at the packet layer are used to identify the type of packet being analyzed, and to skip any non-video packets. When a packet sync

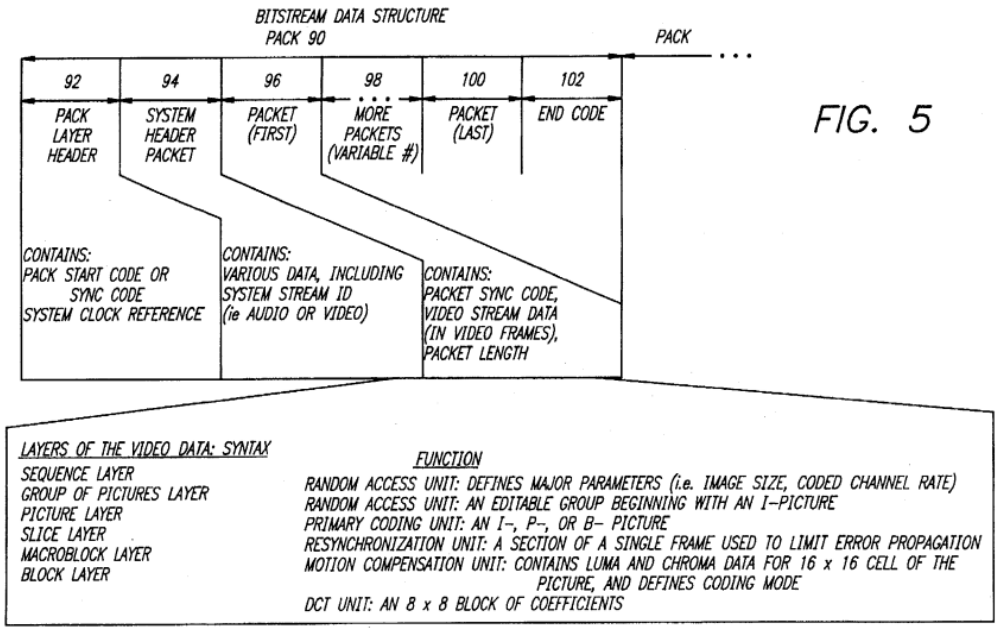
		code has been located, as at 150, so-called stuffing data is removed, and the length of the packet data field is updated to reflect the decrease in length of the packet word resulting from the removal of the stuffing data, as at 152. Stuffing data, also referred to as stuffing bits or stuffing bytes, are typically code words that may be inserted into a compressed bitstream to increase the bit rate of the stream to that required by the digital storage or transmission medium. The stuffing data may later be discarded by the decoder during the decoding process." Auld, 11:31–52.  FIG. 7 Auld, Fig. 7. "Error propagation due to the emulation of non-video sync codes is generally not a concern with respect to the video bitstream input to the video stream synchronizer 65 and video stream pre-parser 66. Once the system stream pre-parser 63 parses the video bitstream from the multiplexed system bitstream, non-video sync codes generally are not present in the video bitstream. The emulation of non-video sync codes associated with conventional decoding schemes results from mis-interpreting audio and other non-video bitstream sync codes as legitimate video bitstream sync codes. Since these extraneous non-video bitstreams have, in accordance with the present invention, been stripped

		from the system bitstream by the system stream pre-parser 63, it is highly unlikely that such errant sync codes could remain within the video bitstream. Moreover, in the event that the video stream synchronizer 65 locates an unexpected legitimate sync code within the video bitstream, the video stream synchronizer 65 will synchronize to this unexpected sync code. Thus, synchronization of the video bitstream would simply be established earlier than expected, and would not result in the propagation of a synchronization error into the post-parser 56." Auld, 12:32–53. "It should be noted that any one packet 96, 98, 100 may carry either audio or video data, but not both simultaneously. With respect to a packet containing video data, typically a video packet header and one or more video frames is contained within the packet. Further illustrated in FIG. 5 are the hierarchical layers comprising a video packet 96. The various layers comprising the video data stream include a sequence layer, a group of pictures layer, a picture layer, a slice layer, a macroblock layer, and a block layer. The functions associated with each of these layers is also provided in FIG. 5. It should be noted that each packet or layer may contain one or more synchronization codes, referred to in the art as sync code, which are necessary to properly synchronize and align the encoded bitstream." Auld, 10:32–45. "One important advantage of the present invention concerns the alignment and synchronization of a multiplexed system bitstream received from a fixed rate channel 50 by the pre-parser 52. As shown in FIG. 4, the system stream synchronizer 62 aligns the multiplexed system bitstream with respect to sync words embedded in the system bitstream data. Synchronization involves the alignment of data within a packet relative to the position of a sync word within the bitstream. A typical sync word may be represented by a 24-bit or 32-bit word made up of consecutive zeros and ending with a "1" bit. Alignment involves the recognition of word or byte boundaries within a system bitstream in accordance with the syntax of the selected compression algorithm. The location of sync words distributed within the system bitstream delineates the boundaries between informational word groupings within the packet. These word groupings conform to the particular grammar and syntax rules dictated by the compression algorithm employed for data encoding. Once aligned, the organization of the data structure comprising a particular informational word grouping

		following a sync code is thus of an established format. In a preferred embodiment, the system stream synchronizer 62 performs synchronization and alignment in a serial fashion, one bit at a time. The synchronized system bitstream is then processed by the system stream pre-parser 63." Auld, 11:7–30. To the extent that Plaintiff contends that Auld does not explicitly or inherently disclose this element, or render obvious this element, this element is obvious in combination with one or more references and/or with the knowledge of a person of ordinary skill in the art, as set forth in Defendant's Invalidity Contentions.
Claim 20		
20[p]	The communications device of claim 19	Auld discloses, expressly or inherently, the transmitting device of claim 19 under any proposed construction. *See* exemplary disclosures and corresponding discussion for claim 19.
20	wherein the symbols of the packet include in-band symbols and the synchronization symbols are out-of-band symbols.	Auld discloses and/or renders obvious this limitation. *See, e.g.*: "The logical structure of a typical MPEG system bitstream is illustrated in FIG. 5, which represents the highest level MPEG data structure, and is often termed an ISO (International Standards Organization) system stream. An MPEG ISO system bitstream typically includes a sequence of packs 90. A typical pack 90, in turn, includes a sequence pack layer headers 92, a system header packet 94, a sequence of packets 96, 98, and 100, and an end code 102. The pack layer header 92 contains a pack start code, or sync code, used for synchronization purposes, and a system clock value. The system header packet 94 contains a variety of information commonly referred to as system housekeeping data, and a system stream ID which is used to differentiate among multiple system streams. A sequence of one or more packets typically contains either encoded audio or encoded video stream data. The final element of a pack 90 is an end code 102. It should be noted that any one packet 96, 98, 100 may carry either audio or video data, but not both simultaneously. With respect to a packet containing video data, typically a video packet header and one or more video frames is contained within the packet. Further illustrated in FIG. 5 are the hierarchical layers comprising a video packet 96. The various layers comprising the video data stream include a sequence layer, a group of pictures layer, a picture layer, a slice layer, a macroblock layer, and a block layer. The functions associated with each of these layers is also provided in FIG. 5. It should

be noted that each packet or layer may contain one or more synchronization codes, referred to in the art as sync code, which are necessary to properly synchronize and align the encoded bitstream."

Auld, 10:16–45.

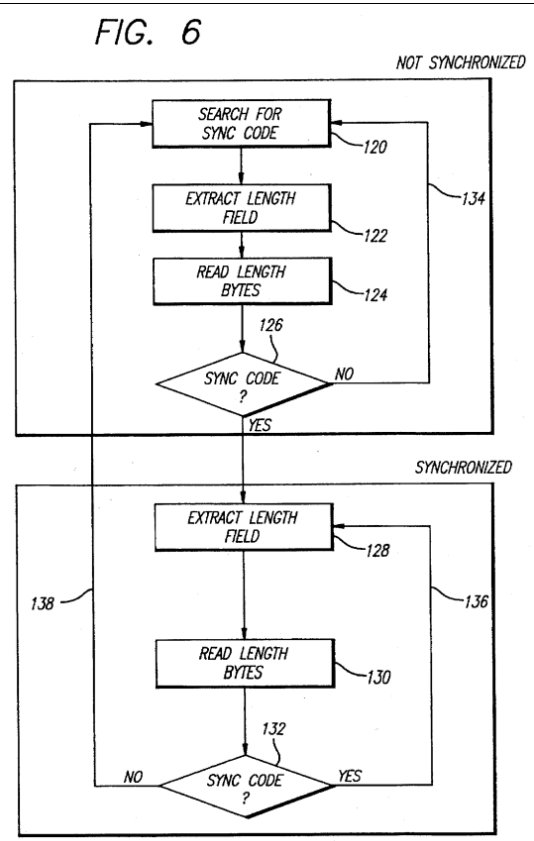


Auld, Fig. 5.

"The packet data field of the packet layer 96 may contain emulations of the sync codes of other bitstream data, such as those comprising audio packets. The system synchronizer 62, however, analyzes the bitstream to determine whether a legitimate video packet sync code has been located, as at 126. If either the video sync code has not been located or a sync code emulation is located, the system synchronizer 62 will continue to predict the occurrence of a legitimate video sync code in the bitstream."

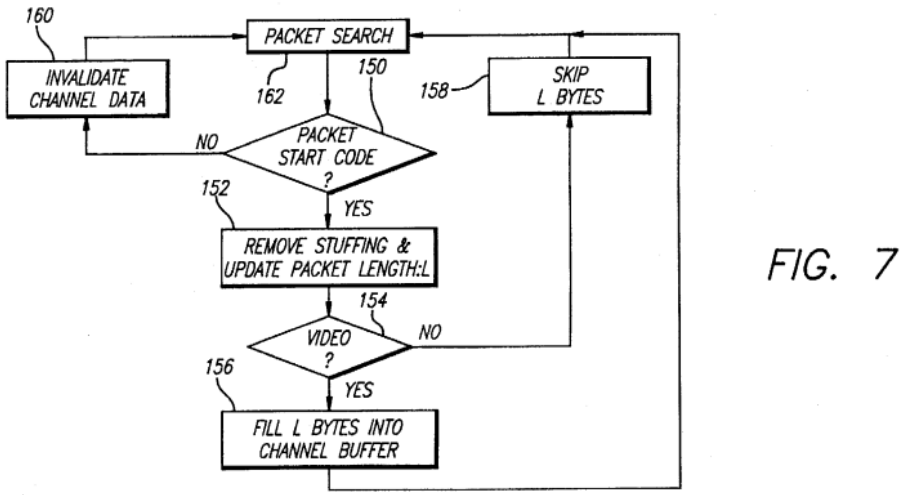
		Auld, 10:53–61. "A typical sync word may be represented by a 24-bit or 32-bit word made up of consecutive zeros and ending with a "1" bit. Alignment involves the recognition of word or byte boundaries within a system bitstream in accordance with the syntax of the selected compression algorithm. The location of sync words distributed within the system bitstream delineates the boundaries between informational word groupings within the packet. These word groupings conform to the particular grammar and syntax rules dictated by the compression algorithm employed for data encoding. Once aligned, the organization of the data structure comprising a particular informational word grouping following a sync code is thus of an established format." Auld, 11:14–26. To the extent that Plaintiff contends that Auld does not explicitly or inherently disclose this element, or render obvious this element, this element is obvious in combination with one or more references and/or with the knowledge of a person of ordinary skill in the art, as set forth in Defendant's Invalidity Contentions.
Claim 21		
21[p]	The communications device of claim 20	Auld discloses, expressly or inherently, the transmitting device of claim 20 under any proposed construction. *See* exemplary disclosures and corresponding discussion for claim 20.
21	wherein the in-band symbols are transition optimized and the out-of-band synchronization symbols are not transition optimized.	One of ordinary skill would find this limitation disclosed either expressly or inherently in the teachings of the reference and its incorporated disclosures taken as a whole, or in combination with the state of the art at the time of the alleged invention, as evidenced, for example, by substantial other references identified in Defendant's invalidity contentions and accompanying charts. Rather than repeat those disclosures here, they are incorporated by reference into this chart.
Claim 22		

22[p]	The communications device of claim 19	Auld discloses, expressly or inherently, the transmitting device of claim 19 under any proposed construction. *See* exemplary disclosures and corresponding discussion for claim 19.
22	wherein the synchronization symbol is transmitted before transmitting the symbols of the packet.	Auld discloses and/or renders obvious this limitation. *See, e.g.*: "A state diagram showing the system synchronization process is shown in FIG. 6. Assuming, for example, that the system synchronizer 62 is not synchronized to the bitstream, the system synchronizer 62 searches for a sync code, or start code, comprising the packet layer 96, as at 120. The length field of the packet layer 96 is then extracted, and the length of the packet data field is read, as at 122 and 124 respectively. The packet data field of the packet layer 96 may contain emulations of the sync codes of other bitstream data, such as those comprising audio packets. The system synchronizer 62, however, analyzes the bitstream to determine whether a legitimate video packet sync code has been located, as at 126. If either the video sync code has not been located or a sync code emulation is located, the system synchronizer 62 will continue to predict the occurrence of a legitimate video sync code in the bitstream. This prediction is based on the length field of the previously occurring header. The system synchronizer 62 then proceeds to repeat the sync code searching process, as at 134. If a sync code is located within the bitstream, as at 126 and 132, the length of the field in bytes is extracted and read, as at 128 and 130 respectively. When the sync code is located, as at 126 and 132, the bitstream is considered synchronized. The system synchronizer 62 will then attempt to locate the sync code of the next packet, as at 136. If no sync code can be located for the next packet 98, the system synchronizer 62 begins to search for a sync code and the bitstream transitions from a synchronized state to a non-synchronized state, as at 138." Auld, 10:46–11:6.

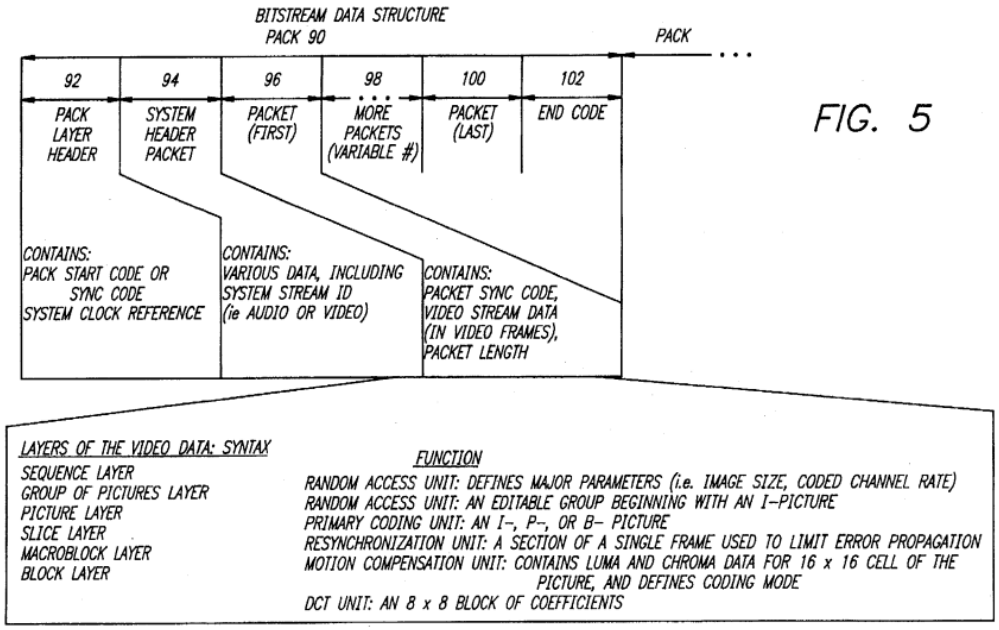


Auld, Fig. 6.

"Turning now to FIG. 7, there is shown a flow diagram illustrating the process of extracting the video bitstream component of a multiplexed system bitstream that is performed by the system stream pre-parser 63. The system stream pre-parser 63 is intended to pass only the video bitstream component of a multiplexed system bitstream. Any non-video information components, such as the audio portion of the system bitstream, are discarded by the system stream pre-parser 63. Essentially, the packet sync code, stream ID, and packet length information at the packet layer are used to identify the type of packet being analyzed, and to skip any non-video packets. When a packet sync

		code has been located, as at 150, so-called stuffing data is removed, and the length of the packet data field is updated to reflect the decrease in length of the packet word resulting from the removal of the stuffing data, as at 152. Stuffing data, also referred to as stuffing bits or stuffing bytes, are typically code words that may be inserted into a compressed bitstream to increase the bit rate of the stream to that required by the digital storage or transmission medium. The stuffing data may later be discarded by the decoder during the decoding process." Auld, 11:31–52.  FIG. 7 Auld, Fig. 7. "After removal of the stuffing data and updating the length of the packet data field, the stream ID of the packet is read to determine whether the bitstream is a video bitstream, as at 154. If the system ID indicates that the bitstream does indeed contain video data, the data representative of the video information is transferred to the channel buffer 54, as shown at 156. If the system ID indicates that the packet does not contain video data, the number of bytes comprising the non-video data are skipped, as at 158, and subsequent packets are searched, as at 162. If a packet sync code cannot be

		located, the data in the fixed rate channel 50 is considered invalid and is purged, as at 160, and subsequent packet searching continues, as at 162." Auld, 11:53–65. To the extent that Plaintiff contends that Auld does not explicitly or inherently disclose this element, or render obvious this element, this element is obvious in combination with one or more references and/or with the knowledge of a person of ordinary skill in the art, as set forth in Defendant's Invalidity Contentions.
Claim 23		
23[p]	The communications device of claim 19	Auld discloses, expressly or inherently, the transmitting device of claim 19 under any proposed construction. *See* exemplary disclosures and corresponding discussion for claim 19.
23	wherein the packet has a header with a field that indicates packet type and the identification component checks the field of the header that indicates packet type.	Auld discloses and/or renders obvious this limitation. *See, e.g.*: "The logical structure of a typical MPEG system bitstream is illustrated in FIG. 5, which represents the highest level MPEG data structure, and is often termed an ISO (International Standards Organization) system stream. An MPEG ISO system bitstream typically includes a sequence of packs 90. A typical pack 90, in turn, includes a sequence pack layer headers 92, a system header packet 94, a sequence of packets 96, 98, and 100, and an end code 102. The pack layer header 92 contains a pack start code, or sync code, used for synchronization purposes, and a system clock value. The system header packet 94 contains a variety of information commonly referred to as system housekeeping data, and a system stream ID which is used to differentiate among multiple system streams. A sequence of one or more packets typically contains either encoded audio or encoded video stream data. The final element of a pack 90 is an end code 102. It should be noted that any one packet 96, 98, 100 may carry either audio or video data, but not both simultaneously. With respect to a packet containing video data, typically a video packet header and one or more video frames is contained within the packet. Further illustrated in FIG. 5 are the hierarchical layers comprising a video packet 96. The various layers comprising the video data stream include a sequence layer, a group of pictures layer, a picture layer, a slice layer, a macroblock layer, and a

		block layer. The functions associated with each of these layers is also provided in FIG. 5. It should be noted that each packet or layer may contain one or more synchronization codes, referred to in the art as sync code, which are necessary to properly synchronize and align the encoded bitstream." Auld, 10:16–45.  Auld, Fig. 5. "Turning now to FIG. 7, there is shown a flow diagram illustrating the process of extracting the video bitstream component of a multiplexed system bitstream that is performed by the system stream pre-parser 63. The system stream pre-parser 63 is intended to pass only the video bitstream component of a multiplexed system bitstream. Any non-video information components, such as the audio portion of the system bitstream, are discarded by the system stream pre-parser 63. Essentially, the packet sync code, stream ID, and packet length information at the packet layer are used to

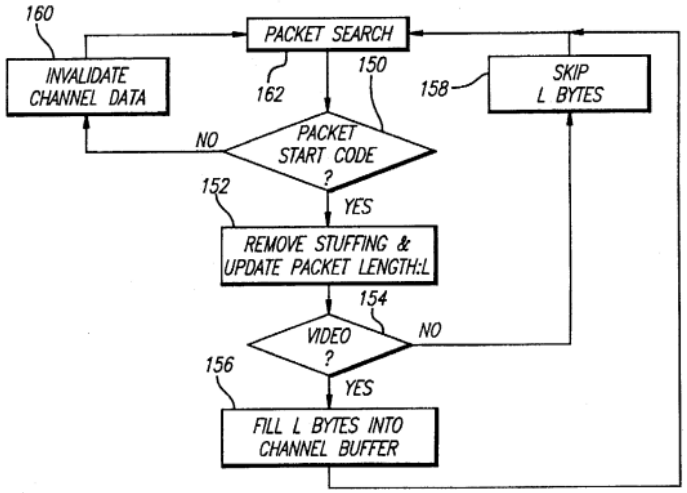
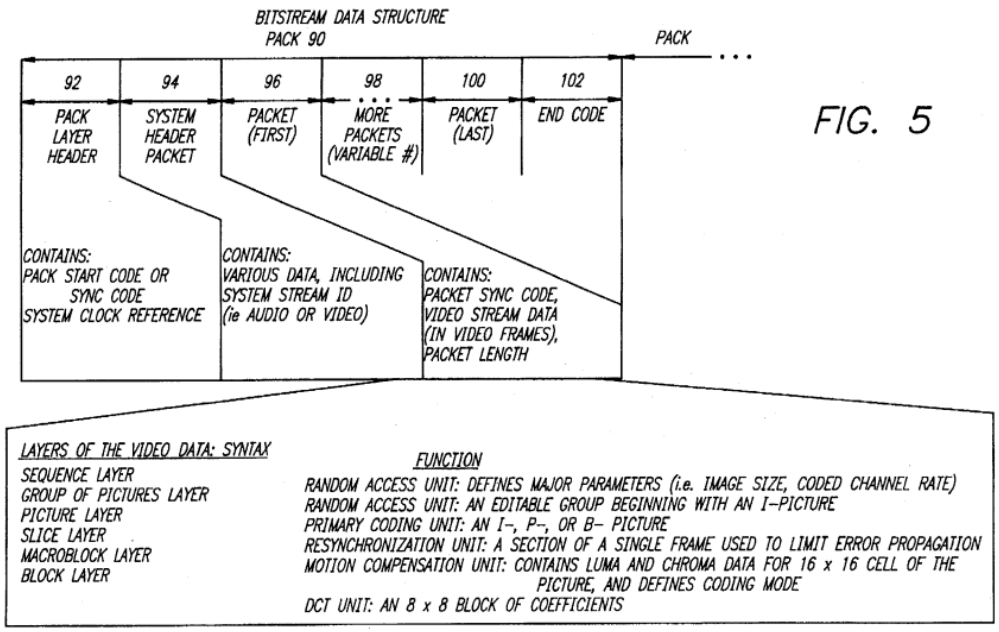
		identify the type of packet being analyzed, and to skip any non-video packets. When a packet sync code has been located, as at 150, so-called stuffing data is removed, and the length of the packet data field is updated to reflect the decrease in length of the packet word resulting from the removal of the stuffing data, as at 152. Stuffing data, also referred to as stuffing bits or stuffing bytes, are typically code words that may be inserted into a compressed bitstream to increase the bit rate of the stream to that required by the digital storage or transmission medium. The stuffing data may later be discarded by the decoder during the decoding process." Auld, 11:31–52.  FIG. 7 Auld, Fig. 7. To the extent that Plaintiff contends that Auld does not explicitly or inherently disclose this element, or render obvious this element, this element is obvious in combination with one or more references and/or with the knowledge of a person of ordinary skill in the art, as set forth in Defendant's Invalidity Contentions.

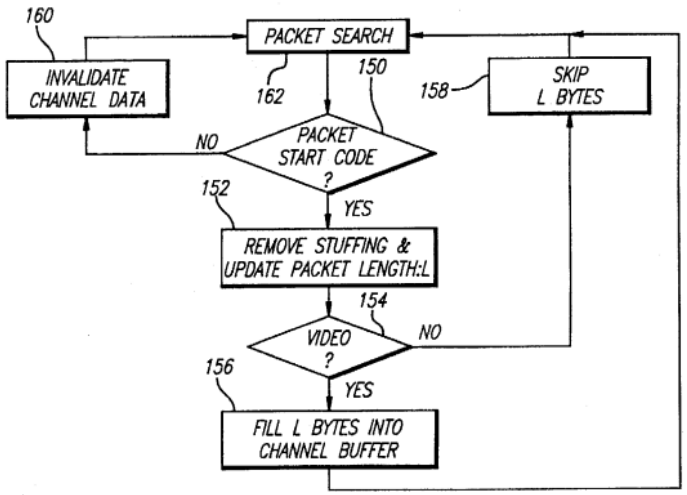
Exhibit C-2

Claim 24		
24[p]	The communications device of claim 19	Auld discloses, expressly or inherently, the transmitting device of claim 19 under any proposed construction. *See* exemplary disclosures and corresponding discussion for claim 19.
24	wherein the packet types include a data packet.	Auld discloses and/or renders obvious this limitation. *See, e.g.*: "The logical structure of a typical MPEG system bitstream is illustrated in FIG. 5, which represents the highest level MPEG data structure, and is often termed an ISO (International Standards Organization) system stream. An MPEG ISO system bitstream typically includes a sequence of packs 90. A typical pack 90, in turn, includes a sequence pack layer headers 92, a system header packet 94, a sequence of packets 96, 98, and 100, and an end code 102. The pack layer header 92 contains a pack start code, or sync code, used for synchronization purposes, and a system clock value. The system header packet 94 contains a variety of information commonly referred to as system housekeeping data, and a system stream ID which is used to differentiate among multiple system streams. A sequence of one or more packets typically contains either encoded audio or encoded video stream data. The final element of a pack 90 is an end code 102. It should be noted that any one packet 96, 98, 100 may carry either audio or video data, but not both simultaneously. With respect to a packet containing video data, typically a video packet header and one or more video frames is contained within the packet. Further illustrated in FIG. 5 are the hierarchical layers comprising a video packet 96. The various layers comprising the video data stream include a sequence layer, a group of pictures layer, a picture layer, a slice layer, a macroblock layer, and a block layer. The functions associated with each of these layers is also provided in FIG. 5. It should be noted that each packet or layer may contain one or more synchronization codes, referred to in the art as sync code, which are necessary to properly synchronize and align the encoded bitstream." Auld, 10:16–45.



Auld, Fig. 5.

"Turning now to FIG. 7, there is shown a flow diagram illustrating the process of extracting the video bitstream component of a multiplexed system bitstream that is performed by the system stream pre-parser 63. The system stream pre-parser 63 is intended to pass only the video bitstream component of a multiplexed system bitstream. Any non-video information components, such as the audio portion of the system bitstream, are discarded by the system stream pre-parser 63. Essentially, the packet sync code, stream ID, and packet length information at the packet layer are used to identify the type of packet being analyzed, and to skip any non-video packets. When a packet sync code has been located, as at 150, so-called stuffing data is removed, and the length of the packet data field is updated to reflect the decrease in length of the packet word resulting from the removal of the stuffing data, as at 152. Stuffing data, also referred to as stuffing bits or stuffing bytes, are typically code words that may be inserted into a compressed bitstream to increase the bit rate of the

		stream to that required by the digital storage or transmission medium. The stuffing data may later be discarded by the decoder during the decoding process." Auld, 11:31–52.  FIG. 7 Auld, Fig. 7. "After removal of the stuffing data and updating the length of the packet data field, the stream ID of the packet is read to determine whether the bitstream is a video bitstream, as at 154. If the system ID indicates that the bitstream does indeed contain video data, the data representative of the video information is transferred to the channel buffer 54, as shown at 156. If the system ID indicates that the packet does not contain video data, the number of bytes comprising the non-video data are skipped, as at 158, and subsequent packets are searched, as at 162. If a packet sync code cannot be located, the data in the fixed rate channel 50 is considered invalid and is purged, as at 160, and subsequent packet searching continues, as at 162." Auld, 11:53–65.

		To the extent that Plaintiff contends that Auld does not explicitly or inherently disclose this element, or render obvious this element, this element is obvious in combination with one or more references and/or with the knowledge of a person of ordinary skill in the art, as set forth in Defendant's Invalidity Contentions.
Claim 25		
25[p]	The communications device of claim 19	Auld discloses, expressly or inherently, the transmitting device of claim 19 under any proposed construction. *See* exemplary disclosures and corresponding discussion for claim 19.
25	wherein the packet types include a control packet.	One of ordinary skill would find this limitation disclosed either expressly or inherently in the teachings of the reference and its incorporated disclosures taken as a whole, or in combination with the state of the art at the time of the alleged invention, as evidenced, for example, by substantial other references identified in Defendant's invalidity contentions and accompanying charts. Rather than repeat those disclosures here, they are incorporated by reference into this chart.
Claim 26		
26[p]	The communications device of claim 19	Auld discloses, expressly or inherently, the transmitting device of claim 19 under any proposed construction. *See* exemplary disclosures and corresponding discussion for claim 19.
26	wherein the symbols are transmitted to a switch network.	One of ordinary skill would find this limitation disclosed either expressly or inherently in the teachings of the reference and its incorporated disclosures taken as a whole, or in combination with the state of the art at the time of the alleged invention, as evidenced, for example, by substantial other references identified in Defendant's invalidity contentions and accompanying charts. Rather than repeat those disclosures here, they are incorporated by reference into this chart.
Claim 28		
28[p]	The communications device of claim 19	Auld discloses, expressly or inherently, the transmitting device of claim 19 under any proposed construction. *See* exemplary disclosures and corresponding discussion for claim 19.
28	wherein the synchronization primitive	One of ordinary skill would find this limitation disclosed either expressly or inherently in the teachings of the reference and its incorporated disclosures taken as a whole, or in combination with

	is made up of a sequence of bit-string synchronization symbols.	the state of the art at the time of the alleged invention, as evidenced, for example, by substantial other references identified in Defendant's invalidity contentions and accompanying charts. Rather than repeat those disclosures here, they are incorporated by reference into this chart.

Exhibit C-8

Invalidity of the ’798 Patent in View of Additional Prior Art References

This chart provides examples of where the prior art reference discloses, either expressly or inherently, the claim elements of asserted U.S. Patent No. 7,746,798 (“the ’798 patent”). The citations herein are exemplary and are not exhaustive; Defendant has cited the most relevant portions of the identified prior art. For each citation, Defendant hereby incorporates by reference the entire cited prior art reference, and reserves the right to rely on uncited portions of each identified prior art reference. To provide context or identify the state of the art at the time of the alleged invention, Defendant reserves the right to rely on: (1) uncited portions of the identified prior art; (2) other prior art not identified herein; (3) references that show the state of the art (irrespective of whether such references themselves qualify as prior art); (4) factual testimony from the inventors or authors of the prior art references, or purveyors of prior art devices; and (5) expert testimony.

Where Defendant cites a figure, the citation includes any text associated with the figure. Where Defendant cites passages relating to a figure, the citation includes the associated figure any other text associated with that figure. Where Defendant cites particular prior art as disclosing a claim element, that citation equally applies to all identical or similar claim elements in this chart. Defendant reserves the right to further assert that a claim element is inherent, the element would have been obvious to a person of ordinary skill in the art, or the element is disclosed in other prior art that may be combined with the cited prior art. To establish the inherency of certain features of the prior art to invalidate the asserted claims, Defendant may rely on cited or uncited portions of the prior art, other documents, factual testimony, and expert testimony.

This invalidity chart is based on information presently available to Defendant, and the information herein is provided as a non-limiting example.

U.S. Patent No. 5,917,830 (“Chen”) was filed on October 18, 1996, and published on June 29, 1999; therefore, Chen is prior art under at least 35 U.S.C. § 102(b).

U.S. Patent No. 6,430,230 (“Cunningham”) was filed on June 30, 1999, and published on August 6, 2002; therefore, Cunningham is prior art under at least 35 U.S.C. § 102(e)(1).

U.S. Patent No. 5,801,781 (“Hiroshima”) was filed on June 19, 1996, and published on September 1, 1998; therefore, Hiroshima is prior art under at least 35 U.S.C. § 102(b).

WIPO Patent Publication No. 2000/011831 (“Haartsen”) was filed on August 17, 1999, and published on March 2, 2000; therefore, Haartsen is prior art under at least 35 U.S.C. § 102(b).

Exhibit C-8

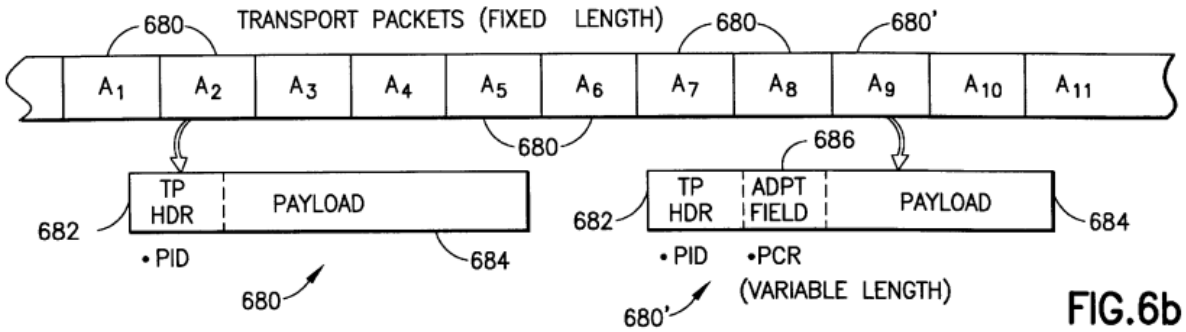
U.S. Patent No. 5,974,464 ("Shin") was filed on September 30, 1996, and published on October 26, 1999; therefore, Shin is prior art under at least 35 U.S.C. § 102(b).

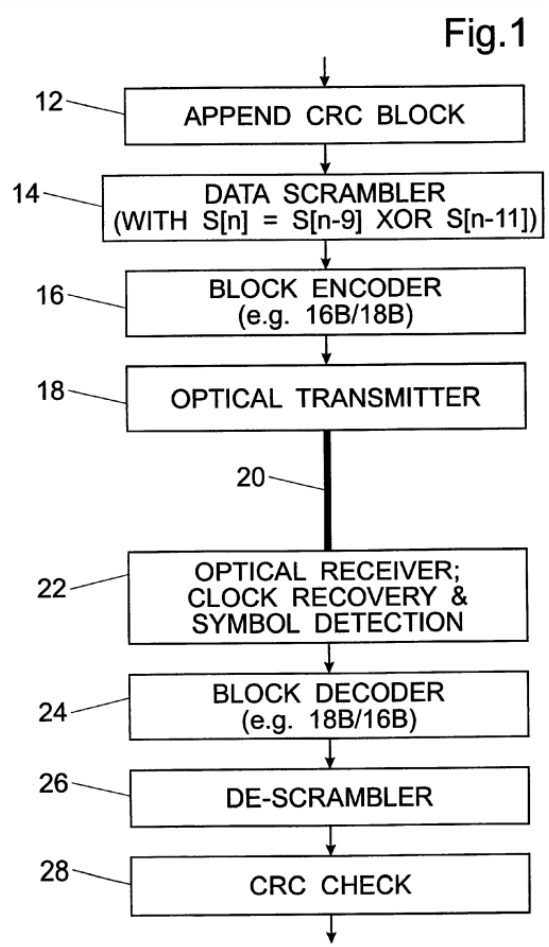
U.S. Patent No. 6,498,568 ("Austin") was filed on September 24, 1997, and published on December 24, 2002; therefore, Austin is prior art under at least 35 U.S.C. 102(e)(1).

This chart refers to the above references collectively as the "Additional Prior Art References."

Exhibit C-8

Claim Limitation		Chen
Claim 11		
11[p]	A method for identifying packet types of packets of symbols, the method comprising:	To the extent the preamble is limiting, Chen discloses this element and/or renders it obvious either alone or in combination with other references and/or the knowledge of one of ordinary skill in the art. *See, e.g.*: "In the MPEG-2 system (and the similar DigiCipher® II system proprietary to General Instrument Corporation, the assignee hereof) a transport stream, or transport multiplex is made up of a contiguous set of fixed length packets. Each packet is 188 total bytes in length, with the first four of those bytes being defined as the packet header. The payload portion of each packet is thus normally 184 bytes. However, a variable length adaptation field may be provided to extend the header, when required. When an adaptation field is present, the payload portion of the packet will be correspondingly shorter. Various timing and identification information is provided in different portions of the transport stream. These include a packet identifier (PID) found in the transport header of each transport packet to provide a reference number for identifying the transport packets carrying a specific service component. This number is included in a service definition or "service map" used by the receiver to identify those transport packets required to reconstruct a television program signal. The PID may also be referenced for various grooming and remultiplexing functions. In the case of video, audio or isochronous control data, the stream of packets labeled with a single PID represents a single video, audio or isochronous data service elementary stream, respectively. Each type of packet will have a different PID identifying the packet type." Chen, 14:45–15:3. "As indicated in FIG. 6b, the transport header of each transport packet includes the PID which identifies the particular service component carried by the transport packet. The PCR will be carried by an adaptation field to provide timing information for a desired service. At a receiver, the PCR for the desired service is detected from the adaptation field. The PIDs of the transport packets will then be monitored in accordance with the timing established by the PCR to recover those transport packets carrying a particular component of the service to be processed."

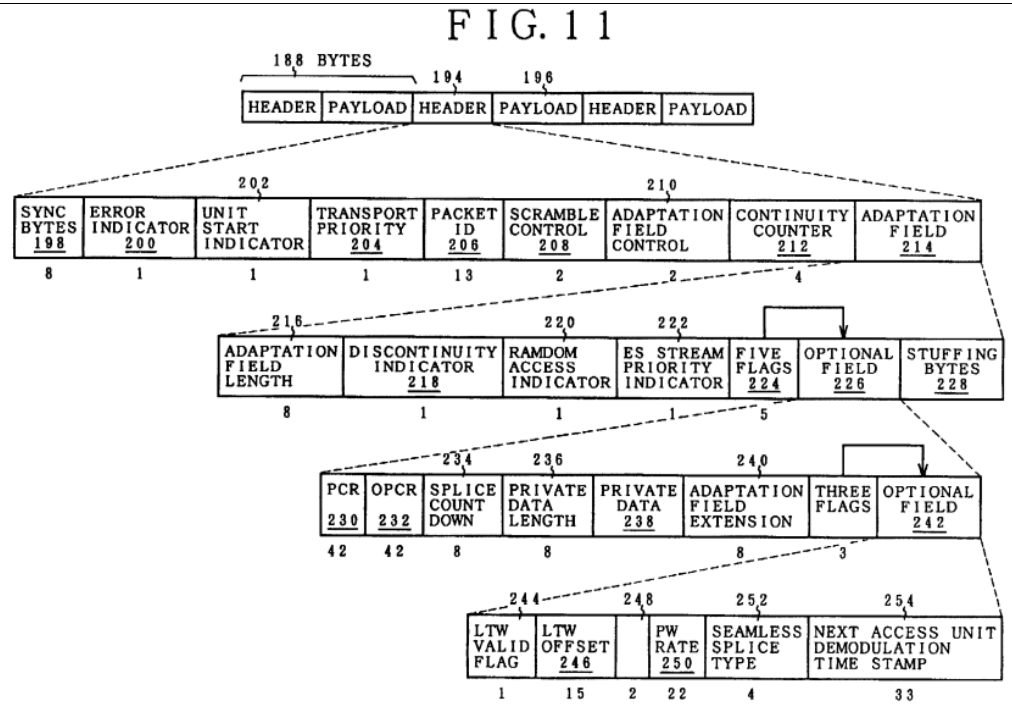
		Chen, 16:41–50.  Chen, Fig. 6b. To the extent the preamble is limiting, Cunningham discloses this element and/or renders it obvious either alone or in combination with other references and/or the knowledge of one of ordinary skill in the art. *See, e.g.*: "According to one aspect of this invention there is provided a method of encoding payload bits for transmission over a communications link such that the running disparity of the transmitted symbol stream is constrained within a predetermined limit, and said payload bits comprising either data or control information …." Cunningham, 1:58–63. "Referring to FIG. 1, a data frame intended to be communicated over an optical fibre channel 20 typically comprises a binary digital message (payload) to be transferred between stations on the network; this message may comprise user data, or control information for managing operation of the network and its components. A circuit 12 generates an associated cyclic redundancy check (CRC) block containing check data derived from this message in known manner in accordance with a predetermined CRC algorithm, and appends the CRC block to the message." Cunningham, 2:65–3:7.



Cunningham, Fig. 1.

"Control information is conveyed by using selected n-bit data block values, together with a distinctive value of 10 for the pair of label bits a and b to indicate that the associated data block represents control information rather than actual data. The data block values used for this purpose

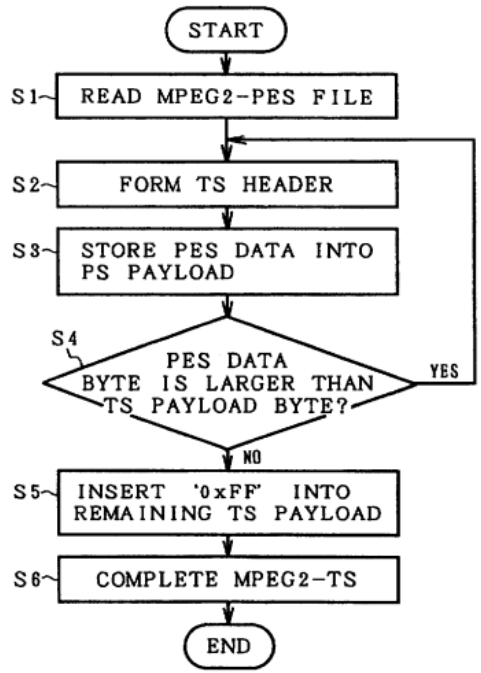
		are selected to be values which inherently have a disparity of zero, such as (for 8-bit blocks) 00001111 or 10101010. It will be noted that the label bit pair value 10 likewise has zero disparity, so the complete encoded block also has zero disparity." Cunningham, 4:23–31. "The encoded blocks are transmitted over the optical fibre channel 20 (leftmost symbol first), and upon receipt at their destination they are detected by a receiver 22 where compensation is made in known manner for the effects of timing drift and other signal degradation to produce a restored digital data stream." Cunningham, 4:45–50. To the extent the preamble is limiting, Hiroshima discloses this element and/or renders it obvious either alone or in combination with other references and/or the knowledge of one of ordinary skill in the art. *See, e.g.*: "FIG. 11 shows the details of the MPEG2-TS. Each packet of the MPEG2-TS has a prefix length of 188 bytes. The header 194 consists of four bytes and the payload 196 consists of 184 bytes. The header 194 is constructed by sync bytes 198, an error indicator 200, a unit start indicator 202, a transport priority 204, a packet ID 206, a scramble control 208, an adaptation field control 210, a continuity counter 212, and an adaptation field 214. The adaptation field 214 is constructed by an adaptation field length 216, a discontinuity indicator 218, a random access indicator 220, an elementary stream priority indicator 222, a 5-flag 224, an optional field 226, and stuffing bytes 228. Among them, the optional field 226 is constructed by a program clock reference 230, an original program clock reference field 232, a splice count down 234, a private data length 236, private data 238, an adaptation field extension 240, 3-flag 241, and an optional field 242." Hiroshima, 10:51–67.



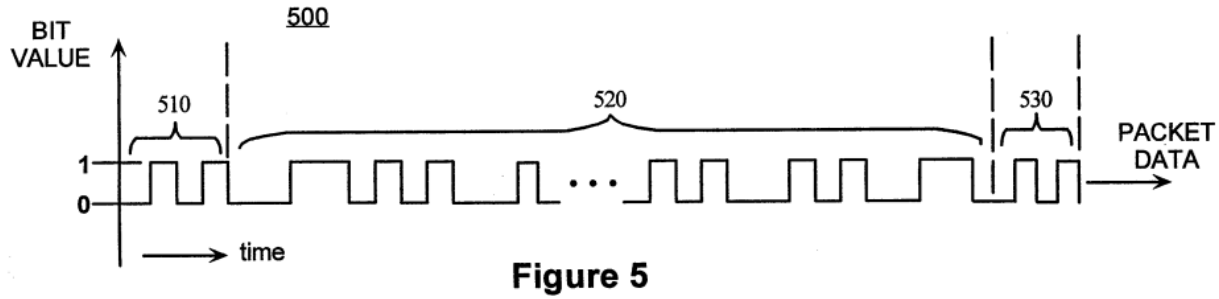
Hiroshima, Fig. 11.

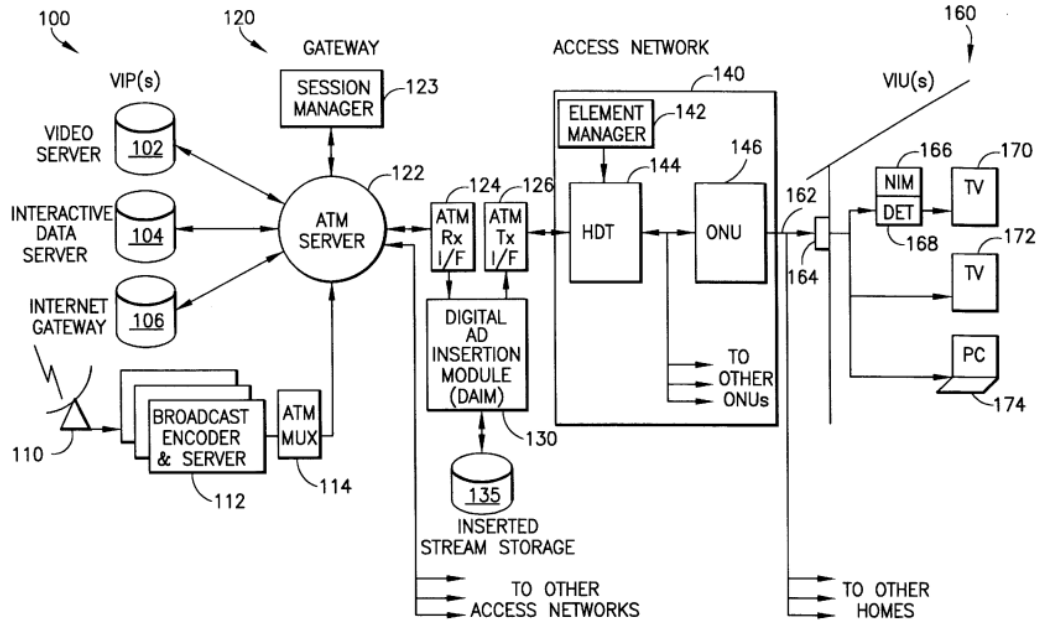
"FIG. 15 shows a process for forming the MPEG2-TS from the PES packet 82 by a multiplexing process in the multiplexer 34 in FIG. 5. First in step S1, the MPEG2-PES file is read. The TS header 194 in FIG. 11 is formed in step S2. The TS header 194 is formed in a manner such that the parameters such as error indicator 200 indicative of the presence or absence of the bit error in the packet, unit start indicator 202 indicating whether a new PES packet appears in the TS payload or not, packet ID 206 indicating whether the packet in the payload is video or audio, adaptation field 214, and adaptation field control 210 indicative of the presence or absence of the payload 196 are set in accordance with the state of the payload storage data of the 184-byte unit divided from the MPEG2-PES packet which had been read in step S1."

Hiroshima, 11:13–27.

		FIG. 15  Hiroshima, Fig. 15. To the extent the preamble is limiting, Haartsen discloses this element and/or renders it obvious either alone or in combination with other references and/or the knowledge of one of ordinary skill in the art. *See, e.g.*: "The present invention fulfills the above-described and other needs by providing a multi-part digital preamble for use in transmitting digital data packets. Advantageously, a preamble according to the invention substantially reduces the overall length of DC-free sequences that are transmitted with each data packet in a digital FM system. According to exemplary embodiments, a multi-part

		preamble includes a short, substantially DC-free leading part which enables coarse estimation of a prevailing DC level at an FM receiver. The exemplary preamble also includes a synchronization part which is not necessarily substantially DC-free and which carries timing and/or other useful information (e.g. , channel identification, destination address, etc.) that can vary from data packet to data packet (e.g. , depending on sender, recipient, time of day, etc.).” Haartsen, 5:2–15. “To alleviate this problem, the present invention discloses that a digital preamble can be strategically divided into three or more parts so that synchronization and other information can be included within the preamble while still enabling accurate DC estimation. Specifically, a short leading DC-free part of the preamble (e.g., alternating logical l 's and 0's, or any other sequence including an equal number of logical l's and logical 0's) can be used to provide initial, coarse DC offset estimation. Thereafter, a synchronization part of the preamble, which is not necessarily DC free, but which is protected by code to prevent detection errors, can be used not only to synchronize detection of a data packet, but also to carry timing and/or other useful information (e.g., channel identification, destination address, time of day, user identification, etc.).” Haartsen, 12:13–13:2. “An exemplary multi-part preamble according to the invention is depicted in Figure 5. In the figure, a three-part preamble 500 is shown to include a DC-free leading sequence 510, followed by a multiple-bit synchronization word 520 which is not necessarily DC free and which can be different for each data packet. The synchronization word 520, which can include timing and/or other information and which is coded to protect against detection errors, is in turn followed by a trailing DC-free sequence 530. Although the leading and trailing DC-free sequences 510, 530 are shown in the figure as 4-bit sequences of alternating logical l 's and 0's, those of ordinary skill will appreciate that each DC-free sequence 510, 530 can, in practice, be shorter or longer in length and can include any DC-free sequence having an equal number of logical l's and logical 0's.” Haartsen, 13:13–14:2.

		 Haartsen, Fig. 5. To the extent the preamble is limiting, Austin discloses this element and/or renders it obvious either alone or in combination with other references and/or the knowledge of one of ordinary skill in the art. *See, e.g.*: "This system uses up to 5 pre-defined data packet sizes and up to 16 user defined data message types within each packet. Each packet has Forward Error Connection (FEC) and bit interleaving applied to it on transmission. Error checking is performed using a standard CRC. The system uses standard run in and byte synchronisation techniques. The byte synchronisation word used is 32 _bits long and there are 5_different words used, one to identify each of the different packet types." Austin, 4:37-45. To the extent that Plaintiff contends that the Additional Prior Art References do not explicitly or inherently disclose this element, or render obvious this element, this element is obvious in combination with one or more references and/or with the knowledge of a person of ordinary skill in the art, as set forth in Defendant's Invalidity Contentions.
11(a)	receiving a synchronization symbol by a receiver indicating a packet type, each packet type having a	Chen discloses and/or renders obvious this limitation. *See, e.g.*: "FIG. 1 is a block diagram of a Fiber-to-the-Curb Digital Video Distribution Network in accordance with the present invention. The distribution network includes Video Information Providers (VIPs) shown generally at 100, a gateway, shown generally at 120, an access network, shown generally at

	different synchronization symbol;	140, and the home of a Video Information User (VIU), shown generally at 160. The gateway 120 may comprise an Ll Gateway manufactured by Teleco. The VIPs may include a video server 102, an interactive data server 104, and an internet gateway 106, all of which communicate with an asynchronous transfer mode (ATM) network 122 via, for example, an OC-3 system, where optical fiber is used to transmit data at 155.52 Mbit/sec. The ATM network 122 may also receive data via a satellite receiving antenna 110, broadcast encoder and server 112, and ATM multiplexer 114. The ATM network 122 communicates with a session manager 123." Chen, 4:11–28.  FIG.1 Chen, Fig. 1. "FIG. 4 is a block diagram of an Insertion Processing Unit (IPU) of a Digital Ad Insertion Module in accordance with the present invention. The IPU, shown generally at 400, receives the main stream at a packet framer 405, while the insertion stream is received at another packet framer 410. After

processing by the packet framer 405, the main stream is provided to a Main Stream Parser (MSP) 415."

Chen, 5:62–6:1.

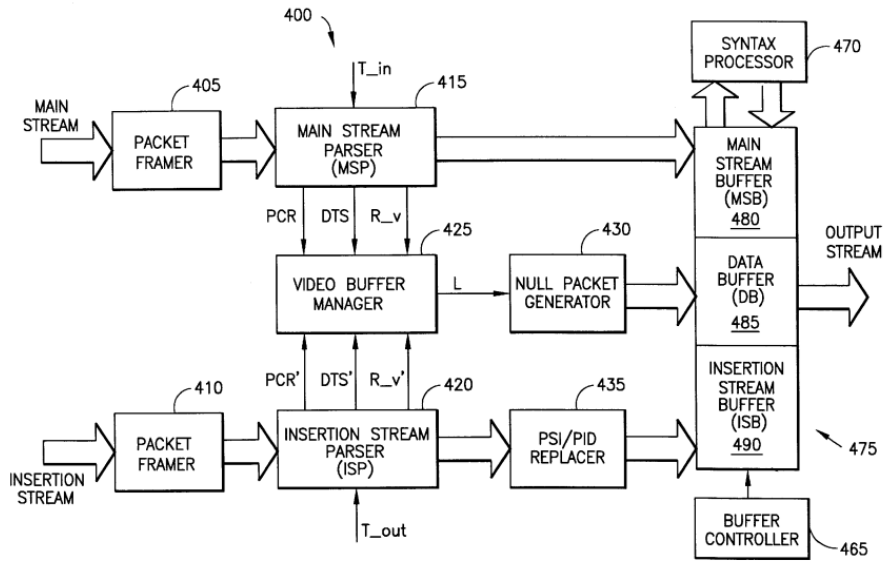


FIG.4

Chen, Fig. 4.

"The main stream parser 415 provides the main stream data to a main stream buffer 480, which communicates with a syntax processor 470. The main stream buffer 480 is part of the output buffer 475. The insertion stream parser 420 provides the insertion stream data to a Program Specific Information (PSI)/Program Identifier (PID) Replacer 435, which replaces the PSI tables and PIDs of the insertion stream with those of the main stream. The PSI/PID replacer 435 retrieves the relevant information from the main stream using a communication path which is not shown. The PSI and PID data of the main stream provide the information to connect the various audio, video, and data packets to a particular programming service. Specifically, this is accomplished using a Program Association Table (PAT) and a Program Map Table (PMT). The insertion stream is then

		provided to an Insertion Stream Buffer (ISB) 490, which is part of the output buffer 475. The ISB 490 communicates with a syntax processor 470. Furthermore, the output buffer 475 receives commands from a buffer controller 465 to provide an output stream with the insertion stream seamlessly spliced into the main stream." Chen, 6:33–54. "FIG. 5 illustrates the processing flow of the Insertion Processing Unit of FIG. 4 in accordance with the present invention. Note that the steps of FIG. 5 are shown as occurring in a sequential manner for clarity, but many of the steps may occur concurrently and/or sequentially. At block 502, the IPU outputs the main transport stream from the main stream buffer (MSB) 480. The IPU is essentially in a bypass mode at this time since the insertion stream has not yet been inserted. At block 506, the main stream parser parses the main stream to locate the splicing signal T-- in. Note that T-- in need not be carried in the main stream, but may be provided locally, for example, at a cable system headend. When T-- in is detected, the IPU continues to output the main transport stream from the main stream buffer. At block 508, the main stream parser parses the main stream to locate the closest packet with an anchor picture (e.g., I or P picture) start code and passes that packet to the syntax processor 470. This packet will be the last packet of the main stream. To ensure that the resulting data stream is MPEG compliant, no data from the next frame or sequence of frames may be included in this last main stream packet. At block 512, the syntax processor adds a sequence end code field to the last packet of the main stream. The sequence end code is a twenty-four bit code which indicates the end of a sequence of pictures. At block 514, the PSI/PID replacer 435 starts to replace the PSI/PID of the insertion stream." Chen, 12:28–53.

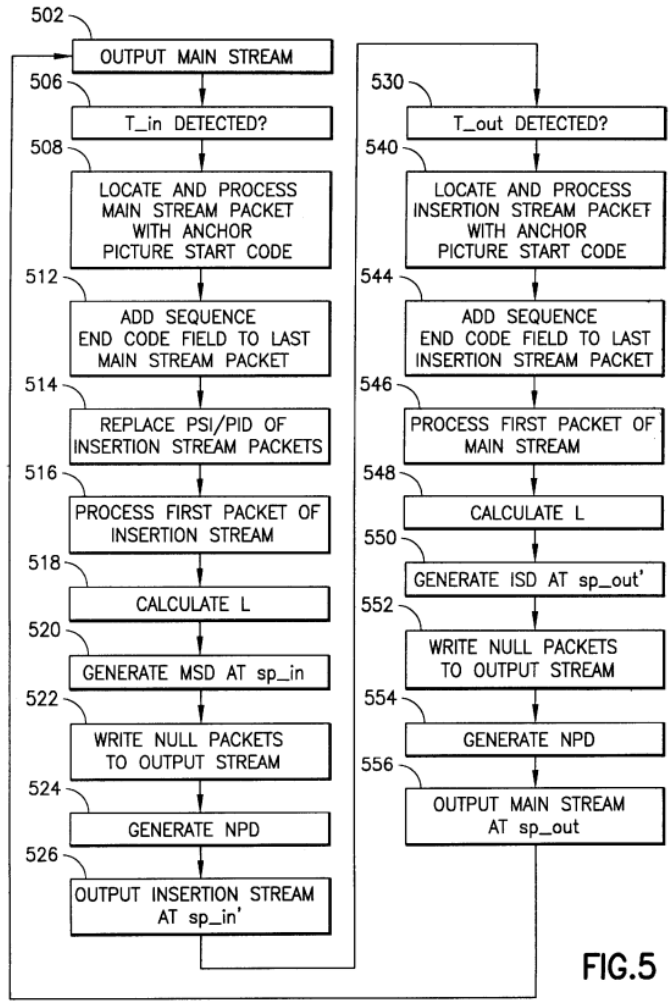
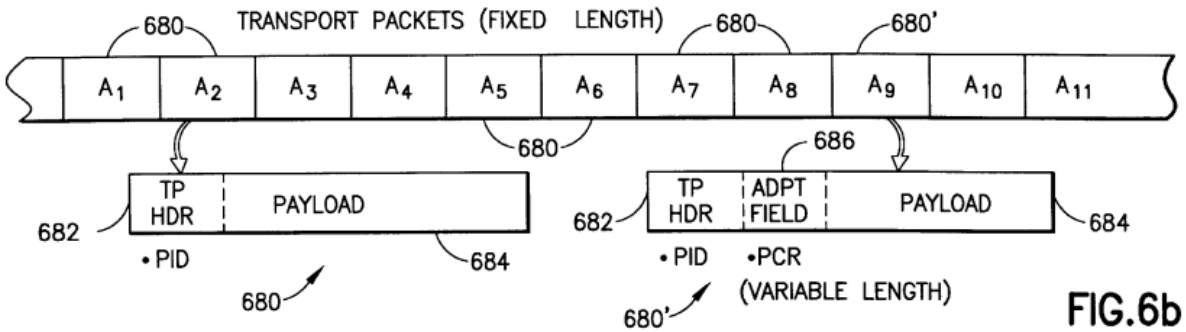


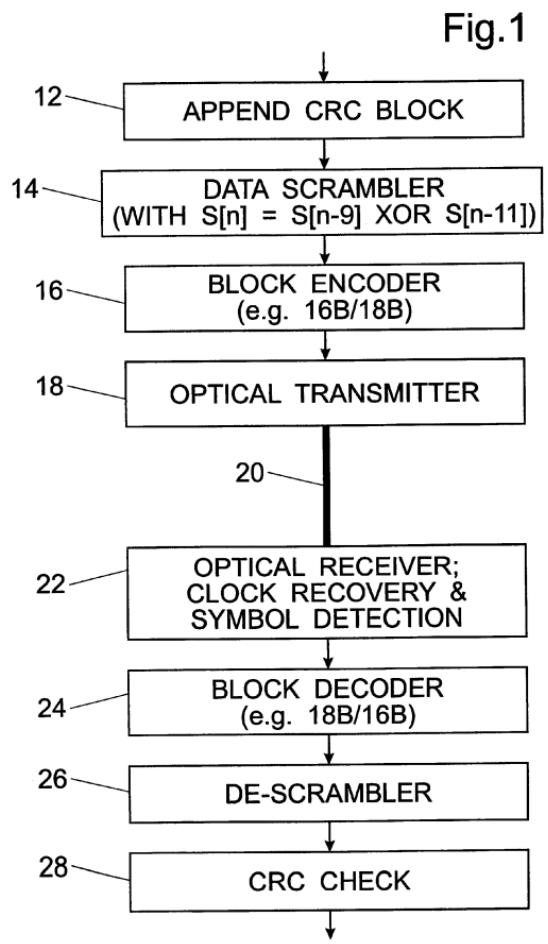
FIG.5

Chen, Fig. 5.

"In the MPEG-2 system (and the similar DigiCipher® II system proprietary to General Instrument Corporation, the assignee hereof) a transport stream, or transport multiplex is made up of a

		contiguous set of fixed length packets. Each packet is 188 total bytes in length, with the first four of those bytes being defined as the packet header. The payload portion of each packet is thus normally 184 bytes. However, a variable length adaptation field may be provided to extend the header, when required. When an adaptation field is present, the payload portion of the packet will be correspondingly shorter. Various timing and identification information is provided in different portions of the transport stream. These include a packet identifier (PID) found in the transport header of each transport packet to provide a reference number for identifying the transport packets carrying a specific service component. This number is included in a service definition or "service map" used by the receiver to identify those transport packets required to reconstruct a television program signal. The PID may also be referenced for various grooming and remultiplexing functions. In the case of video, audio or isochronous control data, the stream of packets labeled with a single PID represents a single video, audio or isochronous data service elementary stream, respectively. Each type of packet will have a different PID identifying the packet type. Timing information carried by the transport stream includes a program clock reference (PCR) which effectively represents a sample of the system time clock (STC) time base that underlies the service composed of the PIDs referenced in the service map. The PID carrying the packet with the PCR is also referenced in the service map. The video, audio and isochronous data components of a service are locked through a defined relationship to the system time clock. The PCR serves to define the transport rate, in the sense that between any two successive PCRs in one PID, the transport rate is constant and nominally equal to the system time clock rate times the ratio of the total number of transport bytes between the PCRs divided by the difference in the PCRs in units of system time clock ticks." Chen, 14:45–15:17. "As indicated in FIG. 6b, the transport header of each transport packet includes the PID which identifies the particular service component carried by the transport packet. The PCR will be carried by an adaptation field to provide timing information for a desired service. At a receiver, the PCR for the desired service is detected from the adaptation field. The PIDs of the transport packets will

		then be monitored in accordance with the timing established by the PCR to recover those transport packets carrying a particular component of the service to be processed." Chen, 16:41–50.  Chen, Fig. 6b. Cunningham discloses and/or renders obvious this limitation. *See, e.g.*: "Referring to FIG. 1, a data frame intended to be communicated over an optical fibre channel 20 typically comprises a binary digital message (payload) to be transferred between stations on the network; this message may comprise user data, or control information for managing operation of the network and its components. A circuit 12 generates an associated cyclic redundancy check (CRC) block containing check data derived from this message in known manner in accordance with a predetermined CRC algorithm, and appends the CRC block to the message." Cunningham, 2:65–3:7.



Cunningham, Fig. 1.

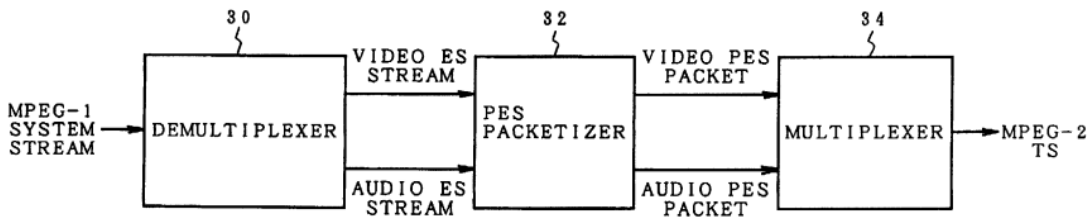
"Control information is conveyed by using selected n-bit data block values, together with a distinctive value of 10 for the pair of label bits a and b to indicate that the associated data block represents control information rather than actual data. The data block values used for this purpose are selected to be values which inherently have a disparity of zero, such as (for 8-bit blocks)

		00001111 or 10101010. It will be noted that the label bit pair value 10 likewise has zero disparity, so the complete encoded block also has zero disparity." Cunningham, 4:23–31. "The encoded blocks are transmitted over the optical fibre channel 20 (leftmost symbol first), and upon receipt at their destination they are detected by a receiver 22 where compensation is made in known manner for the effects of timing drift and other signal degradation to produce a restored digital data stream." Cunningham, 4:45–50. "There are many potential ways to achieve code block synchronization (i.e. correct alignment in the receiver of the label and payload fields). For example, one technique would be to transmit a 'comma sequence' during start-up. As an example of this technique consider a code with n=8 using the simple control payload scheme. A training control state of 1111 1111 would be transmitted as: [10 1111 1111 0000 0000][10 1111 1111 0000 0000][10 1111 1111 0000 0000] The receiver searches for the bit pair 10. This is found in only two locations: the correct location (the label field), and at the end of the run of eight 1's. However, if the receiver (incorrectly) identifies the label field to be the 10 bit pair at the end of the run of 1's, the following 16 bits are not a valid control payload (first 8 bits are not the complement of the second 8 bits). The receiver can therefore detect the mistake. It is also noted that this sequence contains a regular 010 pattern, which aids in phase-locked loop (PLL) clock acquisition." Cunningham, 7:5–23. Hiroshima discloses and/or renders obvious this limitation. *See, e.g.*: "FIG. 4 is a block diagram of the first embodiment of the invention in which the moving picture stream conversion in mode 1 is executed. A moving picture stream converting apparatus of FIG. 4 comprises a demultiplexer 30, a packetizer 32, and a multiplexer 34. The demultiplexer 30 receives an MPEG1 system stream as a conversion source and outputs elementary streams which were

separated into video and audio streams. The packetizer 32 forms packets constructing a packetized elementary stream of MPEG2 with respect to each of the video and audio elementary streams which are supplied from the demultiplexer 30. The packet is usually called a PES packet. The multiplexer 34 divides each of the video and audio PES packets into a TS packet having a fixed length of 188 bytes and subsequently multiplexes, and outputs the resultant data as an MPEG2-TS."

Hiroshima, 7:59–8:7.

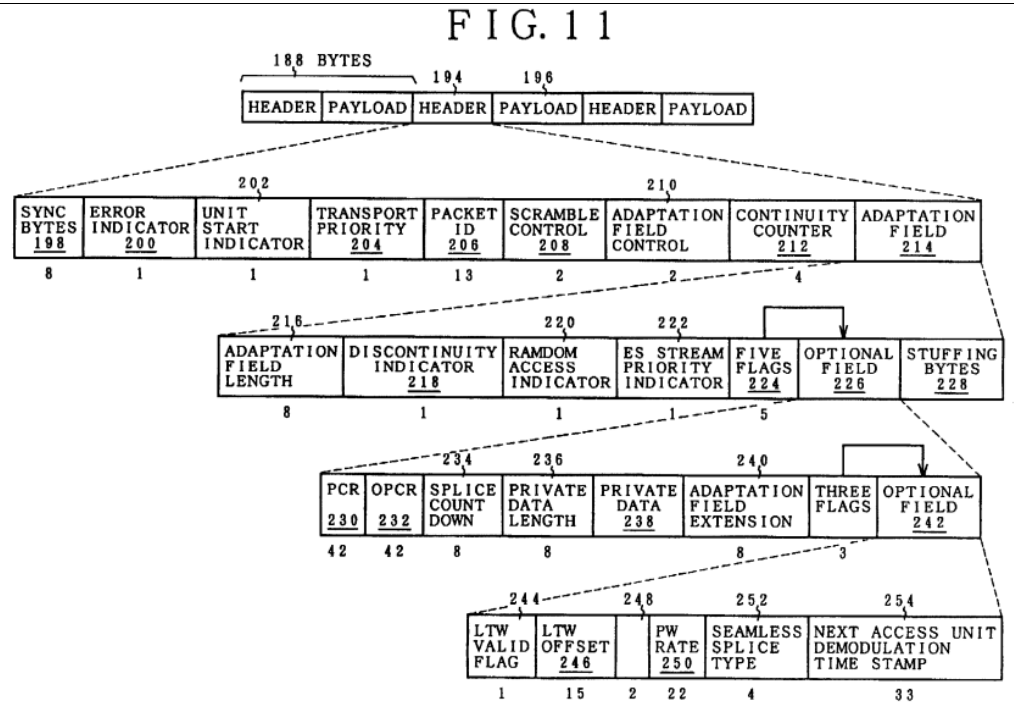
FIG. 4



Hiroshima, Fig. 4.

"FIG. 11 shows the details of the MPEG2-TS. Each packet of the MPEG2-TS has a prefix length of 188 bytes. The header 194 consists of four bytes and the payload 196 consists of 184 bytes. The header 194 is constructed by sync bytes 198, an error indicator 200, a unit start indicator 202, a transport priority 204, a packet ID 206, a scramble control 208, an adaptation field control 210, a continuity counter 212, and an adaptation field 214. The adaptation field 214 is constructed by an adaptation field length 216, a discontinuity indicator 218, a random access indicator 220, an elementary stream priority indicator 222, a 5-flag 224, an optional field 226, and stuffing bytes 228. Among them, the optional field 226 is constructed by a program clock reference 230, an original program clock reference field 232, a splice count down 234, a private data length 236, private data 238, an adaptation field extension 240, 3-flag 241, and an optional field 242."

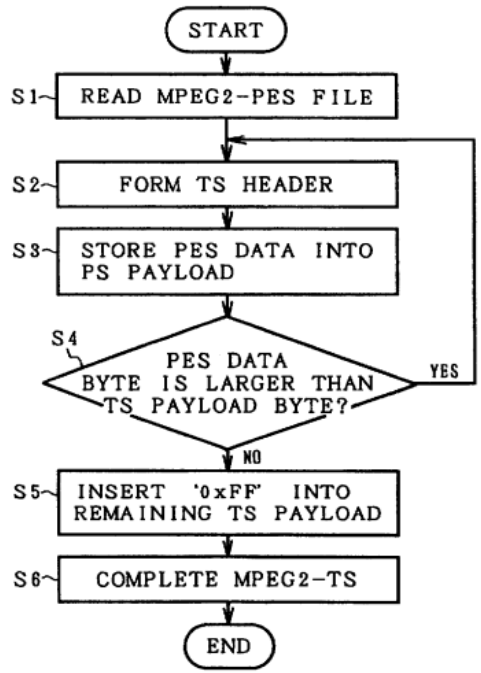
Hiroshima, 10:51–67.



Hiroshima, Fig. 11.

"FIG. 15 shows a process for forming the MPEG2-TS from the PES packet 82 by a multiplexing process in the multiplexer 34 in FIG. 5. First in step S1, the MPEG2-PES file is read. The TS header 194 in FIG. 11 is formed in step S2. The TS header 194 is formed in a manner such that the parameters such as error indicator 200 indicative of the presence or absence of the bit error in the packet, unit start indicator 202 indicating whether a new PES packet appears in the TS payload or not, packet ID 206 indicating whether the packet in the payload is video or audio, adaptation field 214, and adaptation field control 210 indicative of the presence or absence of the payload 196 are set in accordance with the state of the payload storage data of the 184-byte unit divided from the MPEG2-PES packet which had been read in step S1."

Hiroshima, 11:13–27.

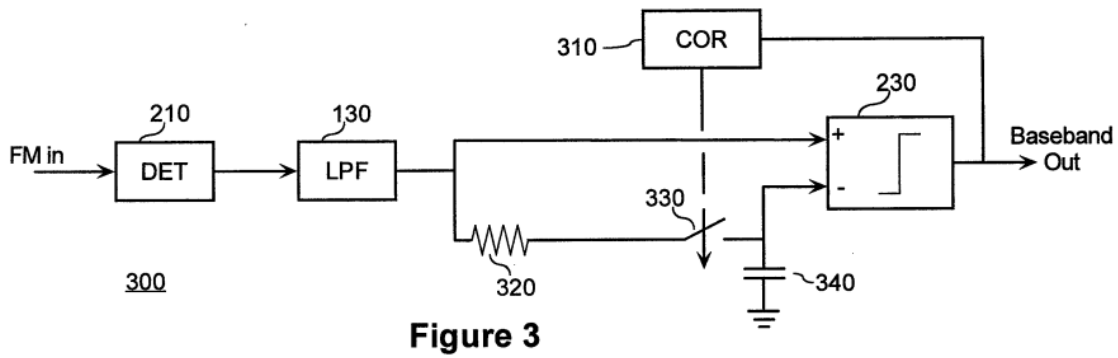
		FIG. 15  Hiroshima, Fig. 15. Haartsen discloses and/or renders obvious this limitation. *See, e.g.*: "The present invention fulfills the above-described and other needs by providing a multi-part digital preamble for use in transmitting digital data packets. Advantageously, a preamble according to the invention substantially reduces the overall length of DC-free sequences that are transmitted with each data packet in a digital FM system. According to exemplary embodiments, a multi-part preamble includes a short, substantially DC-free leading part which enables coarse estimation of a prevailing DC level at an FM receiver. The exemplary preamble also includes a synchronization

part which is not necessarily substantially DC-free and which carries timing and/or other useful information (e.g. , channel identification, destination address, etc.) that can vary from data packet to data packet (e.g. , depending on sender, recipient, time of day, etc.).”

Haartsen, 5:2–15.

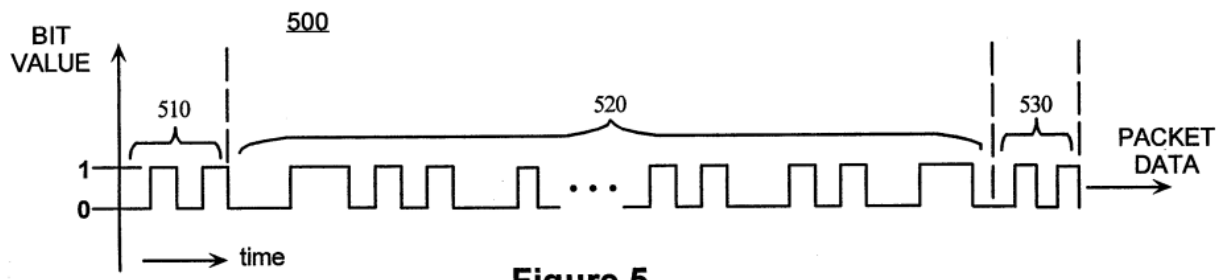
“An exemplary DC estimation processor is depicted in Figure 3, wherein an exemplary FM demodulation processor 300 is shown to include a correlation processor 310, a resistor 320, a switch 330 and a capacitor 340, as well as the FM detector 210, the low-pass filter 130 and the slicer 230 of Figure 2.”

Haartsen, 9:11–14.

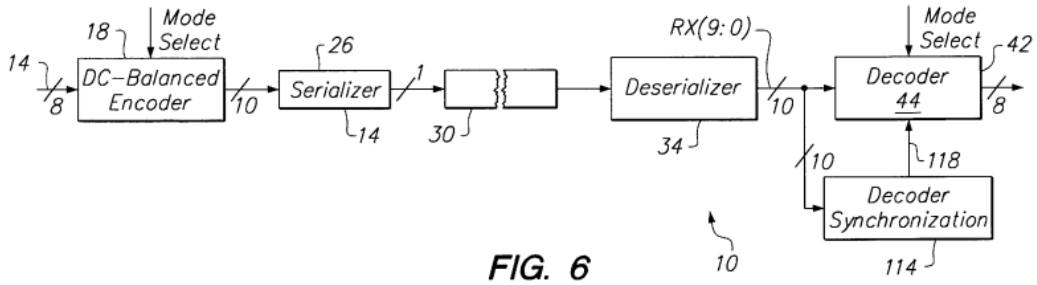


Haartsen, Fig. 3.

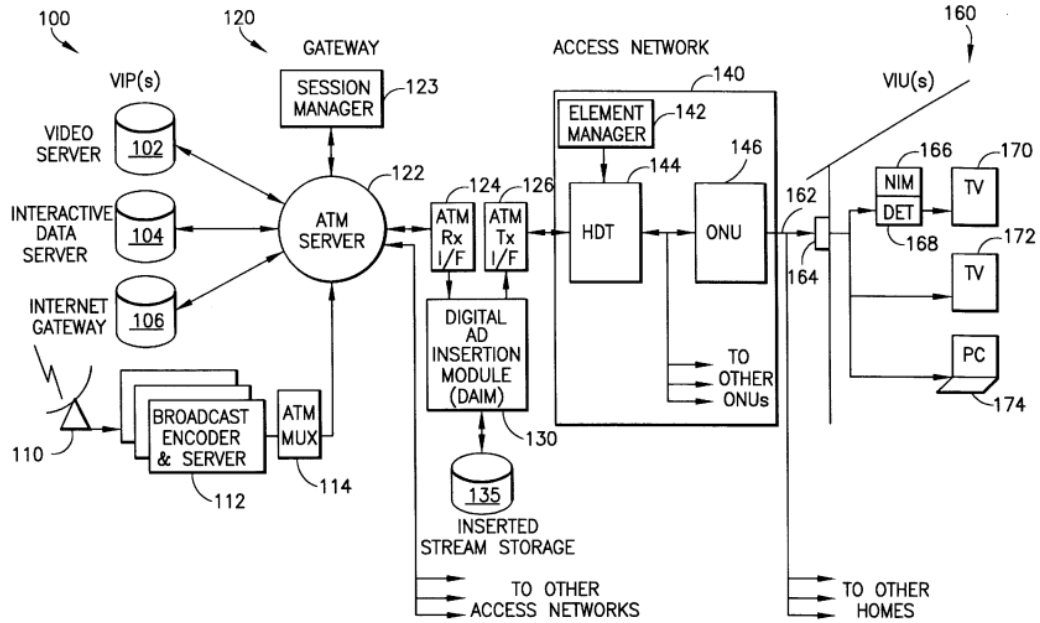
“To alleviate this problem, the present invention discloses that a digital preamble can be strategically divided into three or more parts so that synchronization and other information can be included within the preamble while still enabling accurate DC estimation. Specifically, a short leading DC-free part of the preamble (e.g., alternating logical l 's and 0's, or any other sequence including an equal number of logical l's and logical 0's) can be used to provide initial, coarse DC offset estimation. Thereafter, a synchronization part of the preamble, which is not necessarily DC free, but which is protected by code to prevent detection errors, can be used not only to synchronize

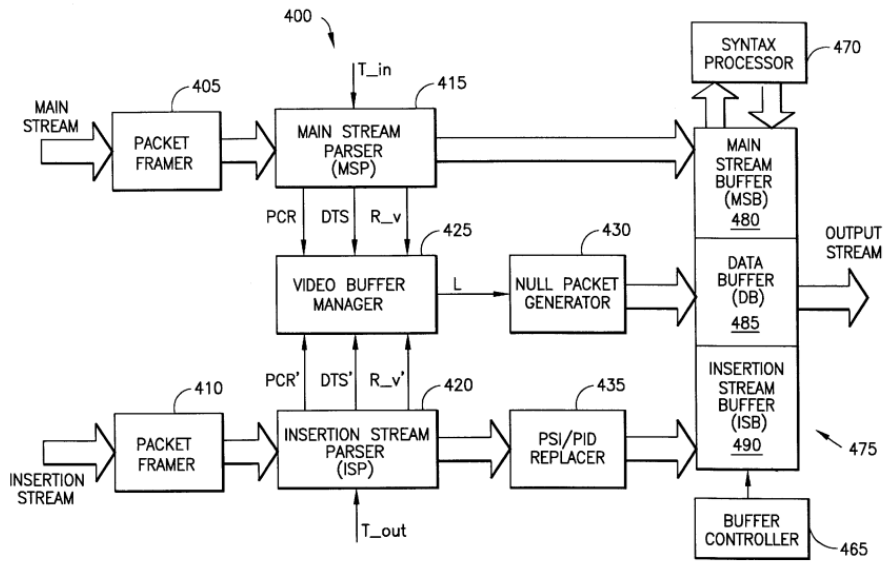
		detection of a data packet, but also to carry timing and/or other useful information (e.g., channel identification, destination address, time of day, user identification, etc.).” Haartsen, 12:13–13:2. “An exemplary multi-part preamble according to the invention is depicted in Figure 5. In the figure, a three-part preamble 500 is shown to include a DC-free leading sequence 510, followed by a multiple-bit synchronization word 520 which is not necessarily DC free and which can be different for each data packet. The synchronization word 520, which can include timing and/or other information and which is coded to protect against detection errors, is in turn followed by a trailing DC-free sequence 530. Although the leading and trailing DC-free sequences 510, 530 are shown in the figure as 4-bit sequences of alternating logical l 's and 0's, those of ordinary skill will appreciate that each DC-free sequence 510, 530 can, in practice, be shorter or longer in length and can include any DC-free sequence having an equal number of logical l's and logical 0's.” Haartsen, 13:13–14:2.  **Figure 5** Haartsen, Fig. 5. “When the exemplary preamble 500 is received by the system 300 of Figure 3, the leading DC-free sequence 510 induces a coarse estimate of the baseband DC offset on the storage capacitor 340. Although the estimate is coarse, it is sufficient to permit successful detection of the coded synchronization word 520 (e.g., by the correlation processor 310). Methods for coding of a synchronization word to protect against erroneous detection in the presence of some DC-offset are

		known to those of skill in the art. Once the correlation processor 310 has detected the synchronization word 520, timing between received bits is well known. Consequently, the correlation processor 310 can open the switch 330 precisely at the end of the trailing DC-free sequence 530 to thereby store an accurate estimate of the DC offset for reception and detection of the incoming digital data packet." Haartsen, 14:3–14. Shin discloses and/or renders obvious this limitation. *See, e.g.*: "It may thus be appreciated that each eight-bit code converted into a 10-bit encoded character during high-transition mode operation includes four or more logical transitions. Similarly, each eight-bit code converted into a 10-bit encoded character during low-transition mode operation includes less than four logical transitions. These sets of 10-bit encoded characters capable of being produced during high-transition and low-transition modes of operation may be characterized as a high-transition set of "in-band" encoded characters, and a low-transition set of in-band encoded characters, respectively. Beyond the 256 in-band characters within the high-transition set and the 256 in-band characters of the low-transition set, there exists a high-transition set of 256 out-of-band 10-bit characters and a low-transition set of 10-bit characters. In accordance with another aspect of the invention, various synchronization and other special characters are defined using the high-transition and low-transition sets of out-of-band characters. Each character corresponding to one of these "out-of-band" characters associated with the high-transition set includes less than four logical transitions, and each of the out-of-band characters associated with the low-transition set of code characters includes more than four logical transitions. The difference in the number of transitions between in-band and out-of-band characters allows selected out-of-band characters to serve as control characters, and to be readily distinguished from in-band characters within the transmitted data stream." Shin, 9:49–10:8. "In accordance with one aspect of the invention, the synchronization characters associated with both the low-transition and high-transition sets of code characters are selected so as to facilitate rapid synchronization during data recovery. When the low-transition set of code characters is being

		employed, a special group of out-of-band characters is used during synchronization. Each special synchronization character includes a predefined number larger than four (e.g., 7) of logical transitions between binary character values, and a predefined number (e.g., two) of "non-transitions" between character values. As will be discussed below, the special synchronization characters are selected such that random logic may be used to distinguish each special synchronization character from the in-band characters of the low-transition set.” Shin, 10:17–34. “Referring to FIG. 6, the deserializer 34 receives the sequence of 10-bit characters T[9:0] produced by the encoder and generates 10-bit parallel received data upon bit lines RX9, RX8, . . . , RX0 (i.e., RX[9:0]). This 10-bit parallel received data is provided over bit lines RX[9:0] to the decoder 44, as well as to a decoder synchronization module 114. As is described below in section IV, the synchronization module 114 is operative to ascertain boundaries within the 10-bit parallel received data corresponding to the frame boundaries of the transmitted data (i.e., to T[9:0]). Specifically, synchronization module 114 determines upon which of the bit lines RX[9;0] the deserializer 34 is providing the received bits corresponding to the first bit T[0] of each transmitted byte T[9:0]. Upon making this determination, the synchronization module 114 provides a frame boundary pointer 118 to decoder 44 identifying the one of the bit lines RX[9:0] corresponding to the first bit T[0] of each transmitted 10-bit character T[9:0]. Upon receiving this synchronization information, the decoder 44 is disposed to decode the received data RX[9:0] in the following manner.” Shin, 13:37–57.  Shin, Fig. 6.

		Austin discloses and/or renders obvious this limitation. *See, e.g.*: "This system uses up to 5 pre-defined data packet sizes and up to 16 user defined data message types within each packet. Each packet has Forward Error Connection (FEC) and bit interleaving applied to it on transmission. Error checking is performed using a standard CRC. The system uses standard run in and byte synchronisation techniques. The byte synchronisation word used is 32 _bits long and there are 5_different words used, one to identify each of the different packet types." Austin, 4:37-45. To the extent that Plaintiff contends that the Additional Prior Art References do not explicitly or inherently disclose this element, or render obvious this element, this element is obvious in combination with one or more references and/or with the knowledge of a person of ordinary skill in the art, as set forth in Defendant's Invalidity Contentions.

11(b)	receiving a packet of symbols;	Chen discloses and/or renders obvious this limitation. *See, e.g.*: "FIG. 1 is a block diagram of a Fiber-to-the-Curb Digital Video Distribution Network in accordance with the present invention. The distribution network includes Video Information Providers (VIPs) shown generally at 100, a gateway, shown generally at 120, an access network, shown generally at 140, and the home of a Video Information User (VIU), shown generally at 160. The gateway 120 may comprise an Ll Gateway manufactured by Teleco. The VIPs may include a video server 102, an interactive data server 104, and an internet gateway 106, all of which communicate with an asynchronous transfer mode (ATM) network 122 via, for example, an OC-3 system, where optical fiber is used to transmit data at 155.52 Mbit/sec. The ATM network 122 may also receive data via a satellite receiving antenna 110, broadcast encoder and server 112, and ATM multiplexer 114. The ATM network 122 communicates with a session manager 123." Chen, 4:11–28.  FIG.1

		Chen, Fig. 1. "FIG. 4 is a block diagram of an Insertion Processing Unit (IPU) of a Digital Ad Insertion Module in accordance with the present invention. The IPU, shown generally at 400, receives the main stream at a packet framer 405, while the insertion stream is received at another packet framer 410. After processing by the packet framer 405, the main stream is provided to a Main Stream Parser (MSP) 415." Chen, 5:62–6:1.  FIG.4 Chen, Fig. 4. "The main stream parser 415 provides the main stream data to a main stream buffer 480, which communicates with a syntax processor 470. The main stream buffer 480 is part of the output buffer 475. The insertion stream parser 420 provides the insertion stream data to a Program Specific Information (PSI)/Program Identifier (PID) Replacer 435, which replaces the PSI tables and PIDs

		of the insertion stream with those of the main stream. The PSI/PID replacer 435 retrieves the relevant information from the main stream using a communication path which is not shown. The PSI and PID data of the main stream provide the information to connect the various audio, video, and data packets to a particular programming service. Specifically, this is accomplished using a Program Association Table (PAT) and a Program Map Table (PMT). The insertion stream is then provided to an Insertion Stream Buffer (ISB) 490, which is part of the output buffer 475. The ISB 490 communicates with a syntax processor 470. Furthermore, the output buffer 475 receives commands from a buffer controller 465 to provide an output stream with the insertion stream seamlessly spliced into the main stream." Chen, 6:33–54. "FIG. 5 illustrates the processing flow of the Insertion Processing Unit of FIG. 4 in accordance with the present invention. Note that the steps of FIG. 5 are shown as occurring in a sequential manner for clarity, but many of the steps may occur concurrently and/or sequentially. At block 502, the IPU outputs the main transport stream from the main stream buffer (MSB) 480. The IPU is essentially in a bypass mode at this time since the insertion stream has not yet been inserted. At block 506, the main stream parser parses the main stream to locate the splicing signal T-- in. Note that T-- in need not be carried in the main stream, but may be provided locally, for example, at a cable system headend. When T-- in is detected, the IPU continues to output the main transport stream from the main stream buffer. At block 508, the main stream parser parses the main stream to locate the closest packet with an anchor picture (e.g., I or P picture) start code and passes that packet to the syntax processor 470. This packet will be the last packet of the main stream. To ensure that the resulting data stream is MPEG compliant, no data from the next frame or sequence of frames may be included in this last main stream packet. At block 512, the syntax processor adds a sequence end code field to the last packet of the main stream. The sequence end code is a twenty-four bit code which indicates the end of a sequence of pictures. At block 514, the PSI/PID replacer 435 starts to replace the PSI/PID of the insertion stream." Chen, 12:28–53.

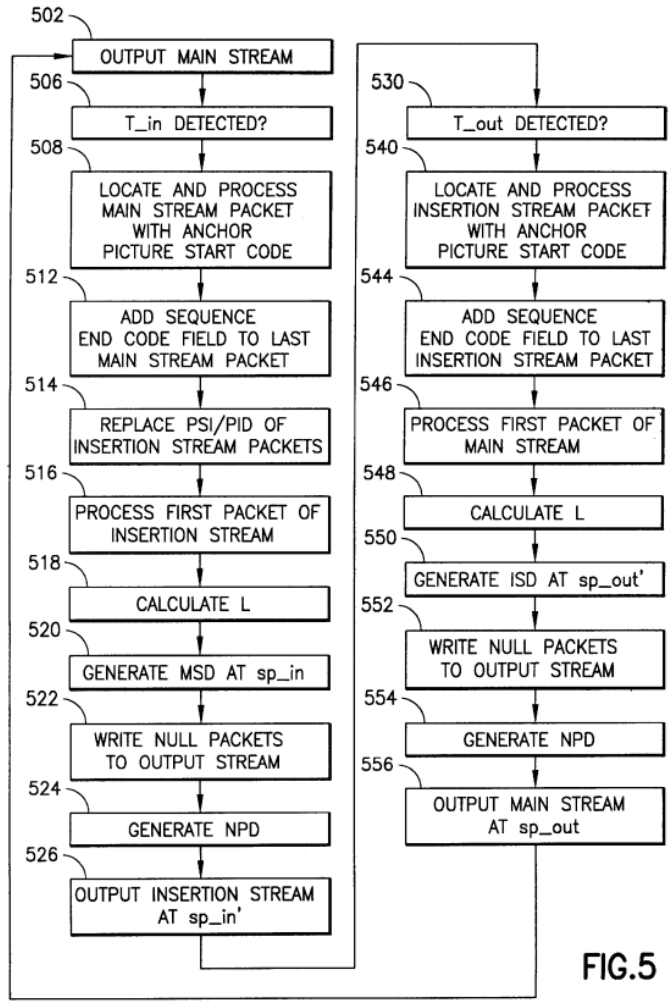
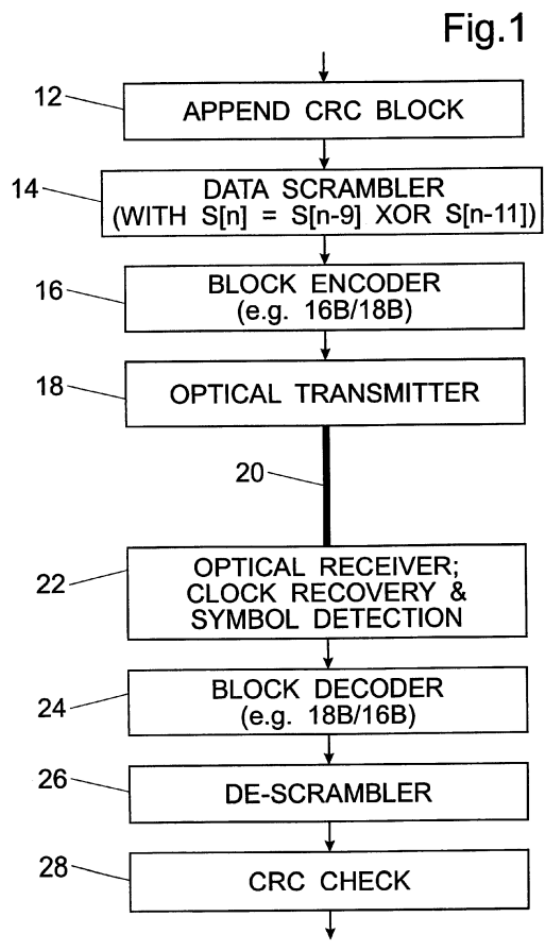


FIG.5

Chen, Fig. 5.

"In the MPEG-2 system (and the similar DigiCipher® II system proprietary to General Instrument Corporation, the assignee hereof) a transport stream, or transport multiplex is made up of a

		contiguous set of fixed length packets. Each packet is 188 total bytes in length, with the first four of those bytes being defined as the packet header. The payload portion of each packet is thus normally 184 bytes. However, a variable length adaptation field may be provided to extend the header, when required. When an adaptation field is present, the payload portion of the packet will be correspondingly shorter." Chen, 14:45–55. Cunningham discloses and/or renders obvious this limitation. *See, e.g.*: "Referring to FIG. 1, a data frame intended to be communicated over an optical fibre channel 20 typically comprises a binary digital message (payload) to be transferred between stations on the network; this message may comprise user data, or control information for managing operation of the network and its components. A circuit 12 generates an associated cyclic redundancy check (CRC) block containing check data derived from this message in known manner in accordance with a predetermined CRC algorithm, and appends the CRC block to the message." Cunningham, 2:65–3:7.



Cunningham, Fig. 1.

"The encoded blocks are transmitted over the optical fibre channel 20 (leftmost symbol first), and upon receipt at their destination they are detected by a receiver 22 where compensation is made in known manner for the effects of timing drift and other signal degradation to produce a restored digital data stream."

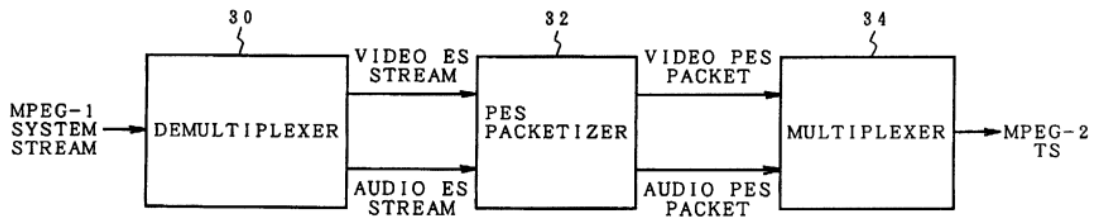
Cunningham, 4:45–50.

Hiroshima discloses and/or renders obvious this limitation. *See, e.g.*:

"FIG. 4 is a block diagram of the first embodiment of the invention in which the moving picture stream conversion in mode 1 is executed. A moving picture stream converting apparatus of FIG. 4 comprises a demultiplexer 30, a packetizer 32, and a multiplexer 34. The demultiplexer 30 receives an MPEG1 system stream as a conversion source and outputs elementary streams which were separated into video and audio streams. The packetizer 32 forms packets constructing a packetized elementary stream of MPEG2 with respect to each of the video and audio elementary streams which are supplied from the demultiplexer 30. The packet is usually called a PES packet. The multiplexer 34 divides each of the video and audio PES packets into a TS packet having a fixed length of 188 bytes and subsequently multiplexes, and outputs the resultant data as an MPEG2-TS."

Hiroshima, 7:59–8:7.

FIG. 4



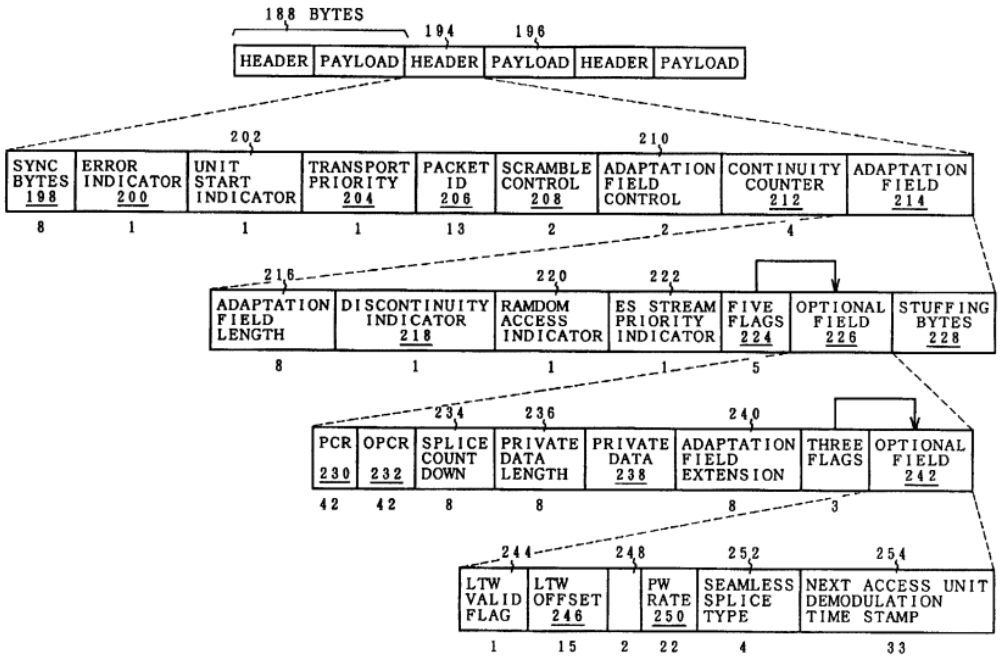
Hiroshima, Fig. 4.

"FIG. 11 shows the details of the MPEG2-TS. Each packet of the MPEG2-TS has a prefix length of 188 bytes. The header 194 consists of four bytes and the payload 196 consists of 184 bytes. The header 194 is constructed by sync bytes 198, an error indicator 200, a unit start indicator 202, a transport priority 204, a packet ID 206, a scramble control 208, an adaptation field control 210, a

continuity counter 212, and an adaptation field 214. The adaptation field 214 is constructed by an adaptation field length 216, a discontinuity indicator 218, a random access indicator 220, an elementary stream priority indicator 222, a 5-flag 224, an optional field 226, and stuffing bytes 228. Among them, the optional field 226 is constructed by a program clock reference 230, an original program clock reference field 232, a splice count down 234, a private data length 236, private data 238, an adaptation field extension 240, 3-flag 241, and an optional field 242."

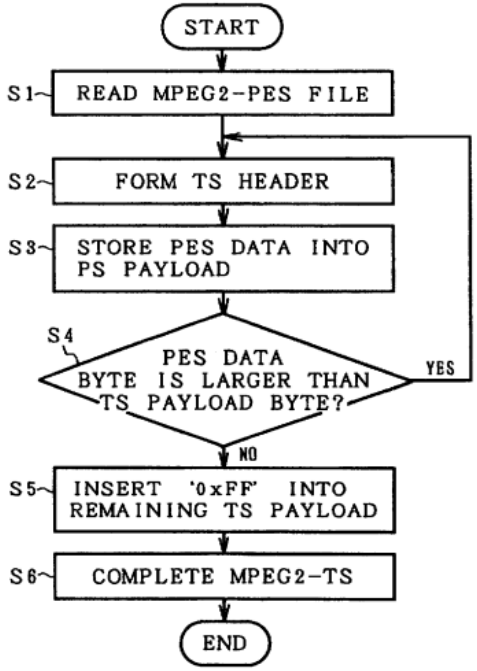
Hiroshima, 10:51–67.

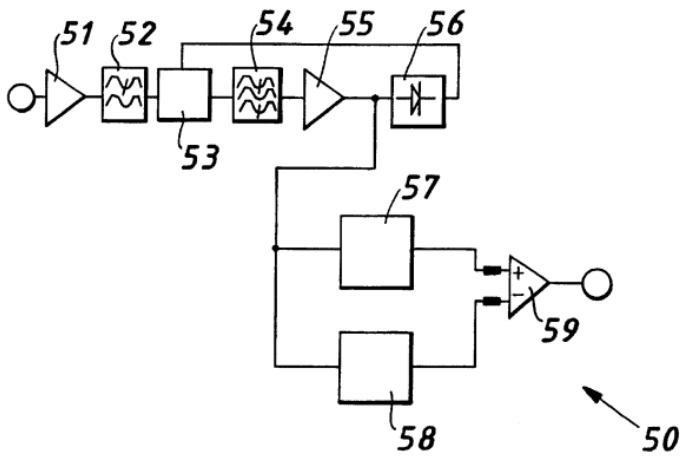
FIG. 11

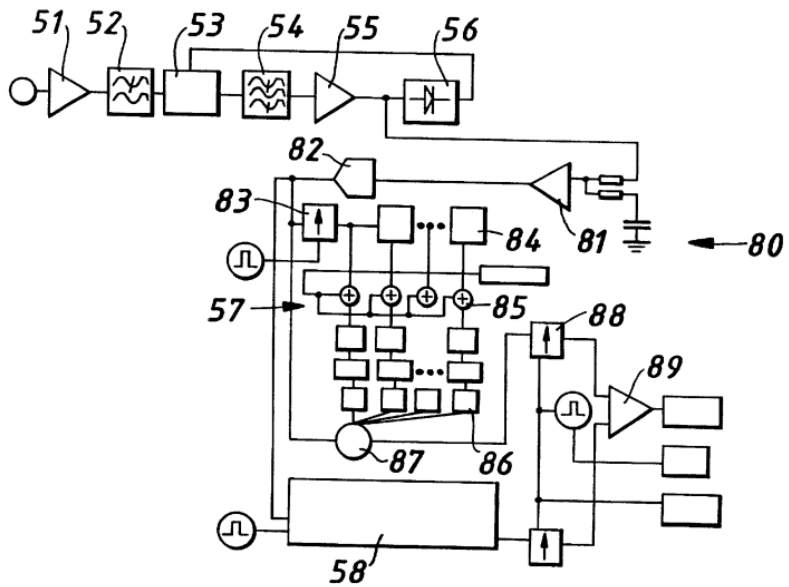


Hiroshima, Fig. 11.

"FIG. 15 shows a process for forming the MPEG2-TS from the PES packet 82 by a multiplexing process in the multiplexer 34 in FIG. 5. First in step S1, the MPEG2-PES file is read. The TS header 194 in FIG. 11 is formed in step S2. The TS header 194 is formed in a manner such that the

		parameters such as error indicator 200 indicative of the presence or absence of the bit error in the packet, unit start indicator 202 indicating whether a new PES packet appears in the TS payload or not, packet ID 206 indicating whether the packet in the payload is video or audio, adaptation field 214, and adaptation field control 210 indicative of the presence or absence of the payload 196 are set in accordance with the state of the payload storage data of the 184-byte unit divided from the MPEG2-PES packet which had been read in step S1." Hiroshima, 11:13–27. FIG. 15  Hiroshima, Fig. 15.

		Austin discloses and/or renders obvious this limitation. *See, e.g.*: "A suitable receiver is shown in FIG. 4. The receiver 50 is a non-coherent FFSK matched filter receiver. The receiver 50 includes amplifier 51, low pass filter 52 and automatic gain control 53 and through the band pass filter 54. The filter output is passed via amplifier 55 to detector 56. This allows the analog signal from the pipe to be received and the digital data recovered into streams F1 and F2. Matched filters 57 and 58 and summer 59 provide the recovered (uncloacked) data at its output. Separate matched filter paths are used for each tone and data decisions are made by comparison of the resultant energy output from each filter. Symbol recovery is derived from the data decision stream. Each matched filter is implemented as an integration of the incoming data stream multiplied by a sine wave at the respective sub-carrier frequency (over each successive study symbol period)." Austin, 2:57-3:5. FIG. 4.  Austin, Fig. 4.

		"When digital implementation is required a receiver can be constructed as shown in FIG. 7. This uses matched filters and employs a series of correlators. The analog first stage output is received by summing amplifier 81 and converted into digital form via analog to digital converter 82. The dual path filtering 57 and 58 is provided as before now in digital form. As shown in the upper (expanded) path a sampler 83 with delays 84 and adders 85, dividers 86 and summer 87 to path sampler 88 and comparator 89. Hence, the received analog signal is amplified, filtered and fed to an automatic gain control loop. Typically the LPF has a −3 dB cut off at 20 HZ. The AGC uses a bandpass filter with −3 dB points at 7.5 HZ and 17.5 HZ to select the band of interest for detection. The level detector input is used to adjust the gain of the loop so that the output AC peak to peak signal most closely fits the peak to peak full scale (FS) limit of the digital convertor (ADC). The ADC gives typically 10 bit resolution. F1 and F2 paths are concerned respectively with each matched filter." Austin, 3:25-43. FIG. 7. 

		Austin, Fig. 7. "This system uses up to 5 pre-defined data packet sizes and up to 16 user defined data message types within each packet. Each packet has Forward Error Connection (FEC) and bit interleaving applied to it on transmission. Error checking is performed using a standard CRC. The system uses standard run in and byte synchronisation techniques. The byte synchronisation word used is 32 _bits long and there are 5_different words used, one to identify each of the different packet types." Austin, 4:37-45. To the extent that Plaintiff contends that the Additional Prior Art References do not explicitly or inherently disclose this element, or render obvious this element, this element is obvious in combination with one or more references and/or with the knowledge of a person of ordinary skill in the art, as set forth in Defendant's Invalidity Contentions.
11(c)	indicating that the received packet of symbols has the packet type of the received synchronization symbol; and	Chen discloses and/or renders obvious this limitation. *See, e.g.*: "FIG. 4 is a block diagram of an Insertion Processing Unit (IPU) of a Digital Ad Insertion Module in accordance with the present invention. The IPU, shown generally at 400, receives the main stream at a packet framer 405, while the insertion stream is received at another packet framer 410. After processing by the packet framer 405, the main stream is provided to a Main Stream Parser (MSP) 415." Chen, 5:62–6:1.

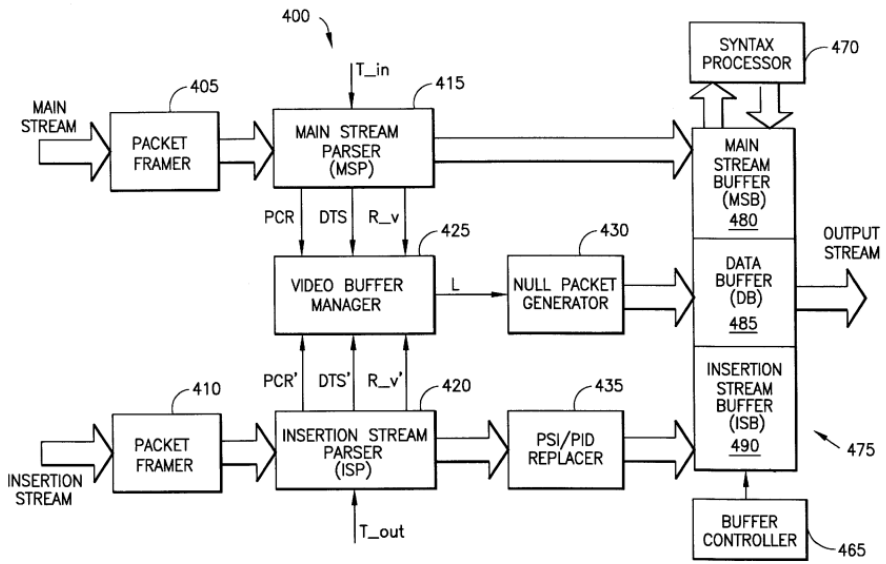


FIG.4

Chen, Fig. 4.

"The main stream parser 415 provides the main stream data to a main stream buffer 480, which communicates with a syntax processor 470. The main stream buffer 480 is part of the output buffer 475. The insertion stream parser 420 provides the insertion stream data to a Program Specific Information (PSI)/Program Identifier (PID) Replacer 435, which replaces the PSI tables and PIDs of the insertion stream with those of the main stream. The PSI/PID replacer 435 retrieves the relevant information from the main stream using a communication path which is not shown. The PSI and PID data of the main stream provide the information to connect the various audio, video, and data packets to a particular programming service. Specifically, this is accomplished using a Program Association Table (PAT) and a Program Map Table (PMT). The insertion stream is then provided to an Insertion Stream Buffer (ISB) 490, which is part of the output buffer 475. The ISB 490 communicates with a syntax processor 470. Furthermore, the output buffer 475 receives commands from a buffer controller 465 to provide an output stream with the insertion stream seamlessly spliced into the main stream."

		Chen, 6:33–54. "FIG. 5 illustrates the processing flow of the Insertion Processing Unit of FIG. 4 in accordance with the present invention. Note that the steps of FIG. 5 are shown as occurring in a sequential manner for clarity, but many of the steps may occur concurrently and/or sequentially. At block 502, the IPU outputs the main transport stream from the main stream buffer (MSB) 480. The IPU is essentially in a bypass mode at this time since the insertion stream has not yet been inserted. At block 506, the main stream parser parses the main stream to locate the splicing signal T-- in. Note that T-- in need not be carried in the main stream, but may be provided locally, for example, at a cable system headend. When T-- in is detected, the IPU continues to output the main transport stream from the main stream buffer. At block 508, the main stream parser parses the main stream to locate the closest packet with an anchor picture (e.g., I or P picture) start code and passes that packet to the syntax processor 470. This packet will be the last packet of the main stream. To ensure that the resulting data stream is MPEG compliant, no data from the next frame or sequence of frames may be included in this last main stream packet. At block 512, the syntax processor adds a sequence end code field to the last packet of the main stream. The sequence end code is a twenty-four bit code which indicates the end of a sequence of pictures. At block 514, the PSI/PID replacer 435 starts to replace the PSI/PID of the insertion stream." Chen, 12:28–53.

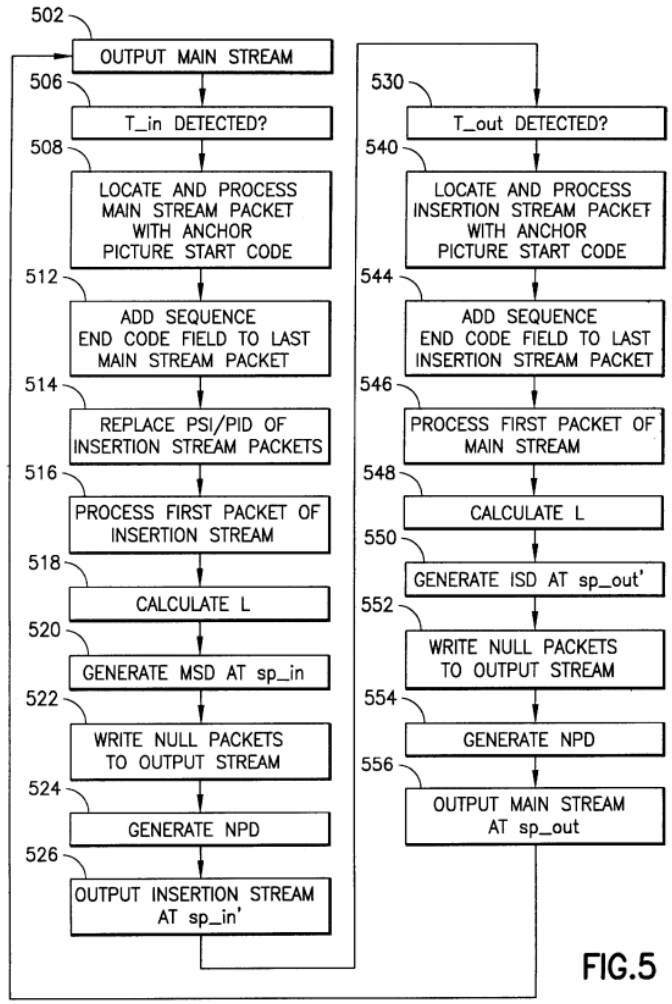
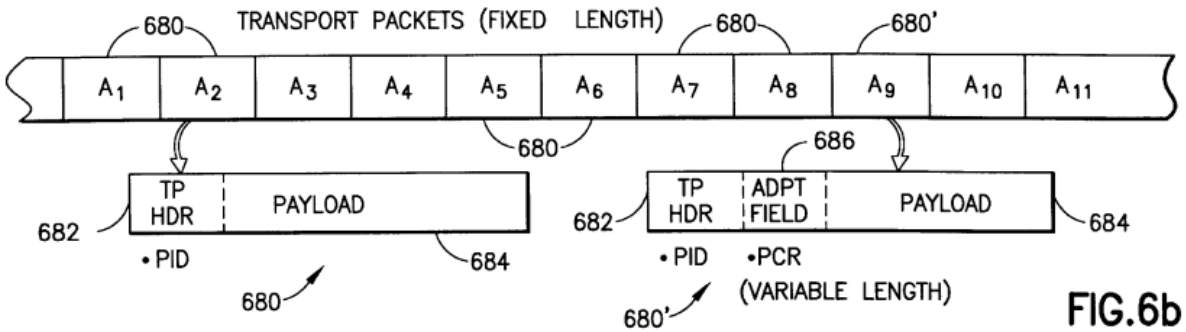


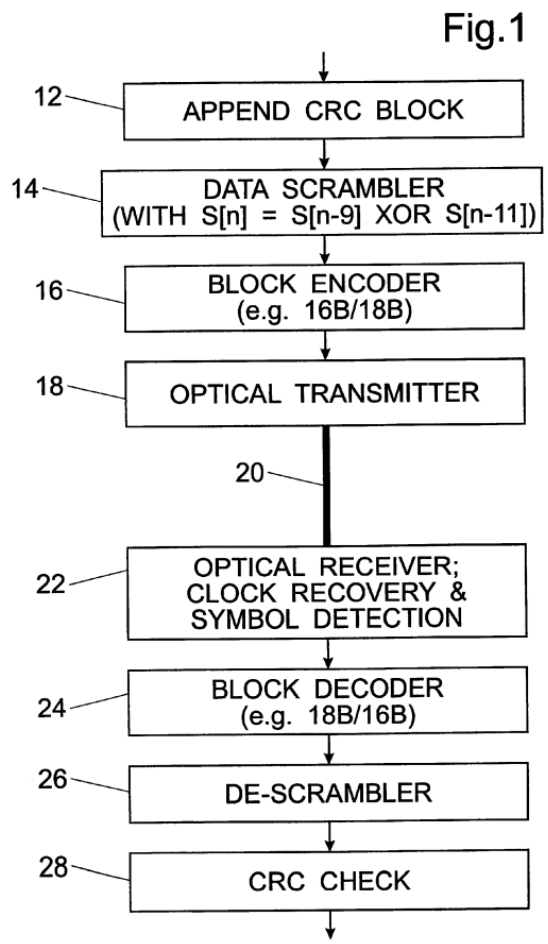
FIG.5

Chen, Fig. 5.

"In the MPEG-2 system (and the similar DigiCipher® II system proprietary to General Instrument Corporation, the assignee hereof) a transport stream, or transport multiplex is made up of a

		contiguous set of fixed length packets. Each packet is 188 total bytes in length, with the first four of those bytes being defined as the packet header. The payload portion of each packet is thus normally 184 bytes. However, a variable length adaptation field may be provided to extend the header, when required. When an adaptation field is present, the payload portion of the packet will be correspondingly shorter. Various timing and identification information is provided in different portions of the transport stream. These include a packet identifier (PID) found in the transport header of each transport packet to provide a reference number for identifying the transport packets carrying a specific service component. This number is included in a service definition or "service map" used by the receiver to identify those transport packets required to reconstruct a television program signal. The PID may also be referenced for various grooming and remultiplexing functions. In the case of video, audio or isochronous control data, the stream of packets labeled with a single PID represents a single video, audio or isochronous data service elementary stream, respectively. Each type of packet will have a different PID identifying the packet type. Timing information carried by the transport stream includes a program clock reference (PCR) which effectively represents a sample of the system time clock (STC) time base that underlies the service composed of the PIDs referenced in the service map. The PID carrying the packet with the PCR is also referenced in the service map. The video, audio and isochronous data components of a service are locked through a defined relationship to the system time clock. The PCR serves to define the transport rate, in the sense that between any two successive PCRs in one PID, the transport rate is constant and nominally equal to the system time clock rate times the ratio of the total number of transport bytes between the PCRs divided by the difference in the PCRs in units of system time clock ticks." Chen, 14:45–15:17. "As indicated in FIG. 6b, the transport header of each transport packet includes the PID which identifies the particular service component carried by the transport packet. The PCR will be carried by an adaptation field to provide timing information for a desired service. At a receiver, the PCR for the desired service is detected from the adaptation field. The PIDs of the transport packets will

		then be monitored in accordance with the timing established by the PCR to recover those transport packets carrying a particular component of the service to be processed." Chen, 16:41–50.  Chen, Fig. 6b. Cunningham discloses and/or renders obvious this limitation. *See, e.g.*: "Referring to FIG. 1, a data frame intended to be communicated over an optical fibre channel 20 typically comprises a binary digital message (payload) to be transferred between stations on the network; this message may comprise user data, or control information for managing operation of the network and its components. A circuit 12 generates an associated cyclic redundancy check (CRC) block containing check data derived from this message in known manner in accordance with a predetermined CRC algorithm, and appends the CRC block to the message." Cunningham, 2:65–3:7.



Cunningham, Fig. 1.

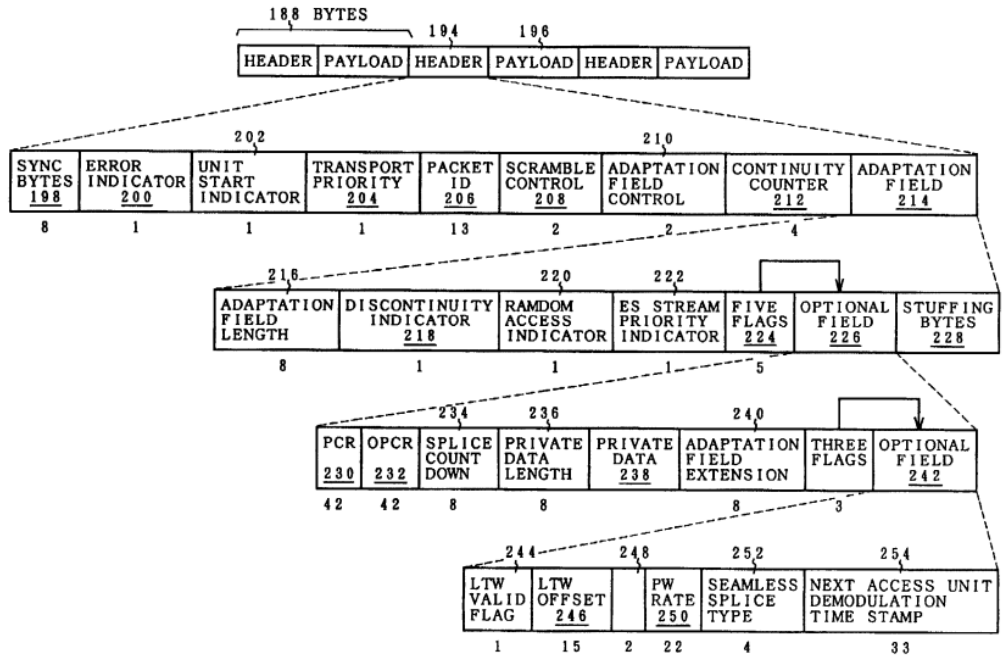
"Control information is conveyed by using selected n-bit data block values, together with a distinctive value of 10 for the pair of label bits a and b to indicate that the associated data block represents control information rather than actual data. The data block values used for this purpose are selected to be values which inherently have a disparity of zero, such as (for 8-bit blocks)

		00001111 or 10101010. It will be noted that the label bit pair value 10 likewise has zero disparity, so the complete encoded block also has zero disparity." Cunningham, 4:23–31. "The encoded blocks are transmitted over the optical fibre channel 20 (leftmost symbol first), and upon receipt at their destination they are detected by a receiver 22 where compensation is made in known manner for the effects of timing drift and other signal degradation to produce a restored digital data stream." Cunningham, 4:45–50. "There are many potential ways to achieve code block synchronization (i.e. correct alignment in the receiver of the label and payload fields). For example, one technique would be to transmit a 'comma sequence' during start-up. As an example of this technique consider a code with n=8 using the simple control payload scheme. A training control state of 1111 1111 would be transmitted as: [10 1111 1111 0000 0000][10 1111 1111 0000 0000][10 1111 1111 0000 0000] The receiver searches for the bit pair 10. This is found in only two locations: the correct location (the label field), and at the end of the run of eight 1's. However, if the receiver (incorrectly) identifies the label field to be the 10 bit pair at the end of the run of 1's, the following 16 bits are not a valid control payload (first 8 bits are not the complement of the second 8 bits). The receiver can therefore detect the mistake. It is also noted that this sequence contains a regular 010 pattern, which aids in phase-locked loop (PLL) clock acquisition." Cunningham, 7:5–23. Hiroshima discloses and/or renders obvious this limitation. *See, e.g.*: "FIG. 11 shows the details of the MPEG2-TS. Each packet of the MPEG2-TS has a prefix length of 188 bytes. The header 194 consists of four bytes and the payload 196 consists of 184 bytes. The header 194 is constructed by sync bytes 198, an error indicator 200, a unit start indicator 202, a transport priority 204, a packet ID 206, a scramble control 208, an adaptation field control 210, a

continuity counter 212, and an adaptation field 214. The adaptation field 214 is constructed by an adaptation field length 216, a discontinuity indicator 218, a random access indicator 220, an elementary stream priority indicator 222, a 5-flag 224, an optional field 226, and stuffing bytes 228. Among them, the optional field 226 is constructed by a program clock reference 230, an original program clock reference field 232, a splice count down 234, a private data length 236, private data 238, an adaptation field extension 240, 3-flag 241, and an optional field 242."

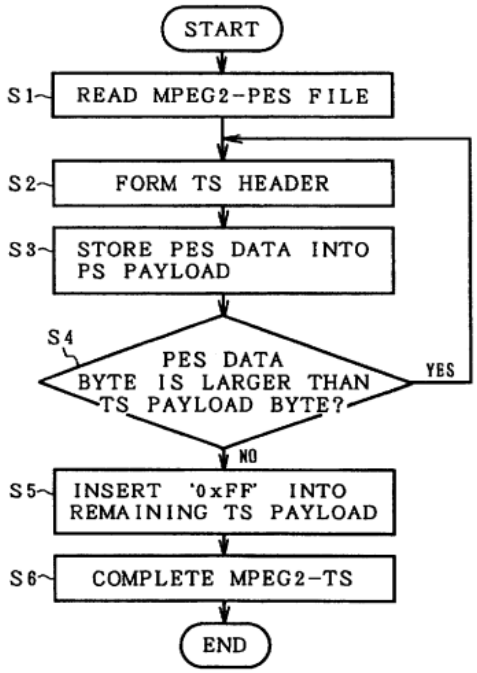
Hiroshima, 10:51–67.

FIG. 11



Hiroshima, Fig. 11.

"FIG. 15 shows a process for forming the MPEG2-TS from the PES packet 82 by a multiplexing process in the multiplexer 34 in FIG. 5. First in step S1, the MPEG2-PES file is read. The TS header 194 in FIG. 11 is formed in step S2. The TS header 194 is formed in a manner such that the

		parameters such as error indicator 200 indicative of the presence or absence of the bit error in the packet, unit start indicator 202 indicating whether a new PES packet appears in the TS payload or not, packet ID 206 indicating whether the packet in the payload is video or audio, adaptation field 214, and adaptation field control 210 indicative of the presence or absence of the payload 196 are set in accordance with the state of the payload storage data of the 184-byte unit divided from the MPEG2-PES packet which had been read in step S1." Hiroshima, 11:13–27. FIG. 15  Hiroshima, Fig. 15.

		Austin discloses and/or renders obvious this limitation. *See, e.g.*: "This system uses up to 5 pre-defined data packet sizes and up to 16 user defined data message types within each packet. Each packet has Forward Error Connection (FEC) and bit interleaving applied to it on transmission. Error checking is performed using a standard CRC. The system uses standard run in and byte synchronisation techniques. The byte synchronisation word used is 32 _bits long and there are 5_different words used, one to identify each of the different packet types." Austin, 4:37-45. To the extent that Plaintiff contends that the Additional Prior Art References do not explicitly or inherently disclose this element, or render obvious this element, this element is obvious in combination with one or more references and/or with the knowledge of a person of ordinary skill in the art, as set forth in Defendant's Invalidity Contentions.
11(d)	the synchronization symbol permitting properly alignment upon receiving with a synchronization primitive to be correctly aligned on a symbol boundary.	Cunningham discloses and/or renders obvious this limitation. *See, e.g.*: "There are many potential ways to achieve code block synchronization (i.e. correct alignment in the receiver of the label and payload fields). For example, one technique would be to transmit a 'comma sequence' during start-up. As an example of this technique consider a code with n=8 using the simple control payload scheme. A training control state of 1111 1111 would be transmitted as: [10 1111 1111 0000 0000][10 1111 1111 0000 0000][10 1111 1111 0000 0000]" Cunningham, 7:5–13. Haartsen discloses and/or renders obvious this limitation. *See, e.g.*: "The present invention fulfills the above-described and other needs by providing a multi-part digital preamble for use in transmitting digital data packets. Advantageously, a preamble according to the invention substantially reduces the overall length of DC-free sequences that are transmitted with each data packet in a digital FM system. According to exemplary embodiments, a multi-part preamble includes a short, substantially DC-free leading part which enables coarse estimation of a

		prevailing DC level at an FM receiver. The exemplary preamble also includes a synchronization part which is not necessarily substantially DC-free and which carries timing and/or other useful information (e.g. , channel identification, destination address, etc.) that can vary from data packet to data packet (e.g. , depending on sender, recipient, time of day, etc.).” Haartsen, 5:2–15. “To alleviate this problem, the present invention discloses that a digital preamble can be strategically divided into three or more parts so that synchronization and other information can be included within the preamble while still enabling accurate DC estimation. Specifically, a short leading DC-free part of the preamble (e.g., alternating logical l 's and 0's, or any other sequence including an equal number of logical l's and logical 0's) can be used to provide initial, coarse DC offset estimation. Thereafter, a synchronization part of the preamble, which is not necessarily DC free, but which is protected by code to prevent detection errors, can be used not only to synchronize detection of a data packet, but also to carry timing and/or other useful information (e.g., channel identification, destination address, time of day, user identification, etc.).” Haartsen, 12:13–13:2. “An exemplary multi-part preamble according to the invention is depicted in Figure 5. In the figure, a three-part preamble 500 is shown to include a DC-free leading sequence 510, followed by a multiple-bit synchronization word 520 which is not necessarily DC free and which can be different for each data packet. The synchronization word 520, which can include timing and/or other information and which is coded to protect against detection errors, is in turn followed by a trailing DC-free sequence 530. Although the leading and trailing DC-free sequences 510, 530 are shown in the figure as 4-bit sequences of alternating logical l 's and 0's, those of ordinary skill will appreciate that each DC-free sequence 510, 530 can, in practice, be shorter or longer in length and can include any DC-free sequence having an equal number of logical l's and logical 0's.” Haartsen, 13:13–14:2.

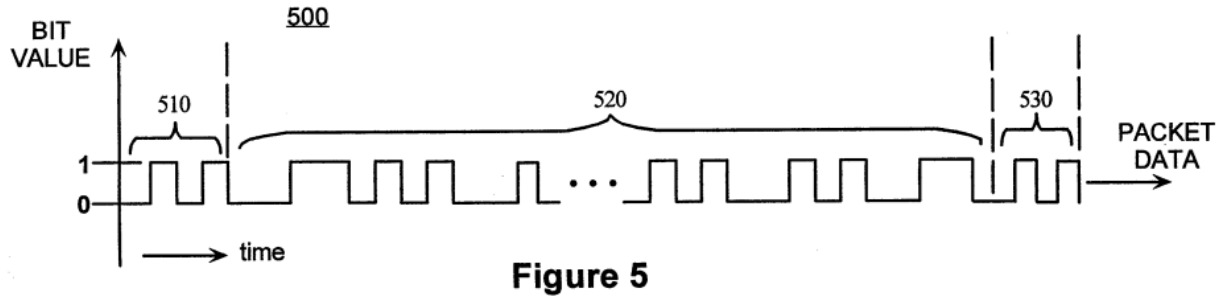


Figure 5

Haartsen, Fig. 5.

"When the exemplary preamble 500 is received by the system 300 of Figure 3, the leading DC-free sequence 510 induces a coarse estimate of the baseband DC offset on the storage capacitor 340. Although the estimate is coarse, it is sufficient to permit successful detection of the coded synchronization word 520 (e.g., by the correlation processor 310). Methods for coding of a synchronization word to protect against erroneous detection in the presence of some DC-offset are known to those of skill in the art. Once the correlation processor 310 has detected the synchronization word 520, timing between received bits is well known. Consequently, the correlation processor 310 can open the switch 330 precisely at the end of the trailing DC-free sequence 530 to thereby store an accurate estimate of the DC offset for reception and detection of the incoming digital data packet."

Haartsen, 14:3–14.

Shin discloses and/or renders obvious this limitation. *See, e.g.*:

"In accordance with one aspect of the invention, the synchronization characters associated with both the low-transition and high-transition sets of code characters are selected so as to facilitate rapid synchronization during data recovery. When the low-transition set of code characters is being employed, a special group of out-of-band characters is used during synchronization. Each special synchronization character includes a predefined number larger than four (e.g., 7) of logical transitions between binary character values, and a predefined number (e.g., two) of "non-transitions" between character values. As will be discussed below, the special synchronization

characters are selected such that random logic may be used to distinguish each special synchronization character from the in-band characters of the low-transition set."

Shin, 10:17–34.

"Referring to FIG. 6, the deserializer 34 receives the sequence of 10-bit characters T[9:0] produced by the encoder and generates 10-bit parallel received data upon bit lines RX9, RX8, . . . , RX0 (i.e., RX[9:0]). This 10-bit parallel received data is provided over bit lines RX[9:0] to the decoder 44, as well as to a decoder synchronization module 114. As is described below in section IV, the synchronization module 114 is operative to ascertain boundaries within the 10-bit parallel received data corresponding to the frame boundaries of the transmitted data (i.e., to T[9:0]). Specifically, synchronization module 114 determines upon which of the bit lines RX[9;0] the deserializer 34 is providing the received bits corresponding to the first bit T[0] of each transmitted byte T[9:0]. Upon making this determination, the synchronization module 114 provides a frame boundary pointer 118 to decoder 44 identifying the one of the bit lines RX[9:0] corresponding to the first bit T[0] of each transmitted 10-bit character T[9:0]. Upon receiving this synchronization information, the decoder 44 is disposed to decode the received data RX[9:0] in the following manner."

Shin, 13:37–57.

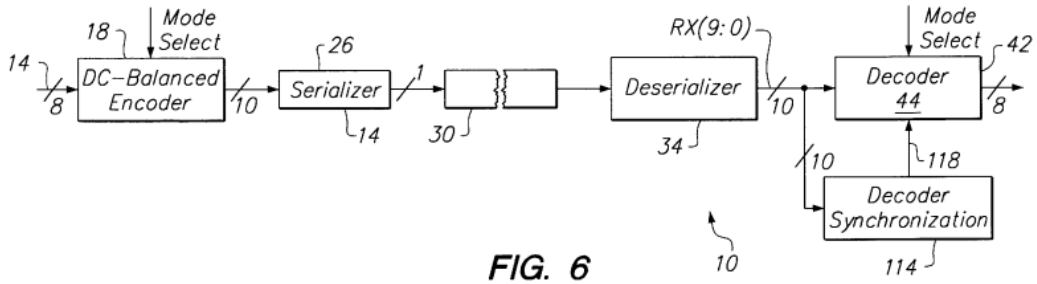


FIG. 6

Shin, Fig. 6.

To the extent that Plaintiff contends that the Additional Prior Art References do not explicitly or inherently disclose this element, or render obvious this element, this element is obvious in

		combination with one or more references and/or with the knowledge of a person of ordinary skill in the art, as set forth in Defendant's Invalidity Contentions.
Claim 19		
19[p]	A communications device for transmitting packet types of packets, comprising:	To the extent the preamble is limiting, Chen discloses this element and/or renders it obvious either alone or in combination with other references and/or the knowledge of one of ordinary skill in the art. *See, e.g.*: "FIG. 1 is a block diagram of a Fiber-to-the-Curb Digital Video Distribution Network in accordance with the present invention. The distribution network includes Video Information Providers (VIPs) shown generally at 100, a gateway, shown generally at 120, an access network, shown generally at 140, and the home of a Video Information User (VIU), shown generally at 160. The gateway 120 may comprise an Ll Gateway manufactured by Teleco. The VIPs may include a video server 102, an interactive data server 104, and an internet gateway 106, all of which communicate with an asynchronous transfer mode (ATM) network 122 via, for example, an OC-3 system, where optical fiber is used to transmit data at 155.52 Mbit/sec. The ATM network 122 may also receive data via a satellite receiving antenna 110, broadcast encoder and server 112, and ATM multiplexer 114. The ATM network 122 communicates with a session manager 123." Chen, 4:11–28.

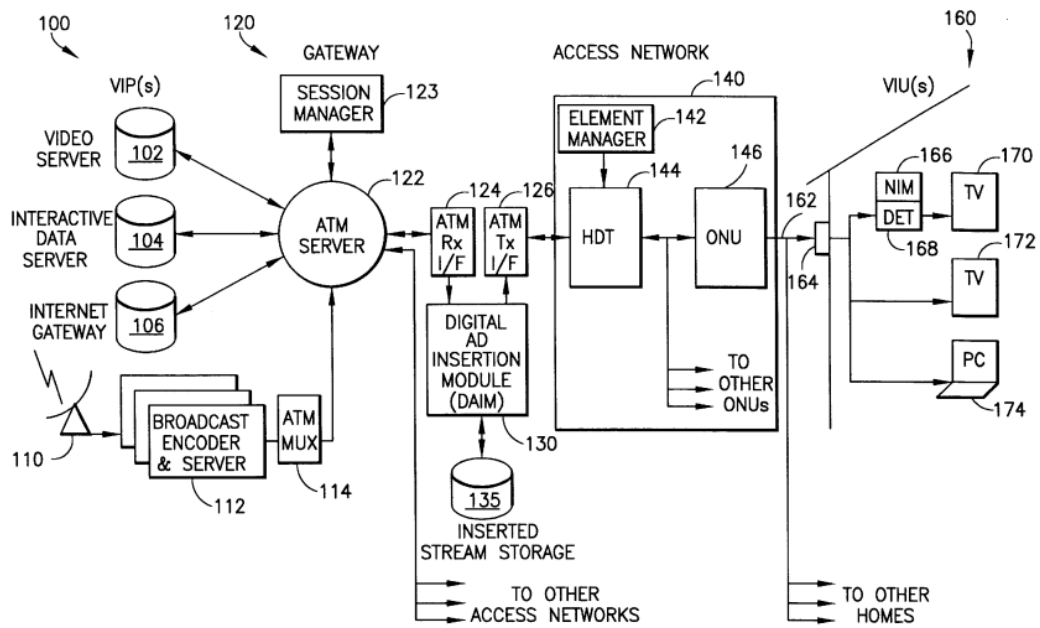


FIG.1

Chen, Fig. 1.

"FIG. 4 is a block diagram of an Insertion Processing Unit (IPU) of a Digital Ad Insertion Module in accordance with the present invention. The IPU, shown generally at 400, receives the main stream at a packet framer 405, while the insertion stream is received at another packet framer 410. After processing by the packet framer 405, the main stream is provided to a Main Stream Parser (MSP) 415."

Chen, 5:62–6:1.

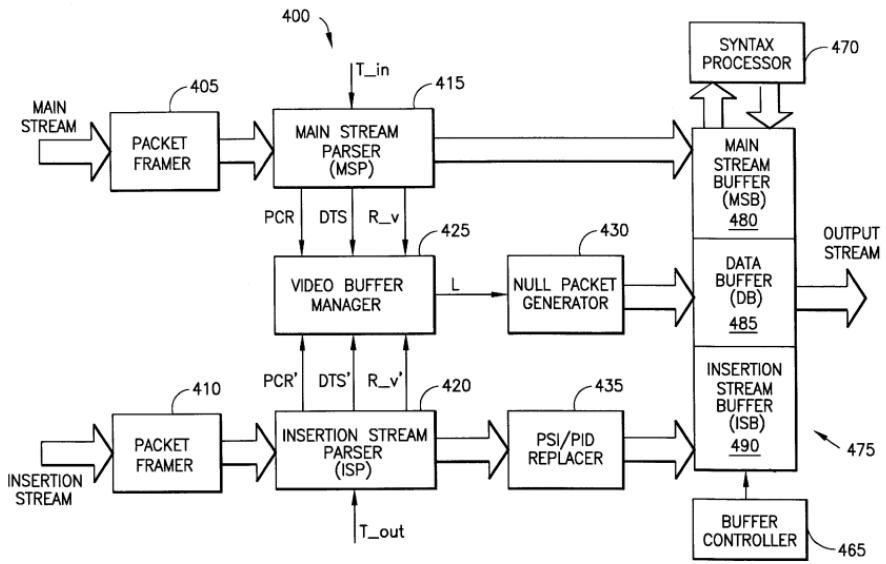


FIG.4

Chen, Fig. 4.

"The main stream parser 415 provides the main stream data to a main stream buffer 480, which communicates with a syntax processor 470. The main stream buffer 480 is part of the output buffer 475. The insertion stream parser 420 provides the insertion stream data to a Program Specific Information (PSI)/Program Identifier (PID) Replacer 435, which replaces the PSI tables and PIDs of the insertion stream with those of the main stream. The PSI/PID replacer 435 retrieves the relevant information from the main stream using a communication path which is not shown. The PSI and PID data of the main stream provide the information to connect the various audio, video, and data packets to a particular programming service. Specifically, this is accomplished using a Program Association Table (PAT) and a Program Map Table (PMT). The insertion stream is then provided to an Insertion Stream Buffer (ISB) 490, which is part of the output buffer 475. The ISB 490 communicates with a syntax processor 470. Furthermore, the output buffer 475 receives commands from a buffer controller 465 to provide an output stream with the insertion stream seamlessly spliced into the main stream."

		Chen, 6:33–54. "FIG. 5 illustrates the processing flow of the Insertion Processing Unit of FIG. 4 in accordance with the present invention. Note that the steps of FIG. 5 are shown as occurring in a sequential manner for clarity, but many of the steps may occur concurrently and/or sequentially. At block 502, the IPU outputs the main transport stream from the main stream buffer (MSB) 480. The IPU is essentially in a bypass mode at this time since the insertion stream has not yet been inserted. At block 506, the main stream parser parses the main stream to locate the splicing signal T-- in. Note that T-- in need not be carried in the main stream, but may be provided locally, for example, at a cable system headend. When T-- in is detected, the IPU continues to output the main transport stream from the main stream buffer. At block 508, the main stream parser parses the main stream to locate the closest packet with an anchor picture (e.g., I or P picture) start code and passes that packet to the syntax processor 470. This packet will be the last packet of the main stream. To ensure that the resulting data stream is MPEG compliant, no data from the next frame or sequence of frames may be included in this last main stream packet. At block 512, the syntax processor adds a sequence end code field to the last packet of the main stream. The sequence end code is a twenty-four bit code which indicates the end of a sequence of pictures. At block 514, the PSI/PID replacer 435 starts to replace the PSI/PID of the insertion stream." Chen, 12:28–53.

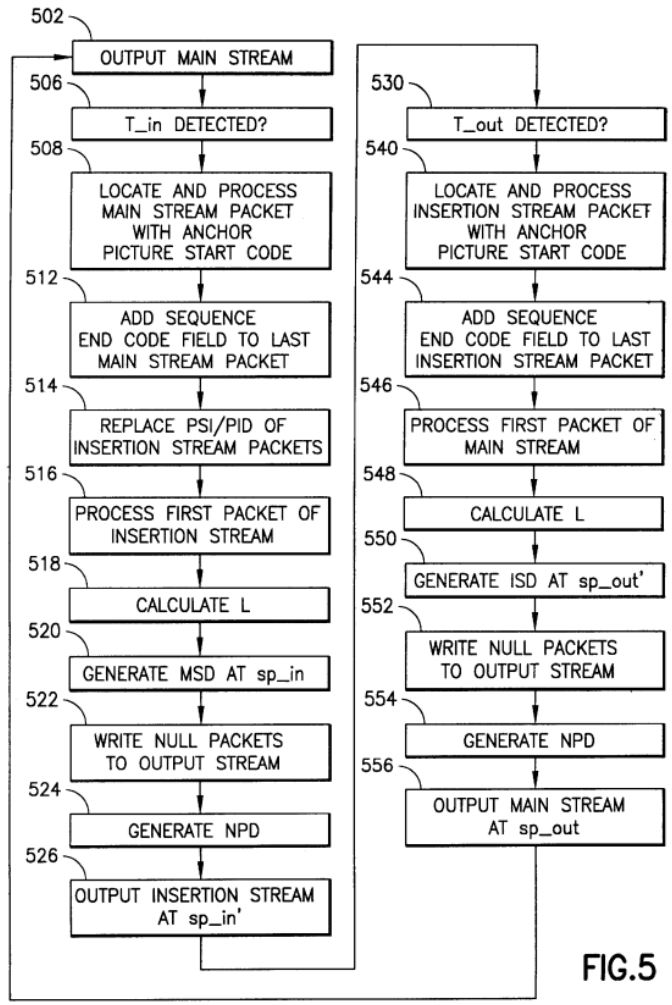
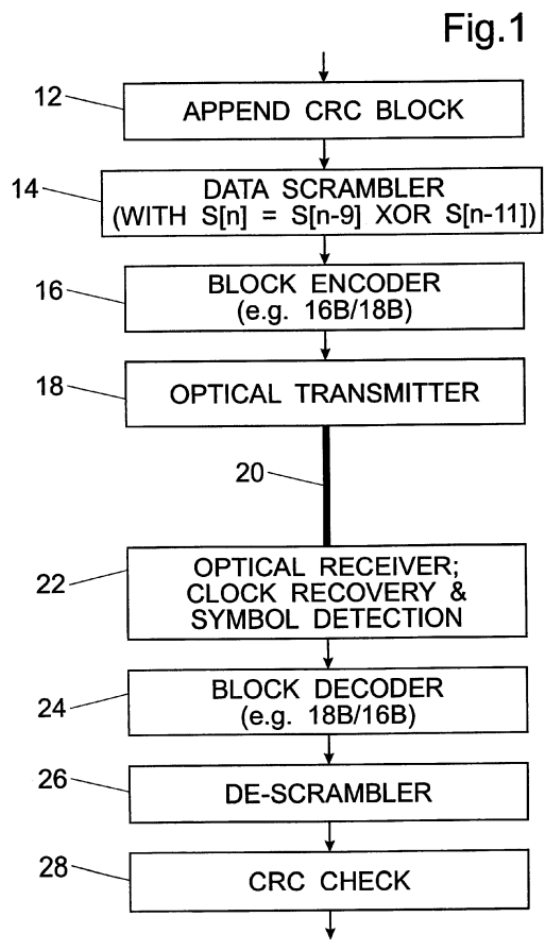


FIG.5

Chen, Fig. 5.

"In the MPEG-2 system (and the similar DigiCipher® II system proprietary to General Instrument Corporation, the assignee hereof) a transport stream, or transport multiplex is made up of a

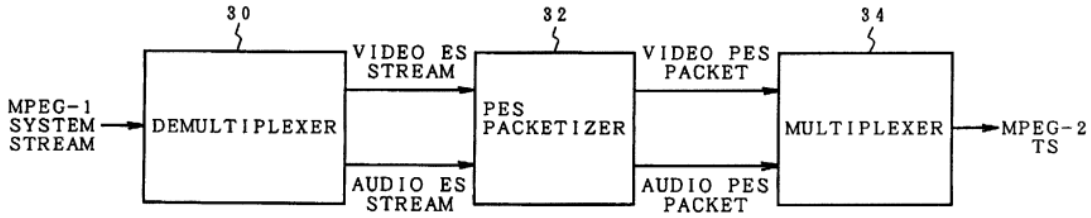
		contiguous set of fixed length packets. Each packet is 188 total bytes in length, with the first four of those bytes being defined as the packet header. The payload portion of each packet is thus normally 184 bytes. However, a variable length adaptation field may be provided to extend the header, when required. When an adaptation field is present, the payload portion of the packet will be correspondingly shorter. Various timing and identification information is provided in different portions of the transport stream. These include a packet identifier (PID) found in the transport header of each transport packet to provide a reference number for identifying the transport packets carrying a specific service component. This number is included in a service definition or "service map" used by the receiver to identify those transport packets required to reconstruct a television program signal. The PID may also be referenced for various grooming and remultiplexing functions. In the case of video, audio or isochronous control data, the stream of packets labeled with a single PID represents a single video, audio or isochronous data service elementary stream, respectively. Each type of packet will have a different PID identifying the packet type." Chen, 14:45–15:3. To the extent the preamble is limiting, Cunningham discloses this element and/or renders it obvious either alone or in combination with other references and/or the knowledge of one of ordinary skill in the art. *See, e.g.*: "Referring to FIG. 1, a data frame intended to be communicated over an optical fibre channel 20 typically comprises a binary digital message (payload) to be transferred between stations on the network; this message may comprise user data, or control information for managing operation of the network and its components. A circuit 12 generates an associated cyclic redundancy check (CRC) block containing check data derived from this message in known manner in accordance with a predetermined CRC algorithm, and appends the CRC block to the message." Cunningham, 2:65–3:7.

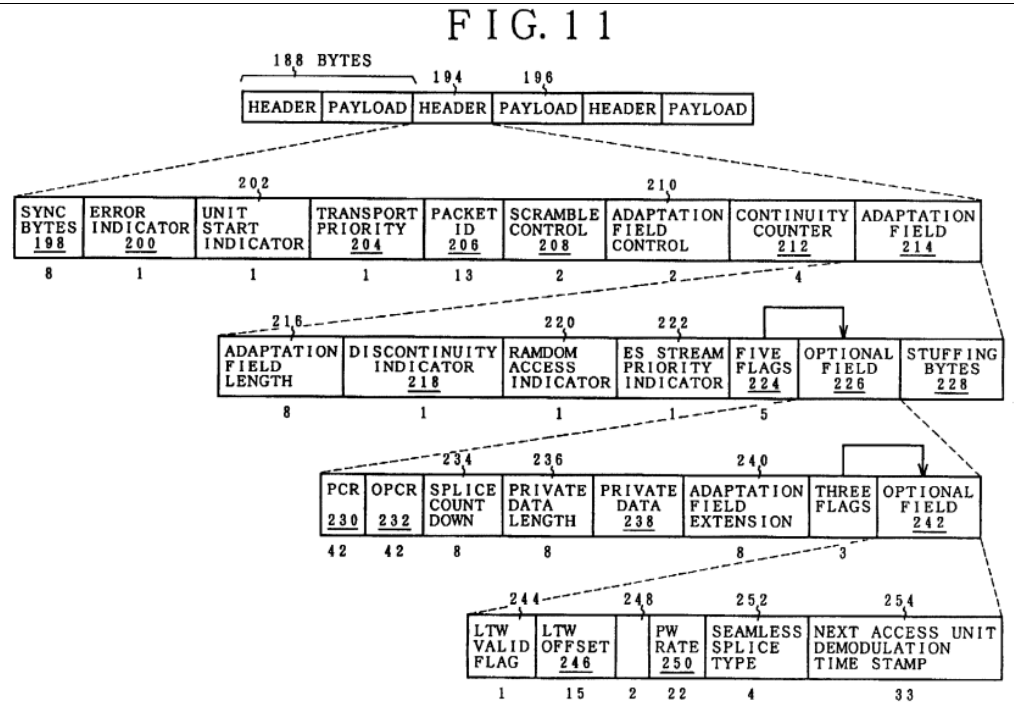


Cunningham, Fig. 1.

"Control information is conveyed by using selected n-bit data block values, together with a distinctive value of 10 for the pair of label bits a and b to indicate that the associated data block represents control information rather than actual data. The data block values used for this purpose are selected to be values which inherently have a disparity of zero, such as (for 8-bit blocks)

		00001111 or 10101010. It will be noted that the label bit pair value 10 likewise has zero disparity, so the complete encoded block also has zero disparity." Cunningham, 4:23–31. "The encoded blocks are transmitted over the optical fibre channel 20 (leftmost symbol first), and upon receipt at their destination they are detected by a receiver 22 where compensation is made in known manner for the effects of timing drift and other signal degradation to produce a restored digital data stream." Cunningham, 4:45–50. To the extent the preamble is limiting, Hiroshima discloses this element and/or renders it obvious either alone or in combination with other references and/or the knowledge of one of ordinary skill in the art. *See, e.g.*: "FIG. 4 is a block diagram of the first embodiment of the invention in which the moving picture stream conversion in mode 1 is executed. A moving picture stream converting apparatus of FIG. 4 comprises a demultiplexer 30, a packetizer 32, and a multiplexer 34. The demultiplexer 30 receives an MPEG1 system stream as a conversion source and outputs elementary streams which were separated into video and audio streams. The packetizer 32 forms packets constructing a packetized elementary stream of MPEG2 with respect to each of the video and audio elementary streams which are supplied from the demultiplexer 30. The packet is usually called a PES packet. The multiplexer 34 divides each of the video and audio PES packets into a TS packet having a fixed length of 188 bytes and subsequently multiplexes, and outputs the resultant data as an MPEG2-TS." Hiroshima, 7:59–8:7.

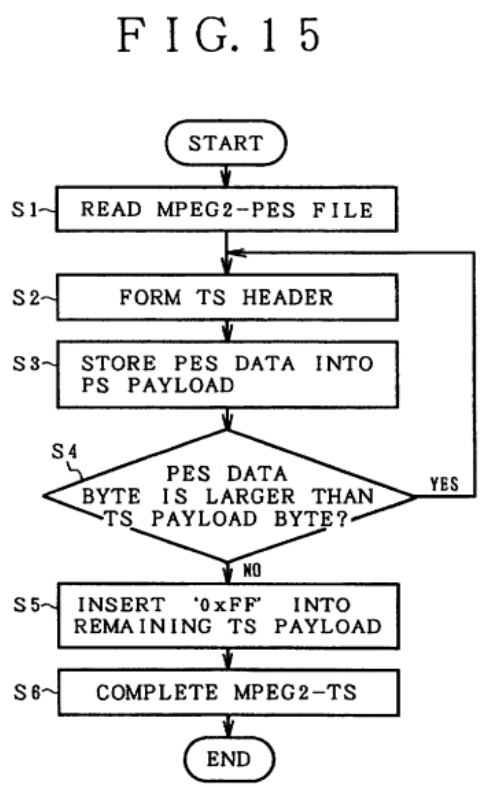
		FIG. 4  Hiroshima, Fig. 4. "FIG. 11 shows the details of the MPEG2-TS. Each packet of the MPEG2-TS has a prefix length of 188 bytes. The header 194 consists of four bytes and the payload 196 consists of 184 bytes. The header 194 is constructed by sync bytes 198, an error indicator 200, a unit start indicator 202, a transport priority 204, a packet ID 206, a scramble control 208, an adaptation field control 210, a continuity counter 212, and an adaptation field 214. The adaptation field 214 is constructed by an adaptation field length 216, a discontinuity indicator 218, a random access indicator 220, an elementary stream priority indicator 222, a 5-flag 224, an optional field 226, and stuffing bytes 228. Among them, the optional field 226 is constructed by a program clock reference 230, an original program clock reference field 232, a splice count down 234, a private data length 236, private data 238, an adaptation field extension 240, 3-flag 241, and an optional field 242." Hiroshima, 10:51–67.



Hiroshima, Fig. 11.

"FIG. 15 shows a process for forming the MPEG2-TS from the PES packet 82 by a multiplexing process in the multiplexer 34 in FIG. 5. First in step S1, the MPEG2-PES file is read. The TS header 194 in FIG. 11 is formed in step S2. The TS header 194 is formed in a manner such that the parameters such as error indicator 200 indicative of the presence or absence of the bit error in the packet, unit start indicator 202 indicating whether a new PES packet appears in the TS payload or not, packet ID 206 indicating whether the packet in the payload is video or audio, adaptation field 214, and adaptation field control 210 indicative of the presence or absence of the payload 196 are set in accordance with the state of the payload storage data of the 184-byte unit divided from the MPEG2-PES packet which had been read in step S1."

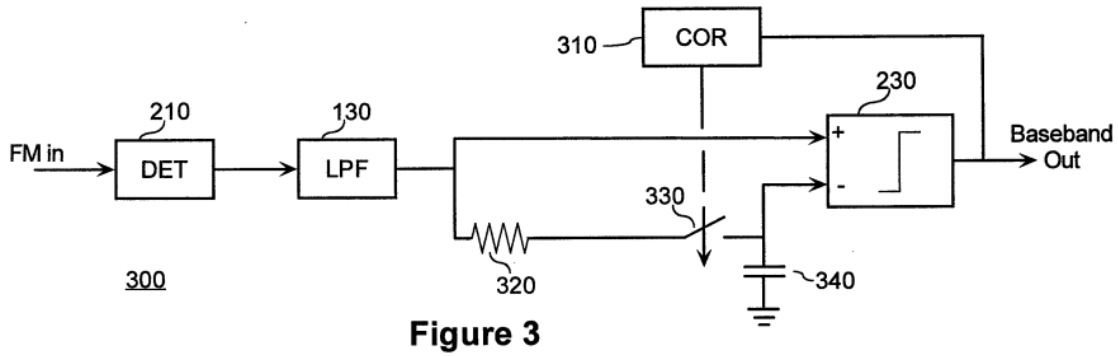
Hiroshima, 11:13–27.

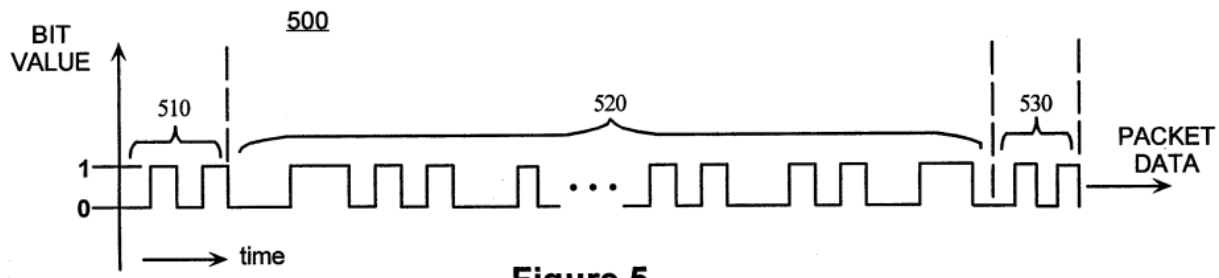


Hiroshima, Fig. 15.

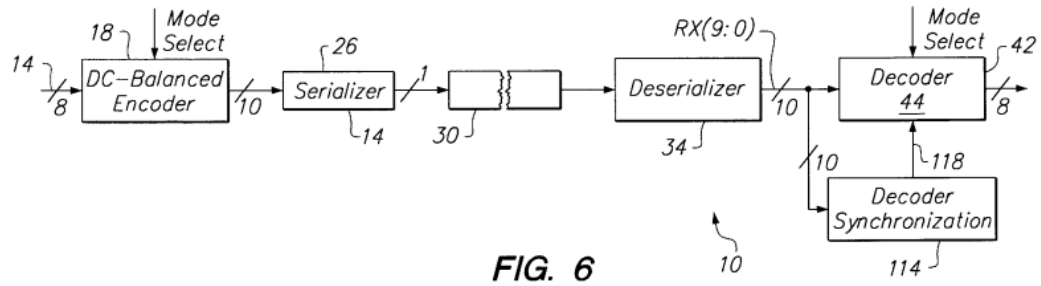
To the extent the preamble is limiting, Haartsen discloses this element and/or renders it obvious either alone or in combination with other references and/or the knowledge of one of ordinary skill in the art. *See, e.g.*:

"The present invention fulfills the above-described and other needs by providing a multi-part digital preamble for use in transmitting digital data packets. Advantageously, a preamble according to the invention substantially reduces the overall length of DC-free sequences that are transmitted with each data packet in a digital FM system. According to exemplary embodiments, a multi-part

		preamble includes a short, substantially DC-free leading part which enables coarse estimation of a prevailing DC level at an FM receiver. The exemplary preamble also includes a synchronization part which is not necessarily substantially DC-free and which carries timing and/or other useful information (e.g. , channel identification, destination address, etc.) that can vary from data packet to data packet (e.g. , depending on sender, recipient, time of day, etc.).” Haartsen, 5:2–15. “An exemplary DC estimation processor is depicted in Figure 3, wherein an exemplary FM demodulation processor 300 is shown to include a correlation processor 310, a resistor 320, a switch 330 and a capacitor 340, as well as the FM detector 210, the low-pass filter 130 and the slicer 230 of Figure 2.” Haartsen, 9:11–14.  **Figure 3** Haartsen, Fig. 3. “To alleviate this problem, the present invention discloses that a digital preamble can be strategically divided into three or more parts so that synchronization and other information can be included within the preamble while still enabling accurate DC estimation. Specifically, a short leading DC-free part of the preamble (e.g., alternating logical l 's and 0's, or any other sequence including an equal number of logical l's and logical 0's) can be used to provide initial, coarse DC offset estimation. Thereafter, a synchronization part of the preamble, which is not necessarily DC

		free, but which is protected by code to prevent detection errors, can be used not only to synchronize detection of a data packet, but also to carry timing and/or other useful information (e.g., channel identification, destination address, time of day, user identification, etc.).” Haartsen, 12:13–13:2. “An exemplary multi-part preamble according to the invention is depicted in Figure 5. In the figure, a three-part preamble 500 is shown to include a DC-free leading sequence 510, followed by a multiple-bit synchronization word 520 which is not necessarily DC free and which can be different for each data packet. The synchronization word 520, which can include timing and/or other information and which is coded to protect against detection errors, is in turn followed by a trailing DC-free sequence 530. Although the leading and trailing DC-free sequences 510, 530 are shown in the figure as 4-bit sequences of alternating logical l 's and 0's, those of ordinary skill will appreciate that each DC-free sequence 510, 530 can, in practice, be shorter or longer in length and can include any DC-free sequence having an equal number of logical l's and logical 0's.” Haartsen, 13:13–14:2.  **Figure 5** Haartsen, Fig. 5. “When the exemplary preamble 500 is received by the system 300 of Figure 3, the leading DC-free sequence 510 induces a coarse estimate of the baseband DC offset on the storage capacitor 340. Although the estimate is coarse, it is sufficient to permit successful detection of the coded synchronization word 520 (e.g., by the correlation processor 310). Methods for coding of a

		synchronization word to protect against erroneous detection in the presence of some DC-offset are known to those of skill in the art. Once the correlation processor 310 has detected the synchronization word 520, timing between received bits is well known. Consequently, the correlation processor 310 can open the switch 330 precisely at the end of the trailing DC-free sequence 530 to thereby store an accurate estimate of the DC offset for reception and detection of the incoming digital data packet." Haartsen, 14:3–14. To the extent the preamble is limiting, Shin discloses this element and/or renders it obvious either alone or in combination with other references and/or the knowledge of one of ordinary skill in the art. *See, e.g.*: "Referring to FIG. 6, the deserializer 34 receives the sequence of 10-bit characters T[9:0] produced by the encoder and generates 10-bit parallel received data upon bit lines RX9, RX8, . . . , RX0 (i.e., RX[9:0]). This 10-bit parallel received data is provided over bit lines RX[9:0] to the decoder 44, as well as to a decoder synchronization module 114. As is described below in section IV, the synchronization module 114 is operative to ascertain boundaries within the 10-bit parallel received data corresponding to the frame boundaries of the transmitted data (i.e., to T[9:0]). Specifically, synchronization module 114 determines upon which of the bit lines RX[9;0] the deserializer 34 is providing the received bits corresponding to the first bit T[0] of each transmitted byte T[9:0]. Upon making this determination, the synchronization module 114 provides a frame boundary pointer 118 to decoder 44 identifying the one of the bit lines RX[9:0] corresponding to the first bit T[0] of each transmitted 10-bit character T[9:0]. Upon receiving this synchronization information, the decoder 44 is disposed to decode the received data RX[9:0] in the following manner." Shin, 13:37–57.

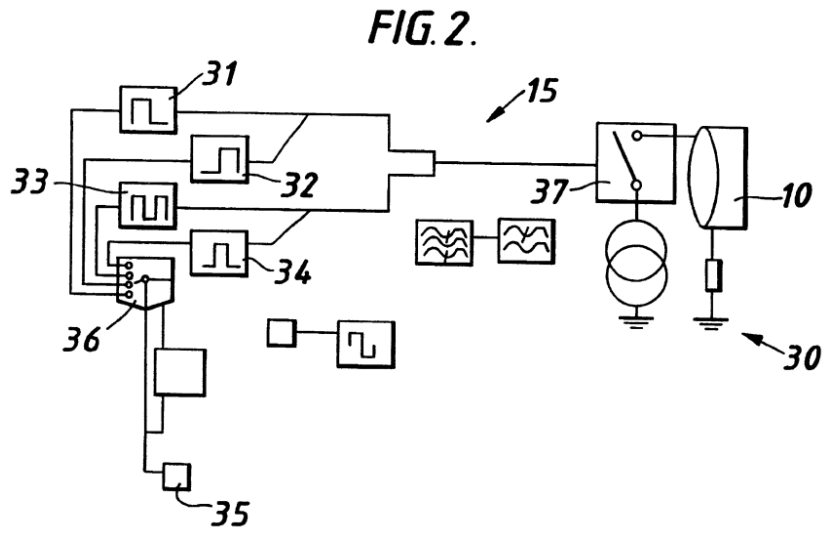


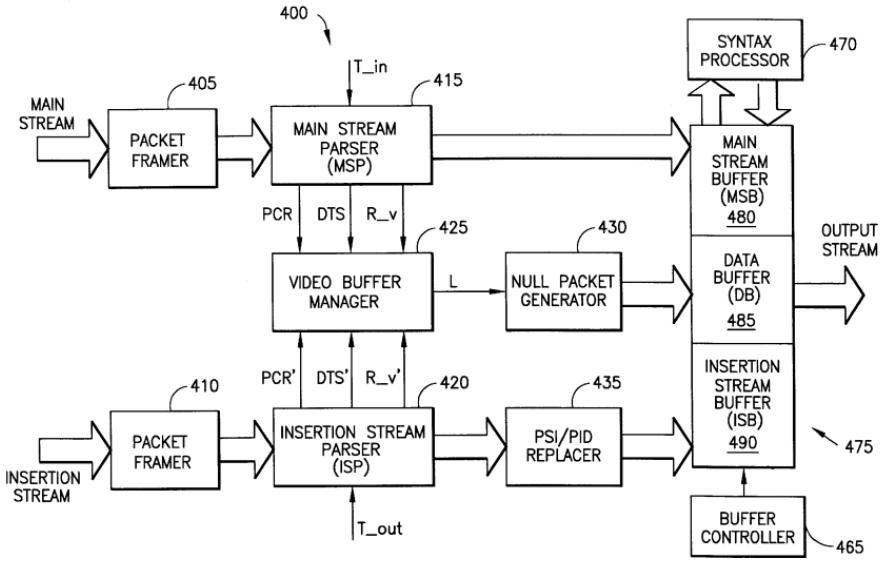
Shin, Fig. 6.

To the extent the preamble is limiting, Austin discloses this element and/or renders it obvious either alone or in combination with other references and/or the knowledge of one of ordinary skill in the art. *See, e.g.*:

"FIG. 2 shows the structure of a transmitter suitable for transformer/rectifier sites. The transmitter 30 comprises waveform generators 31-34 which transmitter receives the data at input 35 which selects 10 Hz tone as an indication of data '1' and a 15 Hz tone as a data '0' via selector 36. Using Fast Frequency Shift Keying (FFSK) at 10 baud, the modulated data signal is applied by modulating part of the current applied to the pipe (current switch 37) which provides cathodic protection (CP) as a result of the pipeline resistance. The signals are in digital form with multistage representations of the two modulating tones. Each state of these tones represents the switching levels for the transformer/rectifier pipeline devices. The bandwidth of the pipe is such that the switching harmonics are considerably reduced through their passage in the pipe so are of no great concern at the receiver."

Austin, 2:32-47.

		 Austin, Fig. 2. "This system uses up to 5 pre-defined data packet sizes and up to 16 user defined data message types within each packet. Each packet has Forward Error Connection (FEC) and bit interleaving applied to it on transmission. Error checking is performed using a standard CRC. The system uses standard run in and byte synchronisation techniques. The byte synchronisation word used is 32 _bits long and there are 5_different words used, one to identify each of the different packet types." Austin, 4:37-45. To the extent that Plaintiff contends that the Additional Prior Art References do not explicitly or inherently disclose this element, or render obvious this element, this element is obvious in combination with one or more references and/or with the knowledge of a person of ordinary skill in the art, as set forth in Defendant's Invalidity Contentions.
19(a)	an identification component that identifies a	Chen discloses and/or renders obvious this limitation. *See, e.g.*:

	packet type of a packet of symbols; and	"FIG. 4 is a block diagram of an Insertion Processing Unit (IPU) of a Digital Ad Insertion Module in accordance with the present invention. The IPU, shown generally at 400, receives the main stream at a packet framer 405, while the insertion stream is received at another packet framer 410. After processing by the packet framer 405, the main stream is provided to a Main Stream Parser (MSP) 415." Chen, 5:62–6:1.  FIG.4 Chen, Fig. 4. "The main stream parser 415 provides the main stream data to a main stream buffer 480, which communicates with a syntax processor 470. The main stream buffer 480 is part of the output buffer 475. The insertion stream parser 420 provides the insertion stream data to a Program Specific Information (PSI)/Program Identifier (PID) Replacer 435, which replaces the PSI tables and PIDs of the insertion stream with those of the main stream. The PSI/PID replacer 435 retrieves the relevant information from the main stream using a communication path which is not shown. The

		PSI and PID data of the main stream provide the information to connect the various audio, video, and data packets to a particular programming service. Specifically, this is accomplished using a Program Association Table (PAT) and a Program Map Table (PMT). The insertion stream is then provided to an Insertion Stream Buffer (ISB) 490, which is part of the output buffer 475. The ISB 490 communicates with a syntax processor 470. Furthermore, the output buffer 475 receives commands from a buffer controller 465 to provide an output stream with the insertion stream seamlessly spliced into the main stream." Chen, 6:33–54. "FIG. 5 illustrates the processing flow of the Insertion Processing Unit of FIG. 4 in accordance with the present invention. Note that the steps of FIG. 5 are shown as occurring in a sequential manner for clarity, but many of the steps may occur concurrently and/or sequentially. At block 502, the IPU outputs the main transport stream from the main stream buffer (MSB) 480. The IPU is essentially in a bypass mode at this time since the insertion stream has not yet been inserted. At block 506, the main stream parser parses the main stream to locate the splicing signal T-- in. Note that T-- in need not be carried in the main stream, but may be provided locally, for example, at a cable system headend. When T-- in is detected, the IPU continues to output the main transport stream from the main stream buffer. At block 508, the main stream parser parses the main stream to locate the closest packet with an anchor picture (e.g., I or P picture) start code and passes that packet to the syntax processor 470. This packet will be the last packet of the main stream. To ensure that the resulting data stream is MPEG compliant, no data from the next frame or sequence of frames may be included in this last main stream packet. At block 512, the syntax processor adds a sequence end code field to the last packet of the main stream. The sequence end code is a twenty-four bit code which indicates the end of a sequence of pictures. At block 514, the PSI/PID replacer 435 starts to replace the PSI/PID of the insertion stream." Chen, 12:28–53.

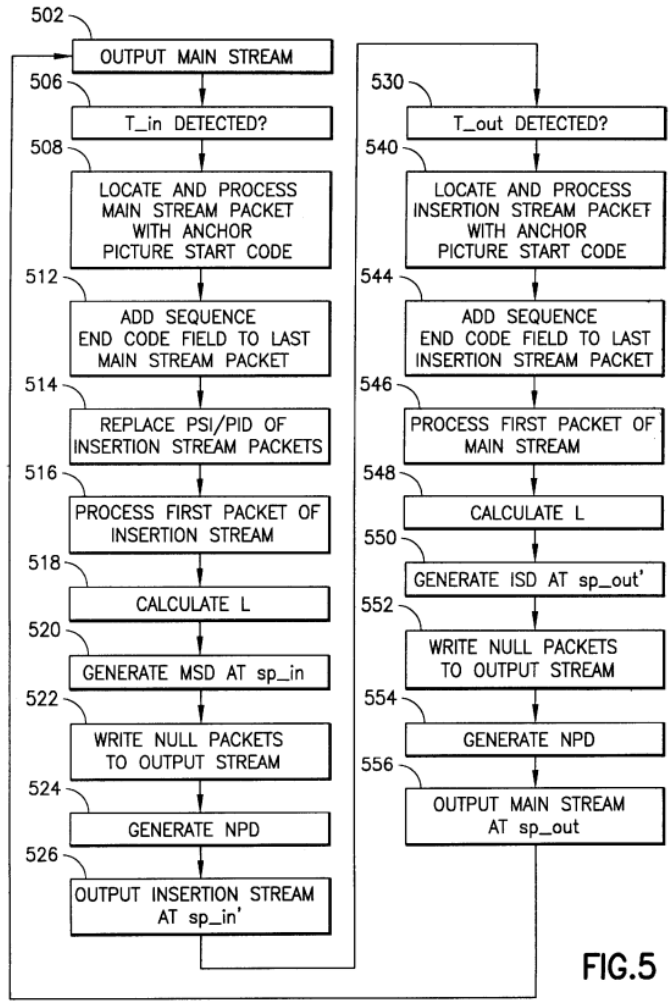
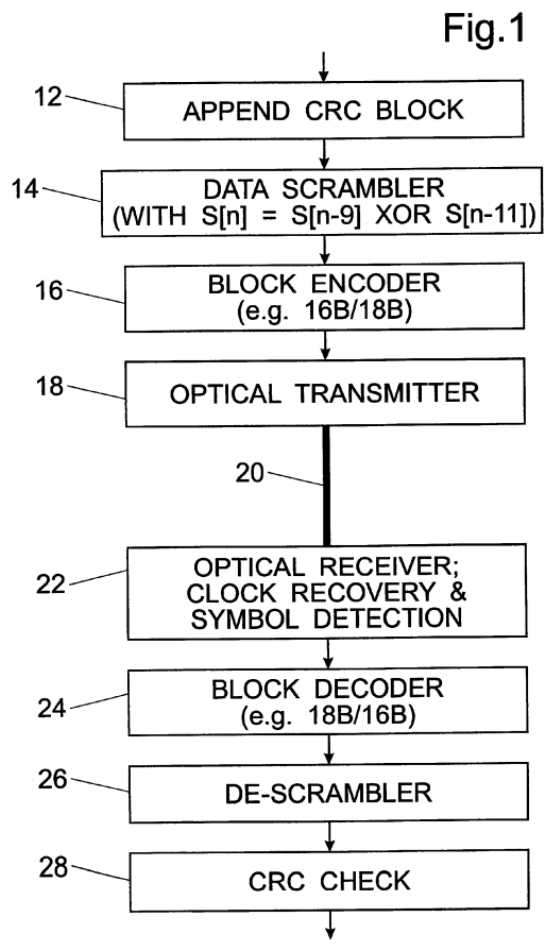


FIG.5

Chen, Fig. 5.

"In the MPEG-2 system (and the similar DigiCipher® II system proprietary to General Instrument Corporation, the assignee hereof) a transport stream, or transport multiplex is made up of a

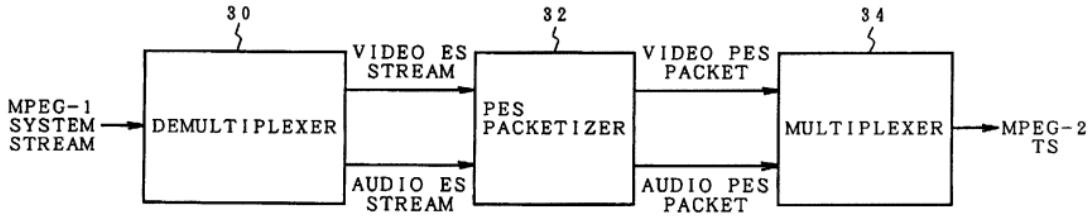
		contiguous set of fixed length packets. Each packet is 188 total bytes in length, with the first four of those bytes being defined as the packet header. The payload portion of each packet is thus normally 184 bytes. However, a variable length adaptation field may be provided to extend the header, when required. When an adaptation field is present, the payload portion of the packet will be correspondingly shorter. Various timing and identification information is provided in different portions of the transport stream. These include a packet identifier (PID) found in the transport header of each transport packet to provide a reference number for identifying the transport packets carrying a specific service component. This number is included in a service definition or "service map" used by the receiver to identify those transport packets required to reconstruct a television program signal. The PID may also be referenced for various grooming and remultiplexing functions. In the case of video, audio or isochronous control data, the stream of packets labeled with a single PID represents a single video, audio or isochronous data service elementary stream, respectively. Each type of packet will have a different PID identifying the packet type." Chen, 14:45–15:3. Cunningham discloses and/or renders obvious this limitation. *See, e.g.*: "Referring to FIG. 1, a data frame intended to be communicated over an optical fibre channel 20 typically comprises a binary digital message (payload) to be transferred between stations on the network; this message may comprise user data, or control information for managing operation of the network and its components. A circuit 12 generates an associated cyclic redundancy check (CRC) block containing check data derived from this message in known manner in accordance with a predetermined CRC algorithm, and appends the CRC block to the message." Cunningham, 2:65–3:7.

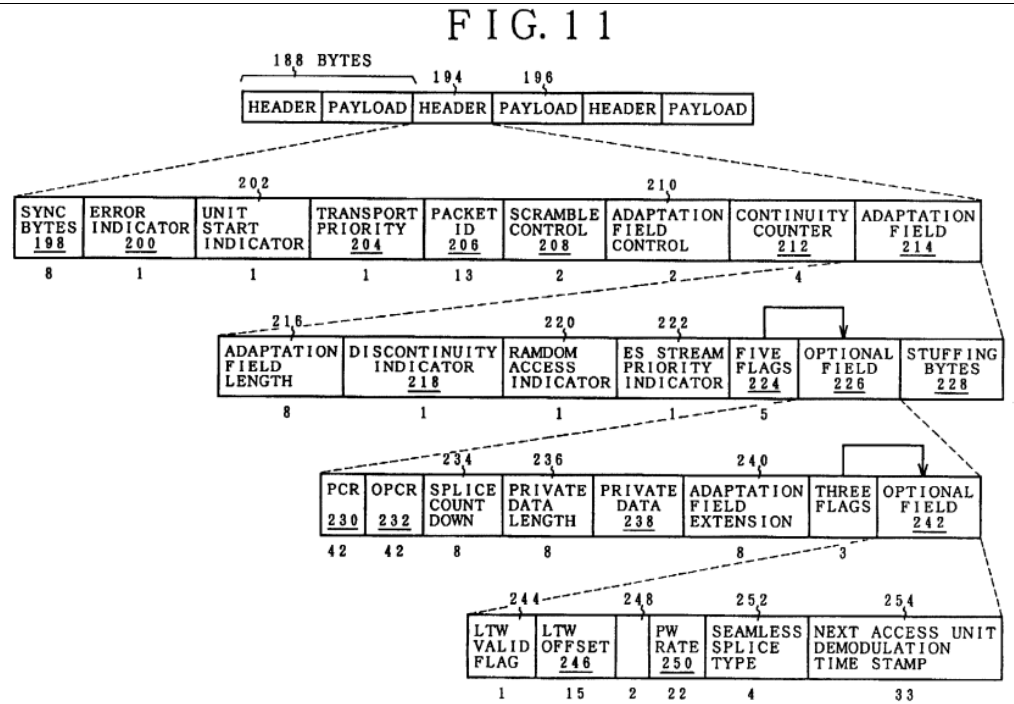


Cunningham, Fig. 1.

"Control information is conveyed by using selected n-bit data block values, together with a distinctive value of 10 for the pair of label bits a and b to indicate that the associated data block represents control information rather than actual data. The data block values used for this purpose are selected to be values which inherently have a disparity of zero, such as (for 8-bit blocks)

		00001111 or 10101010. It will be noted that the label bit pair value 10 likewise has zero disparity, so the complete encoded block also has zero disparity." Cunningham, 4:23–31. "The encoded blocks are transmitted over the optical fibre channel 20 (leftmost symbol first), and upon receipt at their destination they are detected by a receiver 22 where compensation is made in known manner for the effects of timing drift and other signal degradation to produce a restored digital data stream." Cunningham, 4:45–50. Hiroshima discloses and/or renders obvious this limitation. *See, e.g*: "FIG. 4 is a block diagram of the first embodiment of the invention in which the moving picture stream conversion in mode 1 is executed. A moving picture stream converting apparatus of FIG. 4 comprises a demultiplexer 30, a packetizer 32, and a multiplexer 34. The demultiplexer 30 receives an MPEG1 system stream as a conversion source and outputs elementary streams which were separated into video and audio streams. The packetizer 32 forms packets constructing a packetized elementary stream of MPEG2 with respect to each of the video and audio elementary streams which are supplied from the demultiplexer 30. The packet is usually called a PES packet. The multiplexer 34 divides each of the video and audio PES packets into a TS packet having a fixed length of 188 bytes and subsequently multiplexes, and outputs the resultant data as an MPEG2-TS." Hiroshima, 7:59–8:7.

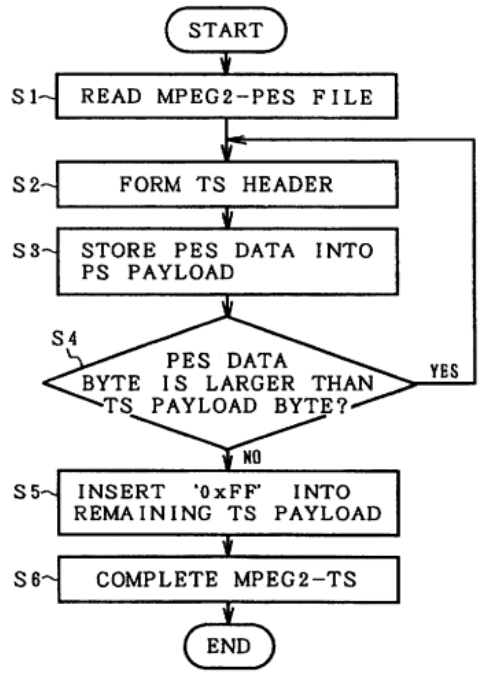
		FIG. 4  Hiroshima, Fig. 4. "FIG. 11 shows the details of the MPEG2-TS. Each packet of the MPEG2-TS has a prefix length of 188 bytes. The header 194 consists of four bytes and the payload 196 consists of 184 bytes. The header 194 is constructed by sync bytes 198, an error indicator 200, a unit start indicator 202, a transport priority 204, a packet ID 206, a scramble control 208, an adaptation field control 210, a continuity counter 212, and an adaptation field 214. The adaptation field 214 is constructed by an adaptation field length 216, a discontinuity indicator 218, a random access indicator 220, an elementary stream priority indicator 222, a 5-flag 224, an optional field 226, and stuffing bytes 228. Among them, the optional field 226 is constructed by a program clock reference 230, an original program clock reference field 232, a splice count down 234, a private data length 236, private data 238, an adaptation field extension 240, 3-flag 241, and an optional field 242." Hiroshima, 10:51–67.



Hiroshima, Fig. 11.

"FIG. 15 shows a process for forming the MPEG2-TS from the PES packet 82 by a multiplexing process in the multiplexer 34 in FIG. 5. First in step S1, the MPEG2-PES file is read. The TS header 194 in FIG. 11 is formed in step S2. The TS header 194 is formed in a manner such that the parameters such as error indicator 200 indicative of the presence or absence of the bit error in the packet, unit start indicator 202 indicating whether a new PES packet appears in the TS payload or not, packet ID 206 indicating whether the packet in the payload is video or audio, adaptation field 214, and adaptation field control 210 indicative of the presence or absence of the payload 196 are set in accordance with the state of the payload storage data of the 184-byte unit divided from the MPEG2-PES packet which had been read in step S1."

Hiroshima, 11:13–27.

		FIG. 15  Hiroshima, Fig. 15. Austin discloses and/or renders obvious this limitation. *See, e.g.*: "This system uses up to 5 pre-defined data packet sizes and up to 16 user defined data message types within each packet. Each packet has Forward Error Connection (FEC) and bit interleaving applied to it on transmission. Error checking is performed using a standard CRC. The system uses standard run in and byte synchronisation techniques. The byte synchronisation word used is 32 _bits long and there are 5_different words used, one to identify each of the different packet types."

		Austin, 4:37-45. To the extent that Plaintiff contends that the Additional Prior Art References do not explicitly or inherently disclose this element, or render obvious this element, this element is obvious in combination with one or more references and/or with the knowledge of a person of ordinary skill in the art, as set forth in Defendant's Invalidity Contentions.
19(b)	a transmission component that transmits a synchronization symbol that corresponds to the identified packet type, the transmitted synchronization symbol providing synchronization information and each packet type having a different synchronization symbol and permitting an external receiving node to properly align with a synchronization primitive to be correctly aligned on a symbol boundary, and that transmits the symbols of the packet.	Chen discloses and/or renders obvious this limitation. *See, e.g.*: "FIG. 4 is a block diagram of an Insertion Processing Unit (IPU) of a Digital Ad Insertion Module in accordance with the present invention. The IPU, shown generally at 400, receives the main stream at a packet framer 405, while the insertion stream is received at another packet framer 410. After processing by the packet framer 405, the main stream is provided to a Main Stream Parser (MSP) 415." Chen, 5:62–6:1.

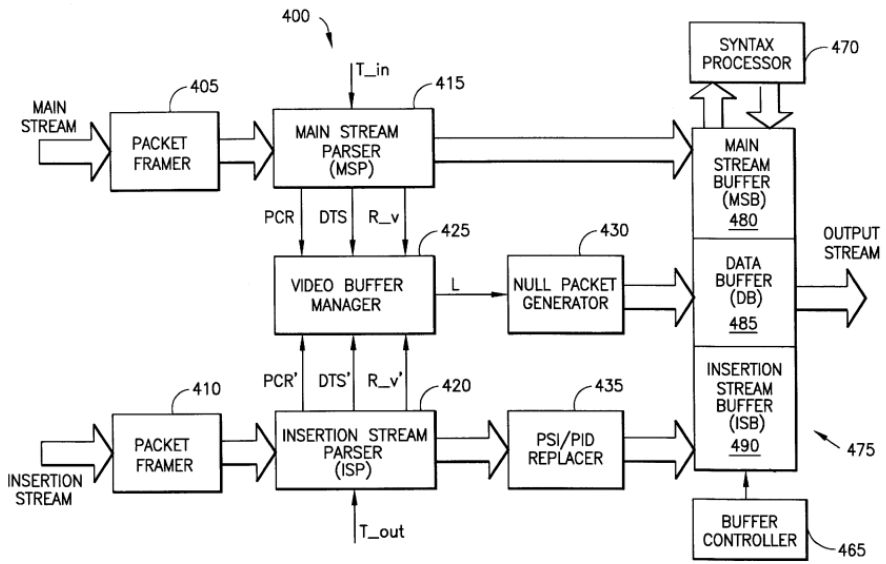


FIG.4

Chen, Fig. 4.

"The main stream parser 415 provides the main stream data to a main stream buffer 480, which communicates with a syntax processor 470. The main stream buffer 480 is part of the output buffer 475. The insertion stream parser 420 provides the insertion stream data to a Program Specific Information (PSI)/Program Identifier (PID) Replacer 435, which replaces the PSI tables and PIDs of the insertion stream with those of the main stream. The PSI/PID replacer 435 retrieves the relevant information from the main stream using a communication path which is not shown. The PSI and PID data of the main stream provide the information to connect the various audio, video, and data packets to a particular programming service. Specifically, this is accomplished using a Program Association Table (PAT) and a Program Map Table (PMT). The insertion stream is then provided to an Insertion Stream Buffer (ISB) 490, which is part of the output buffer 475. The ISB 490 communicates with a syntax processor 470. Furthermore, the output buffer 475 receives commands from a buffer controller 465 to provide an output stream with the insertion stream seamlessly spliced into the main stream."

		Chen, 6:33–54. "FIG. 5 illustrates the processing flow of the Insertion Processing Unit of FIG. 4 in accordance with the present invention. Note that the steps of FIG. 5 are shown as occurring in a sequential manner for clarity, but many of the steps may occur concurrently and/or sequentially. At block 502, the IPU outputs the main transport stream from the main stream buffer (MSB) 480. The IPU is essentially in a bypass mode at this time since the insertion stream has not yet been inserted. At block 506, the main stream parser parses the main stream to locate the splicing signal T-- in. Note that T-- in need not be carried in the main stream, but may be provided locally, for example, at a cable system headend. When T-- in is detected, the IPU continues to output the main transport stream from the main stream buffer. At block 508, the main stream parser parses the main stream to locate the closest packet with an anchor picture (e.g., I or P picture) start code and passes that packet to the syntax processor 470. This packet will be the last packet of the main stream. To ensure that the resulting data stream is MPEG compliant, no data from the next frame or sequence of frames may be included in this last main stream packet. At block 512, the syntax processor adds a sequence end code field to the last packet of the main stream. The sequence end code is a twenty-four bit code which indicates the end of a sequence of pictures. At block 514, the PSI/PID replacer 435 starts to replace the PSI/PID of the insertion stream." Chen, 12:28–53.

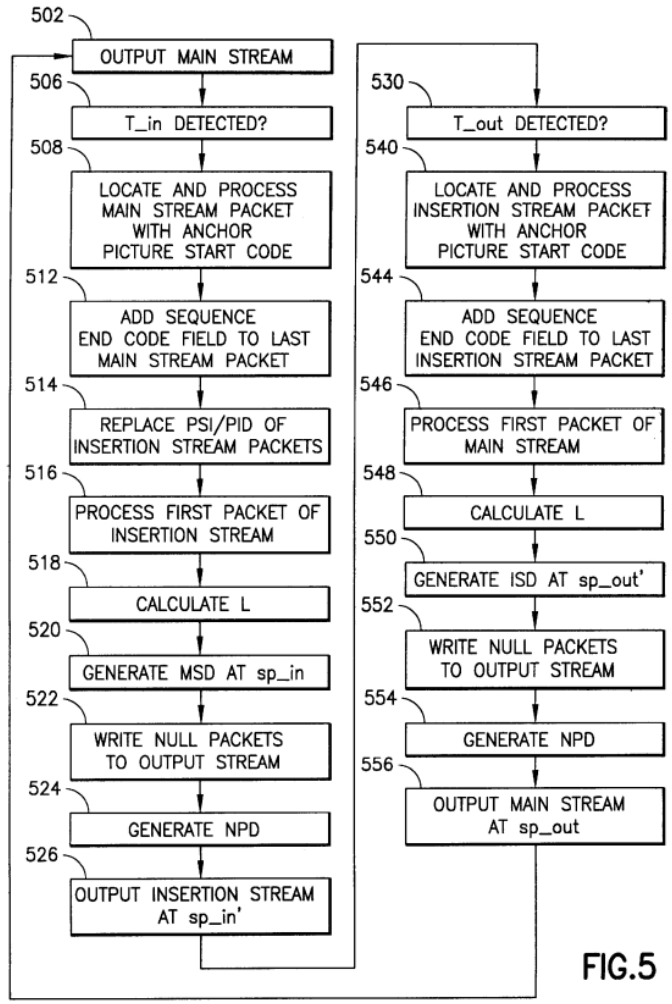
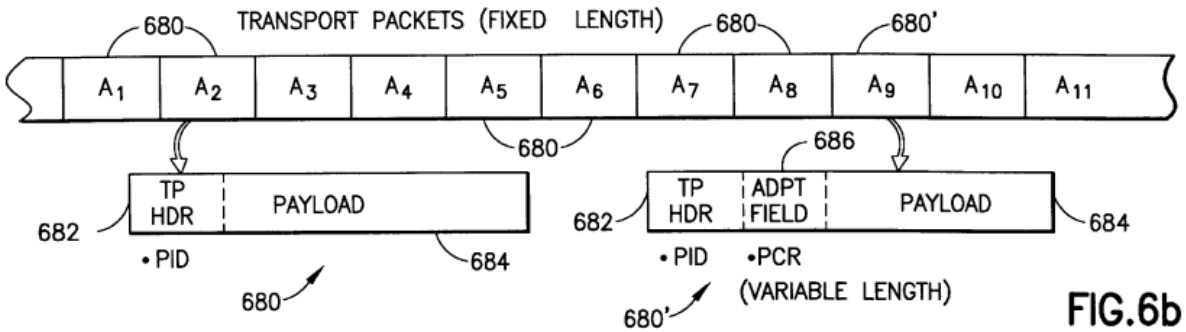


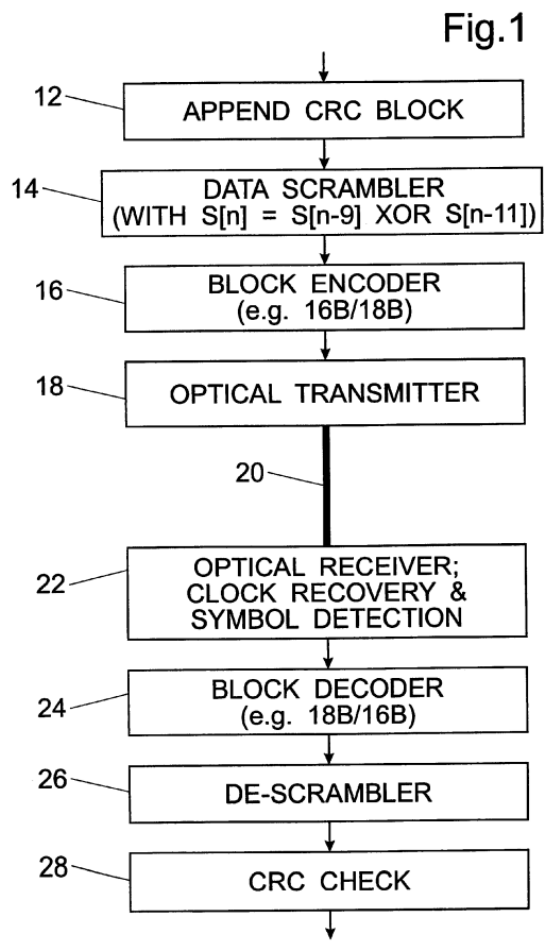
FIG.5

Chen, Fig. 5.

"In the MPEG-2 system (and the similar DigiCipher® II system proprietary to General Instrument Corporation, the assignee hereof) a transport stream, or transport multiplex is made up of a

		contiguous set of fixed length packets. Each packet is 188 total bytes in length, with the first four of those bytes being defined as the packet header. The payload portion of each packet is thus normally 184 bytes. However, a variable length adaptation field may be provided to extend the header, when required. When an adaptation field is present, the payload portion of the packet will be correspondingly shorter. Various timing and identification information is provided in different portions of the transport stream. These include a packet identifier (PID) found in the transport header of each transport packet to provide a reference number for identifying the transport packets carrying a specific service component. This number is included in a service definition or "service map" used by the receiver to identify those transport packets required to reconstruct a television program signal. The PID may also be referenced for various grooming and remultiplexing functions. In the case of video, audio or isochronous control data, the stream of packets labeled with a single PID represents a single video, audio or isochronous data service elementary stream, respectively. Each type of packet will have a different PID identifying the packet type. Timing information carried by the transport stream includes a program clock reference (PCR) which effectively represents a sample of the system time clock (STC) time base that underlies the service composed of the PIDs referenced in the service map. The PID carrying the packet with the PCR is also referenced in the service map. The video, audio and isochronous data components of a service are locked through a defined relationship to the system time clock. The PCR serves to define the transport rate, in the sense that between any two successive PCRs in one PID, the transport rate is constant and nominally equal to the system time clock rate times the ratio of the total number of transport bytes between the PCRs divided by the difference in the PCRs in units of system time clock ticks." Chen, 14:45–15:17. "As indicated in FIG. 6b, the transport header of each transport packet includes the PID which identifies the particular service component carried by the transport packet. The PCR will be carried by an adaptation field to provide timing information for a desired service. At a receiver, the PCR for the desired service is detected from the adaptation field. The PIDs of the transport packets will

		then be monitored in accordance with the timing established by the PCR to recover those transport packets carrying a particular component of the service to be processed." Chen, 16:41–50.  Chen, Fig. 6b. Cunningham discloses and/or renders obvious this limitation. *See, e.g.*: "Referring to FIG. 1, a data frame intended to be communicated over an optical fibre channel 20 typically comprises a binary digital message (payload) to be transferred between stations on the network; this message may comprise user data, or control information for managing operation of the network and its components. A circuit 12 generates an associated cyclic redundancy check (CRC) block containing check data derived from this message in known manner in accordance with a predetermined CRC algorithm, and appends the CRC block to the message." Cunningham, 2:65–3:7.



Cunningham, Fig. 1.

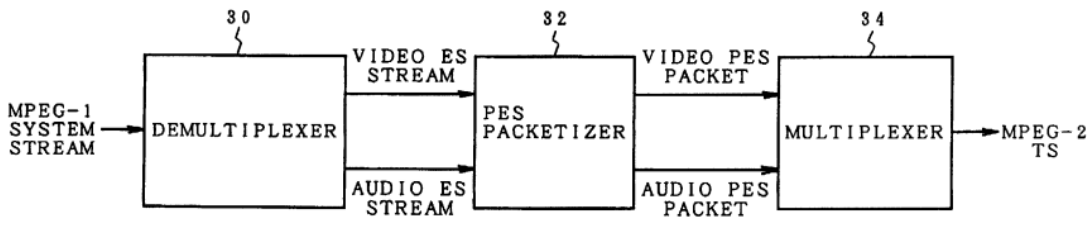
"Control information is conveyed by using selected n-bit data block values, together with a distinctive value of 10 for the pair of label bits a and b to indicate that the associated data block represents control information rather than actual data. The data block values used for this purpose are selected to be values which inherently have a disparity of zero, such as (for 8-bit blocks)

		00001111 or 10101010. It will be noted that the label bit pair value 10 likewise has zero disparity, so the complete encoded block also has zero disparity." Cunningham, 4:23–31. "The encoded blocks are transmitted over the optical fibre channel 20 (leftmost symbol first), and upon receipt at their destination they are detected by a receiver 22 where compensation is made in known manner for the effects of timing drift and other signal degradation to produce a restored digital data stream." Cunningham, 4:45–50. "There are many potential ways to achieve code block synchronization (i.e. correct alignment in the receiver of the label and payload fields). For example, one technique would be to transmit a 'comma sequence' during start-up. As an example of this technique consider a code with n=8 using the simple control payload scheme. A training control state of 1111 1111 would be transmitted as: [10 1111 1111 0000 0000][10 1111 1111 0000 0000][10 1111 1111 0000 0000] The receiver searches for the bit pair 10. This is found in only two locations: the correct location (the label field), and at the end of the run of eight 1's. However, if the receiver (incorrectly) identifies the label field to be the 10 bit pair at the end of the run of 1's, the following 16 bits are not a valid control payload (first 8 bits are not the complement of the second 8 bits). The receiver can therefore detect the mistake. It is also noted that this sequence contains a regular 010 pattern, which aids in phase-locked loop (PLL) clock acquisition." Cunningham, 7:5–23. Hiroshima discloses and/or renders obvious this limitation. *See, e.g*: "FIG. 4 is a block diagram of the first embodiment of the invention in which the moving picture stream conversion in mode 1 is executed. A moving picture stream converting apparatus of FIG. 4 comprises a demultiplexer 30, a packetizer 32, and a multiplexer 34. The demultiplexer 30 receives an MPEG1 system stream as a conversion source and outputs elementary streams which were

separated into video and audio streams. The packetizer 32 forms packets constructing a packetized elementary stream of MPEG2 with respect to each of the video and audio elementary streams which are supplied from the demultiplexer 30. The packet is usually called a PES packet. The multiplexer 34 divides each of the video and audio PES packets into a TS packet having a fixed length of 188 bytes and subsequently multiplexes, and outputs the resultant data as an MPEG2-TS."

Hiroshima, 7:59–8:7.

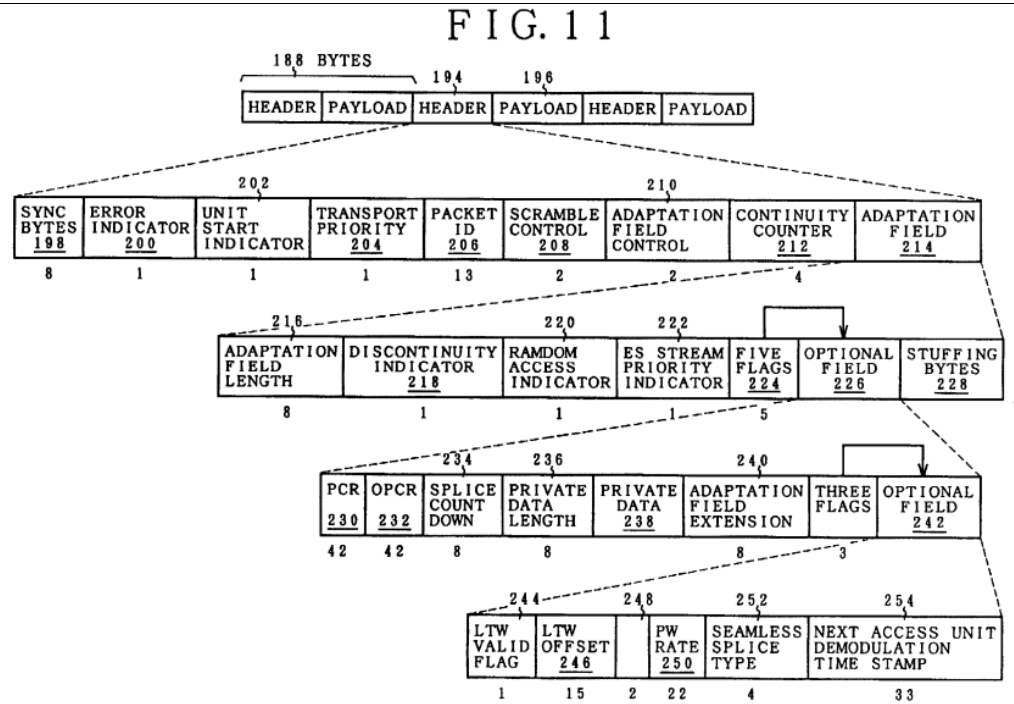
FIG. 4



Hiroshima, Fig. 4.

"FIG. 11 shows the details of the MPEG2-TS. Each packet of the MPEG2-TS has a prefix length of 188 bytes. The header 194 consists of four bytes and the payload 196 consists of 184 bytes. The header 194 is constructed by sync bytes 198, an error indicator 200, a unit start indicator 202, a transport priority 204, a packet ID 206, a scramble control 208, an adaptation field control 210, a continuity counter 212, and an adaptation field 214. The adaptation field 214 is constructed by an adaptation field length 216, a discontinuity indicator 218, a random access indicator 220, an elementary stream priority indicator 222, a 5-flag 224, an optional field 226, and stuffing bytes 228. Among them, the optional field 226 is constructed by a program clock reference 230, an original program clock reference field 232, a splice count down 234, a private data length 236, private data 238, an adaptation field extension 240, 3-flag 241, and an optional field 242."

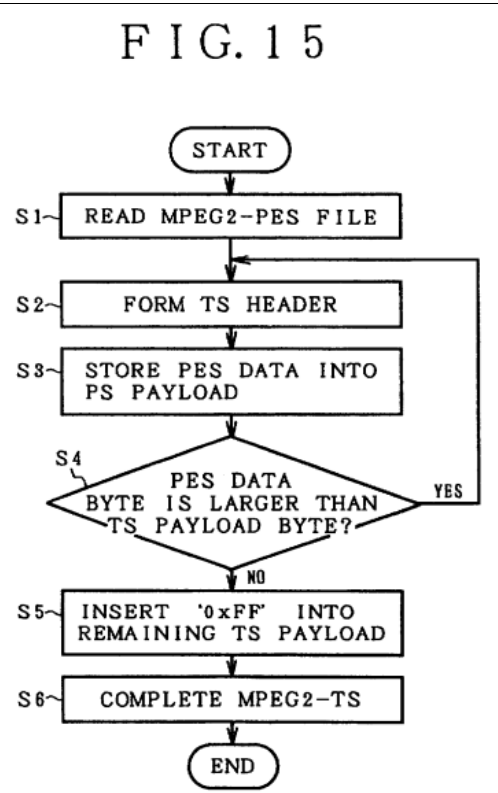
Hiroshima, 10:51–67.



Hiroshima, Fig. 11.

"FIG. 15 shows a process for forming the MPEG2-TS from the PES packet 82 by a multiplexing process in the multiplexer 34 in FIG. 5. First in step S1, the MPEG2-PES file is read. The TS header 194 in FIG. 11 is formed in step S2. The TS header 194 is formed in a manner such that the parameters such as error indicator 200 indicative of the presence or absence of the bit error in the packet, unit start indicator 202 indicating whether a new PES packet appears in the TS payload or not, packet ID 206 indicating whether the packet in the payload is video or audio, adaptation field 214, and adaptation field control 210 indicative of the presence or absence of the payload 196 are set in accordance with the state of the payload storage data of the 184-byte unit divided from the MPEG2-PES packet which had been read in step S1."

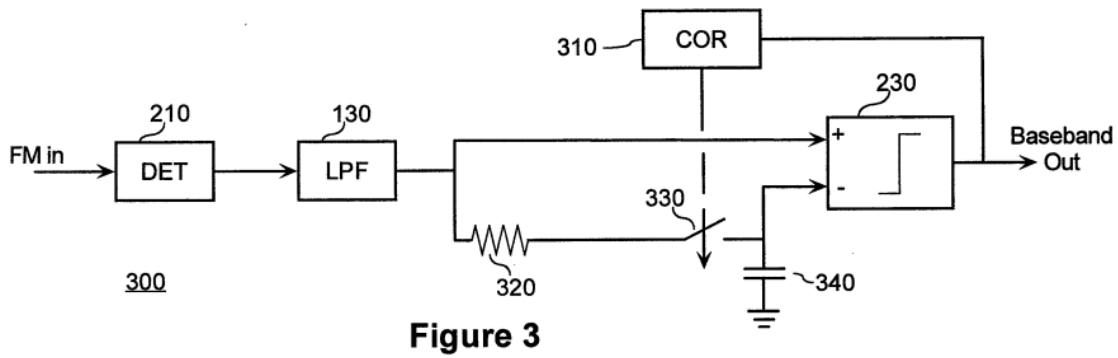
Hiroshima, 11:13–27.

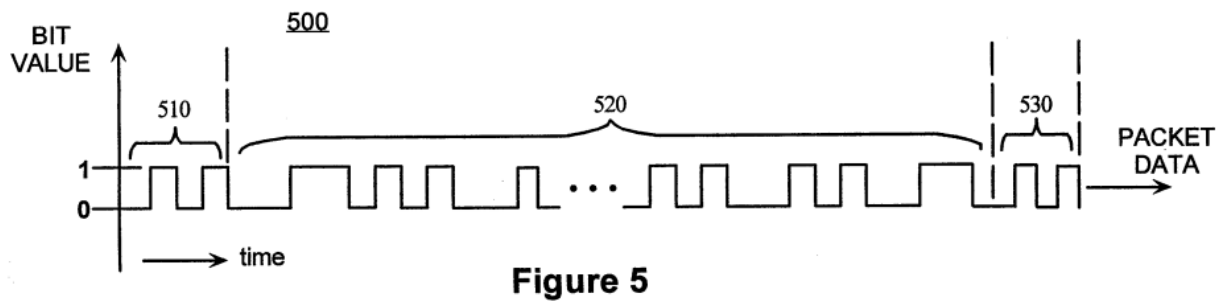


Hiroshima, Fig. 15.

Haartsen discloses and/or renders obvious this limitation. *See, e.g.*:

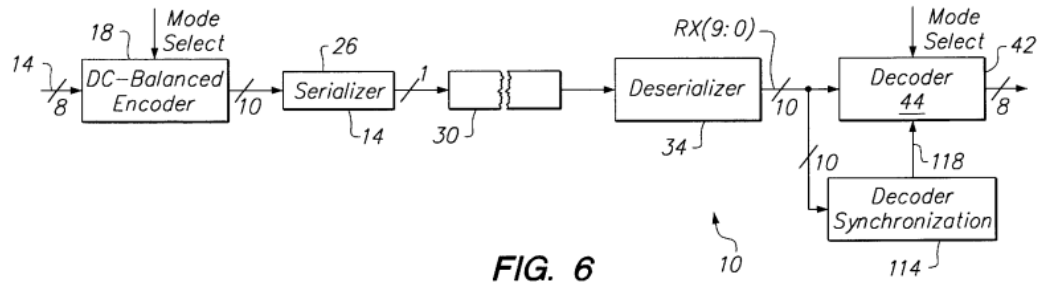
"The present invention fulfills the above-described and other needs by providing a multi-part digital preamble for use in transmitting digital data packets. Advantageously, a preamble according to the invention substantially reduces the overall length of DC-free sequences that are transmitted with each data packet in a digital FM system. According to exemplary embodiments, a multi-part preamble includes a short, substantially DC-free leading part which enables coarse estimation of a prevailing DC level at an FM receiver. The exemplary preamble also includes a synchronization

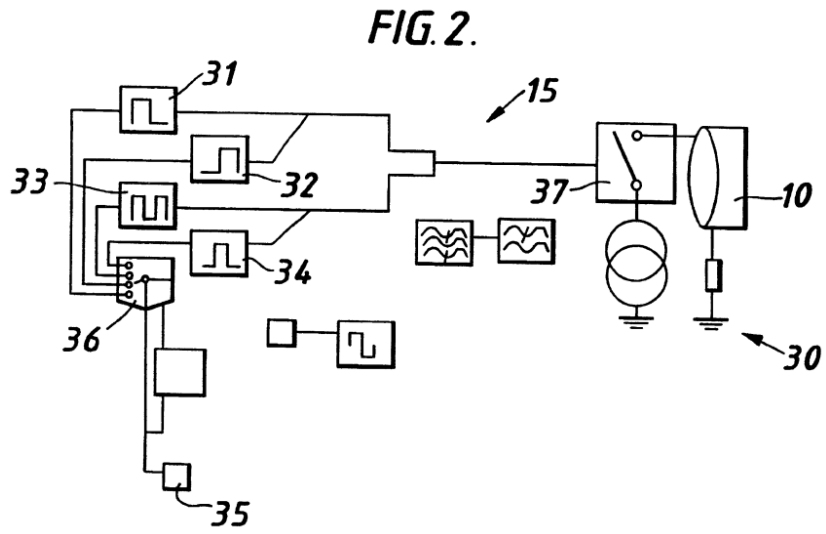
		part which is not necessarily substantially DC-free and which carries timing and/or other useful information (e.g. , channel identification, destination address, etc.) that can vary from data packet to data packet (e.g. , depending on sender, recipient, time of day, etc.).” Haartsen, 5:2–15. “An exemplary DC estimation processor is depicted in Figure 3, wherein an exemplary FM demodulation processor 300 is shown to include a correlation processor 310, a resistor 320, a switch 330 and a capacitor 340, as well as the FM detector 210, the low-pass filter 130 and the slicer 230 of Figure 2.” Haartsen, 9:11–14.  **Figure 3** Haartsen, Fig. 3. “To alleviate this problem, the present invention discloses that a digital preamble can be strategically divided into three or more parts so that synchronization and other information can be included within the preamble while still enabling accurate DC estimation. Specifically, a short leading DC-free part of the preamble (e.g., alternating logical l 's and 0's, or any other sequence including an equal number of logical l's and logical 0's) can be used to provide initial, coarse DC offset estimation. Thereafter, a synchronization part of the preamble, which is not necessarily DC free, but which is protected by code to prevent detection errors, can be used not only to synchronize

		detection of a data packet, but also to carry timing and/or other useful information (e.g., channel identification, destination address, time of day, user identification, etc.).” Haartsen, 12:13–13:2. “An exemplary multi-part preamble according to the invention is depicted in Figure 5. In the figure, a three-part preamble 500 is shown to include a DC-free leading sequence 510, followed by a multiple-bit synchronization word 520 which is not necessarily DC free and which can be different for each data packet. The synchronization word 520, which can include timing and/or other information and which is coded to protect against detection errors, is in turn followed by a trailing DC-free sequence 530. Although the leading and trailing DC-free sequences 510, 530 are shown in the figure as 4-bit sequences of alternating logical l 's and 0's, those of ordinary skill will appreciate that each DC-free sequence 510, 530 can, in practice, be shorter or longer in length and can include any DC-free sequence having an equal number of logical l's and logical 0's.” Haartsen, 13:13–14:2.  Haartsen, Fig. 5. “When the exemplary preamble 500 is received by the system 300 of Figure 3, the leading DC-free sequence 510 induces a coarse estimate of the baseband DC offset on the storage capacitor 340. Although the estimate is coarse, it is sufficient to permit successful detection of the coded synchronization word 520 (e.g., by the correlation processor 310). Methods for coding of a synchronization word to protect against erroneous detection in the presence of some DC-offset are

		known to those of skill in the art. Once the correlation processor 310 has detected the synchronization word 520, timing between received bits is well known. Consequently, the correlation processor 310 can open the switch 330 precisely at the end of the trailing DC-free sequence 530 to thereby store an accurate estimate of the DC offset for reception and detection of the incoming digital data packet. As is described above, the fact that the DC-free sequences 510, 530 can be very short, combined with the fact that the synchronization word 520 can carry useful information, means that the overhead associated with the exemplary three- part preamble 500 is considerably reduced as compared to a conventional DC-free preamble. Those of skill in the art will appreciate that the precise lengths of the three parts 510, 520, 530 of the preamble 500 shown in Figure 5 can vary and are a matter of design choice. Additionally, the exemplary preamble 500 can include additional trailing DC-free sequences which can, for example, be embedded within the synchronization word 520." Haartsen, 14:3–15:2. Shin discloses and/or renders obvious this limitation. *See, e.g.*: "It may thus be appreciated that each eight-bit code converted into a 10-bit encoded character during high-transition mode operation includes four or more logical transitions. Similarly, each eight-bit code converted into a 10-bit encoded character during low-transition mode operation includes less than four logical transitions. These sets of 10-bit encoded characters capable of being produced during high-transition and low-transition modes of operation may be characterized as a high-transition set of "in-band" encoded characters, and a low-transition set of in-band encoded characters, respectively. Beyond the 256 in-band characters within the high-transition set and the 256 in-band characters of the low-transition set, there exists a high-transition set of 256 out-of-band 10-bit characters and a low-transition set of 10-bit characters. In accordance with another aspect of the invention, various synchronization and other special characters are defined using the high-transition and low-transition sets of out-of-band characters. Each character corresponding to one of these "out-of-band" characters associated with the high-transition set includes less than four logical transitions, and each of the out-of-band characters associated with the low-transition set of code characters includes more than four logical transitions. The difference in the number of transitions

		between in-band and out-of-band characters allows selected out-of-band characters to serve as control characters, and to be readily distinguished from in-band characters within the transmitted data stream." Shin, 9:49–10:8. "In accordance with one aspect of the invention, the synchronization characters associated with both the low-transition and high-transition sets of code characters are selected so as to facilitate rapid synchronization during data recovery. When the low-transition set of code characters is being employed, a special group of out-of-band characters is used during synchronization. Each special synchronization character includes a predefined number larger than four (e.g., 7) of logical transitions between binary character values, and a predefined number (e.g., two) of "non-transitions" between character values. As will be discussed below, the special synchronization characters are selected such that random logic may be used to distinguish each special synchronization character from the in-band characters of the low-transition set." Shin, 10:17–34. "Referring to FIG. 6, the deserializer 34 receives the sequence of 10-bit characters T[9:0] produced by the encoder and generates 10-bit parallel received data upon bit lines RX9, RX8, . . . , RX0 (i.e., RX[9:0]). This 10-bit parallel received data is provided over bit lines RX[9:0] to the decoder 44, as well as to a decoder synchronization module 114. As is described below in section IV, the synchronization module 114 is operative to ascertain boundaries within the 10-bit parallel received data corresponding to the frame boundaries of the transmitted data (i.e., to T[9:0]). Specifically, synchronization module 114 determines upon which of the bit lines RX[9;0] the deserializer 34 is providing the received bits corresponding to the first bit T[0] of each transmitted byte T[9:0]. Upon making this determination, the synchronization module 114 provides a frame boundary pointer 118 to decoder 44 identifying the one of the bit lines RX[9:0] corresponding to the first bit T[0] of each transmitted 10-bit character T[9:0]. Upon receiving this synchronization information, the decoder 44 is disposed to decode the received data RX[9:0] in the following manner." Shin, 13:37–57.

		 Shin, Fig. 6. Austin discloses and/or renders obvious this limitation. *See, e.g.:* "FIG. 2 shows the structure of a transmitter suitable for transformer/rectifier sites. The transmitter 30 comprises waveform generators 31-34 which transmitter receives the data at input 35 which selects 10 Hz tone as an indication of data '1' and a 15 Hz tone as a data '0' via selector 36. Using Fast Frequency Shift Keying (FFSK) at 10 baud, the modulated data signal is applied by modulating part of the current applied to the pipe (current switch 37) which provides cathodic protection (CP) as a result of the pipeline resistance. The signals are in digital form with multistage representations of the two modulating tones. Each state of these tones represents the switching levels for the transformer/rectifier pipeline devices. The bandwidth of the pipe is such that the switching harmonics are considerably reduced through their passage in the pipe so are of no great concern at the receiver." Austin, 2:32-47.



Austin, Fig. 2.

"This system uses up to 5 pre-defined data packet sizes and up to 16 user defined data message types within each packet. Each packet has Forward Error Connection (FEC) and bit interleaving applied to it on transmission. Error checking is performed using a standard CRC. The system uses standard run in and byte synchronisation techniques. The byte synchronisation word used is 32 _bits long and there are 5_different words used, one to identify each of the different packet types."

Austin, 4:37-45.

To the extent that Plaintiff contends that the Additional Prior Art References do not explicitly or inherently disclose this element, or render obvious this element, this element is obvious in combination with one or more references and/or with the knowledge of a person of ordinary skill in the art, as set forth in Defendant's Invalidity Contentions.

Claim 20

20[p]	The communications device of claim 19	The Additional Prior Art References disclose, expressly or inherently, the transmitting device of claim 19 under any proposed construction. *See* exemplary disclosures and corresponding discussion for claim 19.
20	wherein the symbols of the packet include in-band symbols and the synchronization symbols are out-of-band symbols.	Shin discloses and/or renders obvious this limitation. *See, e.g.*: "It may thus be appreciated that each eight-bit code converted into a 10-bit encoded character during high-transition mode operation includes four or more logical transitions. Similarly, each eight-bit code converted into a 10-bit encoded character during low-transition mode operation includes less than four logical transitions. These sets of 10-bit encoded characters capable of being produced during high-transition and low-transition modes of operation may be characterized as a high-transition set of "in-band" encoded characters, and a low-transition set of in-band encoded characters, respectively. Beyond the 256 in-band characters within the high-transition set and the 256 in-band characters of the low-transition set, there exists a high-transition set of 256 out-of-band 10-bit characters and a low-transition set of 10-bit characters. In accordance with another aspect of the invention, various synchronization and other special characters are defined using the high-transition and low-transition sets of out-of-band characters. Each character corresponding to one of these "out-of-band" characters associated with the high-transition set includes less than four logical transitions, and each of the out-of-band characters associated with the low-transition set of code characters includes more than four logical transitions. The difference in the number of transitions between in-band and out-of-band characters allows selected out-of-band characters to serve as control characters, and to be readily distinguished from in-band characters within the transmitted data stream." Shin, 9:49–10:8. "In accordance with one aspect of the invention, the synchronization characters associated with both the low-transition and high-transition sets of code characters are selected so as to facilitate rapid synchronization during data recovery. When the low-transition set of code characters is being employed, a special group of out-of-band characters is used during synchronization. Each special synchronization character includes a predefined number larger than four (e.g., 7) of logical transitions between binary character values, and a predefined number (e.g., two) of "non-transitions" between character values. As will be discussed below, the special synchronization

		characters are selected such that random logic may be used to distinguish each special synchronization character from the in-band characters of the low-transition set. The following constitutes an exemplary set of out-of-band synchronization characters for use with the low-transition set of code characters: 1100101010 1101001010 1101010010 1101010100 ” Shin, 10:17–40. “As mentioned above, the decoder synchronization module 114 provides an indication to the decoder 44 of the frame boundary of each transmitted 10-bit character T[9:0]. The decoder module 114, together with the decoder switch 150 (FIG. 8), effectively function as a barrel-shifter to shuffle the parallel data RX[9:0] from the deserializer into the frame-aligned data S[9:0]. In accordance with the invention, a preamble sequence is produced by the encoder 18 at various times (e.g., at system power-up) in order to facilitate frame boundary detection by the synchronization module 114. In the exemplary embodiment this preamble sequence includes several repetitions of selected out-of-band characters easily distinguishable from in-band characters. Again, during high-transition mode operation each out-of-band character will include less than four logical transitions, and during low-transition mode operation each out-of-band character will includes four or more logical transitions. As is discussed below, during operation in each mode several repetitions of specially selected out-of-band characters are produced by the encoder 18 during the preambling period as a means of ensuring rapid frame boundary identification within the decoder synchronization module 114. At the conclusion of the preambling period, the module 114 will "know" which of the bit lines RX[9:0] corresponds to the first bit T[0] of the 10-bit transmitted character, and will inform decoder via frame boundary pointer 118. By selecting an appropriate subset of out-of-band characters for transmission during the preambling period, the worst-case time required for synchronization to be achieved may be reduced relative to that required by conventional synchronization schemes.”

<table>
<tr><td></td><td></td><td>Shin, 14:37–62.

“Since each character is either an in-band (e.g., DATA) character or an out-of-band command or synchronization (i.e., SYNC) character, the following sets forth the possible combinations of DATA and SYNC characters among the 1st, 2nd and 3rd 10-bit characters contributing to the 21-bit window:

<table>
<tr><th>Combination</th><th>1st Character</th><th>2nd Character</th><th>3rd Character</th></tr>
<tr><td>A</td><td>SYNC</td><td>SYNC</td><td>SYNC</td></tr>
<tr><td>B</td><td>SYNC</td><td>SYNC</td><td>DATA</td></tr>
<tr><td>C</td><td>SYNC</td><td>DATA</td><td>DATA</td></tr>
<tr><td>D</td><td>DATA</td><td>DATA</td><td>DATA</td></tr>
<tr><td>E</td><td>DATA</td><td>DATA</td><td>SYNC</td></tr>
<tr><td>F</td><td>DATA</td><td>SYNC</td><td>SYNC</td></tr>
</table>
”

Shin, 17:16–32.

To the extent that Plaintiff contends that the Additional Prior Art References do not explicitly or inherently disclose this element, or render obvious this element, this element is obvious in combination with one or more references and/or with the knowledge of a person of ordinary skill in the art, as set forth in Defendant’s Invalidity Contentions.</td></tr>
<tr><td colspan="3">Claim 21</td></tr>
<tr><td>21[p]</td><td>The communications device of claim 20</td><td>The Additional Prior Art References disclose, expressly or inherently, the transmitting device of claim 20 under any proposed construction.

See exemplary disclosures and corresponding discussion for claim 20.</td></tr>
<tr><td>21</td><td>wherein the in-band symbols are transition optimized and the out-of-band synchronization</td><td>Cunningham discloses and/or renders obvious this limitation. See, e.g.:

“Control information is conveyed by using selected n-bit data block values, together with a distinctive value of 10 for the pair of label bits a and b to indicate that the associated data block represents control information rather than actual data. The data block values used for this purpose</td></tr>
</table>

	symbols are not transition optimized.	are selected to be values which inherently have a disparity of zero, such as (for 8-bit blocks) 00001111 or 10101010. It will be noted that the label bit pair value 10 likewise has zero disparity, so the complete encoded block also has zero disparity." Cunningham, 4:23–31. "There are many potential ways to achieve code block synchronization (i.e. correct alignment in the receiver of the label and payload fields). For example, one technique would be to transmit a 'comma sequence' during start-up. As an example of this technique consider a code with n=8 using the simple control payload scheme. A training control state of 1111 1111 would be transmitted as: [10 1111 1111 0000 0000][10 1111 1111 0000 0000][10 1111 1111 0000 0000] The receiver searches for the bit pair 10. This is found in only two locations: the correct location (the label field), and at the end of the run of eight 1's. However, if the receiver (incorrectly) identifies the label field to be the 10 bit pair at the end of the run of 1's, the following 16 bits are not a valid control payload (first 8 bits are not the complement of the second 8 bits). The receiver can therefore detect the mistake. It is also noted that this sequence contains a regular 010 pattern, which aids in phase-locked loop (PLL) clock acquisition." Cunningham, 7:5–23. Shin discloses and/or renders obvious this limitation. *See, e.g.*: "It may thus be appreciated that each eight-bit code converted into a 10-bit encoded character during high-transition mode operation includes four or more logical transitions. Similarly, each eight-bit code converted into a 10-bit encoded character during low-transition mode operation includes less than four logical transitions. These sets of 10-bit encoded characters capable of being produced during high-transition and low-transition modes of operation may be characterized as a high-transition set of "in-band" encoded characters, and a low-transition set of in-band encoded characters, respectively. Beyond the 256 in-band characters within the high-transition set and the 256 in-band characters of the low-transition set, there exists a high-transition set of 256 out-of-band 10-bit characters and a low-transition set of 10-bit characters. In accordance with another aspect of the invention, various synchronization and other special characters are defined using the high-

		transition and low-transition sets of out-of-band characters. Each character corresponding to one of these "out-of-band" characters associated with the high-transition set includes less than four logical transitions, and each of the out-of-band characters associated with the low-transition set of code characters includes more than four logical transitions. The difference in the number of transitions between in-band and out-of-band characters allows selected out-of-band characters to serve as control characters, and to be readily distinguished from in-band characters within the transmitted data stream." Shin, 9:49–10:8. "In accordance with one aspect of the invention, the synchronization characters associated with both the low-transition and high-transition sets of code characters are selected so as to facilitate rapid synchronization during data recovery. When the low-transition set of code characters is being employed, a special group of out-of-band characters is used during synchronization. Each special synchronization character includes a predefined number larger than four (e.g., 7) of logical transitions between binary character values, and a predefined number (e.g., two) of "non-transitions" between character values. As will be discussed below, the special synchronization characters are selected such that random logic may be used to distinguish each special synchronization character from the in-band characters of the low-transition set. The following constitutes an exemplary set of out-of-band synchronization characters for use with the low-transition set of code characters: 1100101010 1101001010 1101010010 1101010100 " Shin, 10:17–40. "As mentioned above, the decoder synchronization module 114 provides an indication to the decoder 44 of the frame boundary of each transmitted 10-bit character T[9:0]. The decoder module 114, together with the decoder switch 150 (FIG. 8), effectively function as a barrel-shifter to shuffle the parallel data RX[9:0] from the deserializer into the frame-aligned data S[9:0]. In accordance

		with the invention, a preamble sequence is produced by the encoder 18 at various times (e.g., at system power-up) in order to facilitate frame boundary detection by the synchronization module 114. In the exemplary embodiment this preamble sequence includes several repetitions of selected out-of-band characters easily distinguishable from in-band characters. Again, during high-transition mode operation each out-of-band character will include less than four logical transitions, and during low-transition mode operation each out-of-band character will includes four or more logical transitions. As is discussed below, during operation in each mode several repetitions of specially selected out-of-band characters are produced by the encoder 18 during the preambling period as a means of ensuring rapid frame boundary identification within the decoder synchronization module 114. At the conclusion of the preambling period, the module 114 will "know" which of the bit lines RX[9:0] corresponds to the first bit T[0] of the 10-bit transmitted character, and will inform decoder via frame boundary pointer 118. By selecting an appropriate subset of out-of-band characters for transmission during the preambling period, the worst-case time required for synchronization to be achieved may be reduced relative to that required by conventional synchronization schemes." Shin, 14:37–62. To the extent that Plaintiff contends that the Additional Prior Art References do not explicitly or inherently disclose this element, or render obvious this element, this element is obvious in combination with one or more references and/or with the knowledge of a person of ordinary skill in the art, as set forth in Defendant's Invalidity Contentions.
Claim 22		
22[p]	The communications device of claim 19	The Additional Prior Art References disclose, expressly or inherently, the transmitting device of claim 19 under any proposed construction. *See* exemplary disclosures and corresponding discussion for claim 19.
22	wherein the synchronization symbol is transmitted before	Chen discloses and/or renders obvious this limitation. *See, e.g.*:

	transmitting the symbols of the packet.	"FIG. 5 illustrates the processing flow of the Insertion Processing Unit of FIG. 4 in accordance with the present invention. Note that the steps of FIG. 5 are shown as occurring in a sequential manner for clarity, but many of the steps may occur concurrently and/or sequentially. At block 502, the IPU outputs the main transport stream from the main stream buffer (MSB) 480. The IPU is essentially in a bypass mode at this time since the insertion stream has not yet been inserted. At block 506, the main stream parser parses the main stream to locate the splicing signal T-- in. Note that T-- in need not be carried in the main stream, but may be provided locally, for example, at a cable system headend. When T-- in is detected, the IPU continues to output the main transport stream from the main stream buffer. At block 508, the main stream parser parses the main stream to locate the closest packet with an anchor picture (e.g., I or P picture) start code and passes that packet to the syntax processor 470. This packet will be the last packet of the main stream. To ensure that the resulting data stream is MPEG compliant, no data from the next frame or sequence of frames may be included in this last main stream packet. At block 512, the syntax processor adds a sequence end code field to the last packet of the main stream. The sequence end code is a twenty-four bit code which indicates the end of a sequence of pictures. At block 514, the PSI/PID replacer 435 starts to replace the PSI/PID of the insertion stream." Chen, 12:28–53.

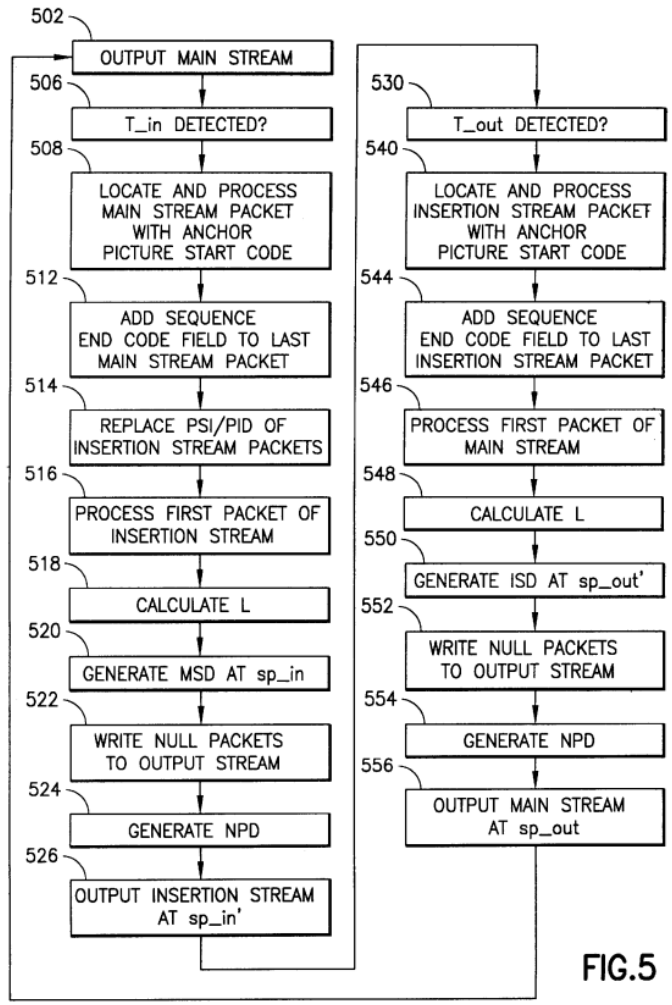
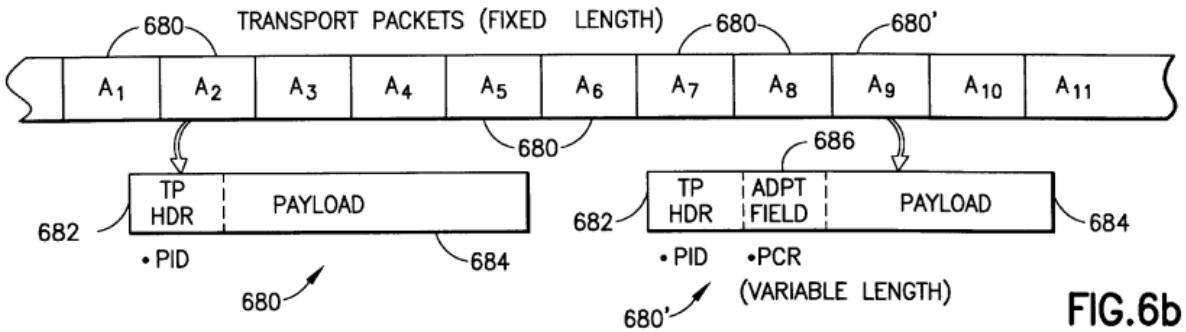


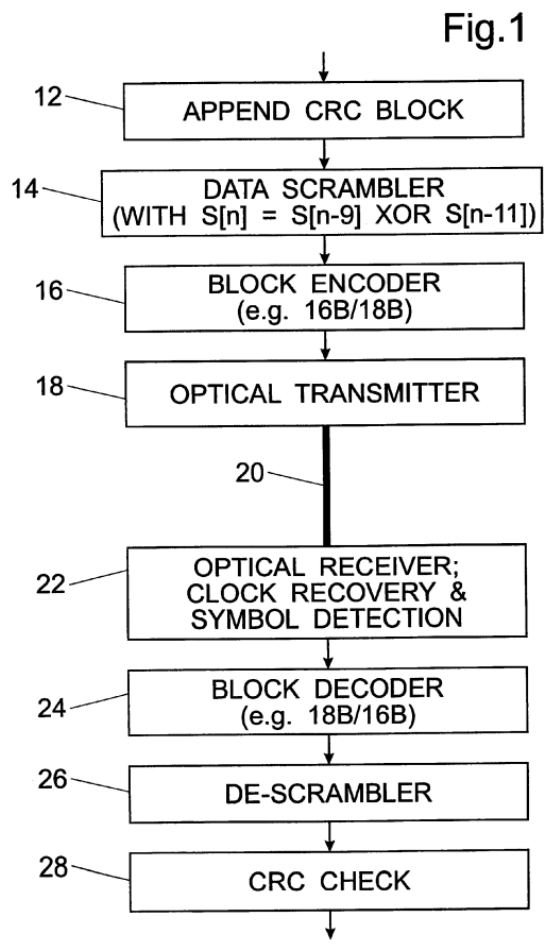
FIG.5

Chen, Fig. 5.

"In the MPEG-2 system (and the similar DigiCipher® II system proprietary to General Instrument Corporation, the assignee hereof) a transport stream, or transport multiplex is made up of a

		contiguous set of fixed length packets. Each packet is 188 total bytes in length, with the first four of those bytes being defined as the packet header. The payload portion of each packet is thus normally 184 bytes. However, a variable length adaptation field may be provided to extend the header, when required. When an adaptation field is present, the payload portion of the packet will be correspondingly shorter. Various timing and identification information is provided in different portions of the transport stream. These include a packet identifier (PID) found in the transport header of each transport packet to provide a reference number for identifying the transport packets carrying a specific service component. This number is included in a service definition or "service map" used by the receiver to identify those transport packets required to reconstruct a television program signal. The PID may also be referenced for various grooming and remultiplexing functions. In the case of video, audio or isochronous control data, the stream of packets labeled with a single PID represents a single video, audio or isochronous data service elementary stream, respectively. Each type of packet will have a different PID identifying the packet type. Timing information carried by the transport stream includes a program clock reference (PCR) which effectively represents a sample of the system time clock (STC) time base that underlies the service composed of the PIDs referenced in the service map. The PID carrying the packet with the PCR is also referenced in the service map. The video, audio and isochronous data components of a service are locked through a defined relationship to the system time clock. The PCR serves to define the transport rate, in the sense that between any two successive PCRs in one PID, the transport rate is constant and nominally equal to the system time clock rate times the ratio of the total number of transport bytes between the PCRs divided by the difference in the PCRs in units of system time clock ticks." Chen, 14:45–15:17. "As indicated in FIG. 6b, the transport header of each transport packet includes the PID which identifies the particular service component carried by the transport packet. The PCR will be carried by an adaptation field to provide timing information for a desired service. At a receiver, the PCR for the desired service is detected from the adaptation field. The PIDs of the transport packets will

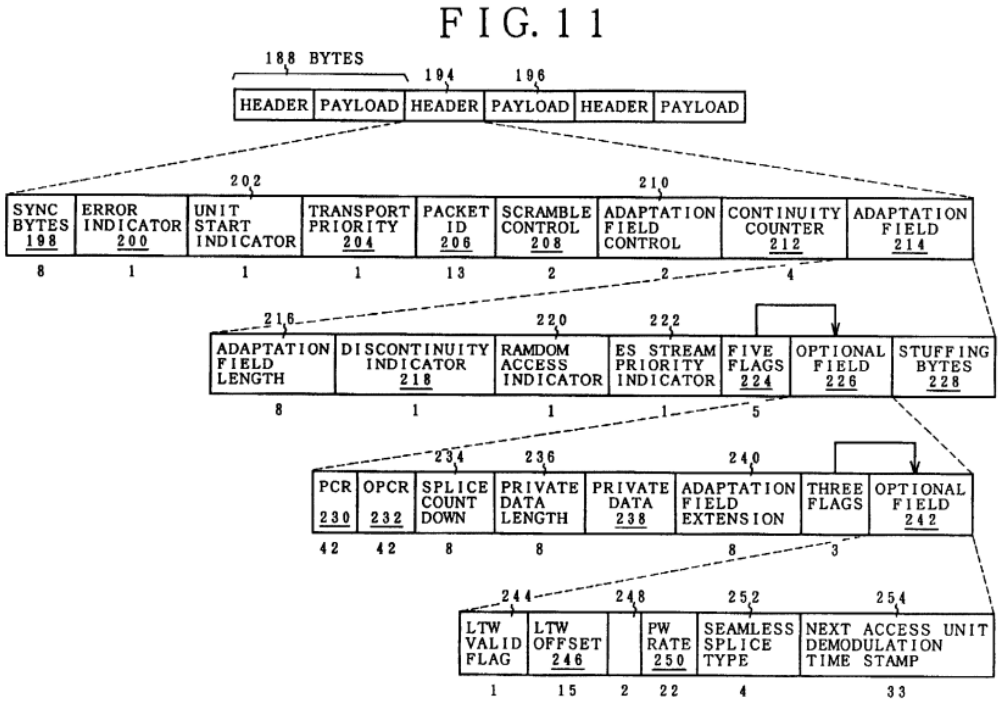
		then be monitored in accordance with the timing established by the PCR to recover those transport packets carrying a particular component of the service to be processed." Chen, 16:41–50.  Chen, Fig. 6b. Cunningham discloses and/or renders obvious this limitation. *See, e.g.*: "Referring to FIG. 1, a data frame intended to be communicated over an optical fibre channel 20 typically comprises a binary digital message (payload) to be transferred between stations on the network; this message may comprise user data, or control information for managing operation of the network and its components. A circuit 12 generates an associated cyclic redundancy check (CRC) block containing check data derived from this message in known manner in accordance with a predetermined CRC algorithm, and appends the CRC block to the message." Cunningham, 2:65–3:7.

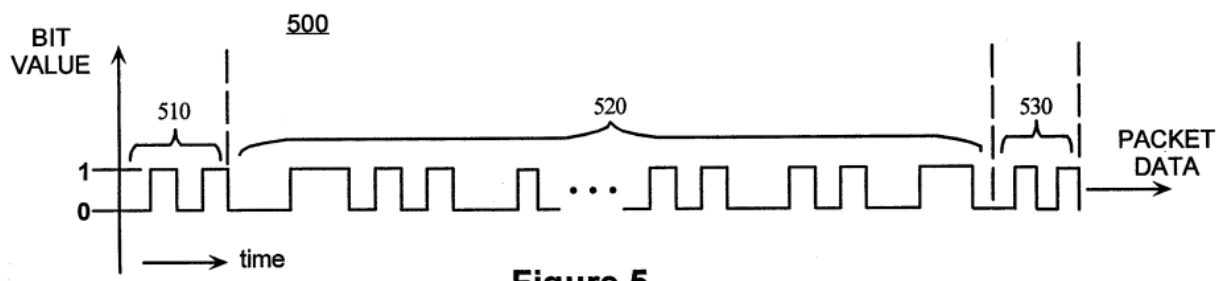


Cunningham, Fig. 1.

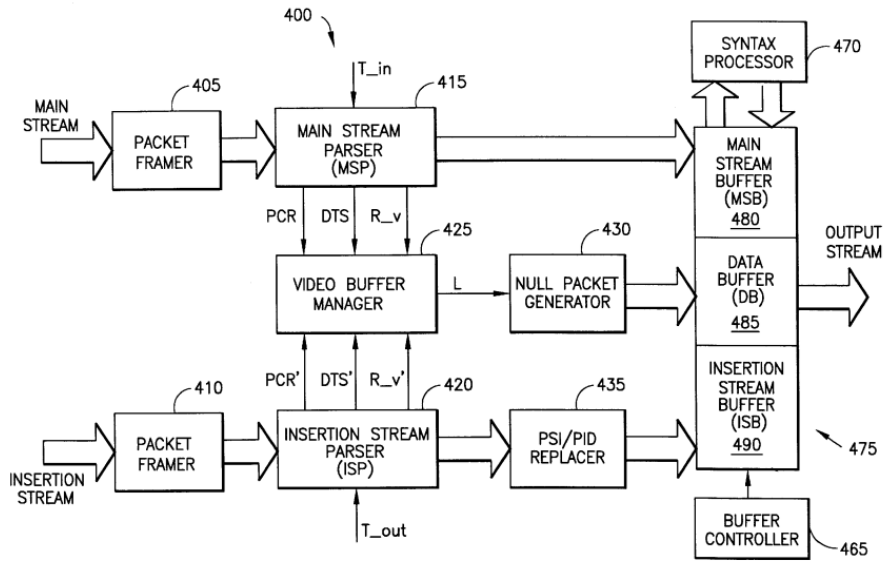
"Control information is conveyed by using selected n-bit data block values, together with a distinctive value of 10 for the pair of label bits a and b to indicate that the associated data block represents control information rather than actual data. The data block values used for this purpose are selected to be values which inherently have a disparity of zero, such as (for 8-bit blocks)

		00001111 or 10101010. It will be noted that the label bit pair value 10 likewise has zero disparity, so the complete encoded block also has zero disparity." Cunningham, 4:23–31. "The encoded blocks are transmitted over the optical fibre channel 20 (leftmost symbol first), and upon receipt at their destination they are detected by a receiver 22 where compensation is made in known manner for the effects of timing drift and other signal degradation to produce a restored digital data stream." Cunningham, 4:45–50. "There are many potential ways to achieve code block synchronization (i.e. correct alignment in the receiver of the label and payload fields). For example, one technique would be to transmit a 'comma sequence' during start-up. As an example of this technique consider a code with n=8 using the simple control payload scheme. A training control state of 1111 1111 would be transmitted as: [10 1111 1111 0000 0000][10 1111 1111 0000 0000][10 1111 1111 0000 0000] The receiver searches for the bit pair 10. This is found in only two locations: the correct location (the label field), and at the end of the run of eight 1's. However, if the receiver (incorrectly) identifies the label field to be the 10 bit pair at the end of the run of 1's, the following 16 bits are not a valid control payload (first 8 bits are not the complement of the second 8 bits). The receiver can therefore detect the mistake. It is also noted that this sequence contains a regular 010 pattern, which aids in phase-locked loop (PLL) clock acquisition." Cunningham, 7:5–23. Hiroshima discloses and/or renders obvious this limitation. *See, e.g.*: "FIG. 11 shows the details of the MPEG2-TS. Each packet of the MPEG2-TS has a prefix length of 188 bytes. The header 194 consists of four bytes and the payload 196 consists of 184 bytes. The header 194 is constructed by sync bytes 198, an error indicator 200, a unit start indicator 202, a transport priority 204, a packet ID 206, a scramble control 208, an adaptation field control 210, a

		continuity counter 212, and an adaptation field 214. The adaptation field 214 is constructed by an adaptation field length 216, a discontinuity indicator 218, a random access indicator 220, an elementary stream priority indicator 222, a 5-flag 224, an optional field 226, and stuffing bytes 228. Among them, the optional field 226 is constructed by a program clock reference 230, an original program clock reference field 232, a splice count down 234, a private data length 236, private data 238, an adaptation field extension 240, 3-flag 241, and an optional field 242." Hiroshima, 10:51–67.  Hiroshima, Fig. 11. Haartsen discloses and/or renders obvious this limitation. *See, e.g.*:

		"To alleviate this problem, the present invention discloses that a digital preamble can be strategically divided into three or more parts so that synchronization and other information can be included within the preamble while still enabling accurate DC estimation. Specifically, a short leading DC-free part of the preamble (e.g., alternating logical l 's and 0's, or any other sequence including an equal number of logical l's and logical 0's) can be used to provide initial, coarse DC offset estimation. Thereafter, a synchronization part of the preamble, which is not necessarily DC free, but which is protected by code to prevent detection errors, can be used not only to synchronize detection of a data packet, but also to carry timing and/or other useful information (e.g., channel identification, destination address, time of day, user identification, etc.)." Haartsen, 12:13–13:2. "An exemplary multi-part preamble according to the invention is depicted in Figure 5. In the figure, a three-part preamble 500 is shown to include a DC-free leading sequence 510, followed by a multiple-bit synchronization word 520 which is not necessarily DC free and which can be different for each data packet. The synchronization word 520, which can include timing and/or other information and which is coded to protect against detection errors, is in turn followed by a trailing DC-free sequence 530. Although the leading and trailing DC-free sequences 510, 530 are shown in the figure as 4-bit sequences of alternating logical l 's and 0's, those of ordinary skill will appreciate that each DC-free sequence 510, 530 can, in practice, be shorter or longer in length and can include any DC-free sequence having an equal number of logical l's and logical 0's." Haartsen, 13:13–14:2.  **Figure 5**

		Haartsen, Fig. 5. "When the exemplary preamble 500 is received by the system 300 of Figure 3, the leading DC-free sequence 510 induces a coarse estimate of the baseband DC offset on the storage capacitor 340. Although the estimate is coarse, it is sufficient to permit successful detection of the coded synchronization word 520 (e.g., by the correlation processor 310). Methods for coding of a synchronization word to protect against erroneous detection in the presence of some DC-offset are known to those of skill in the art. Once the correlation processor 310 has detected the synchronization word 520, timing between received bits is well known. Consequently, the correlation processor 310 can open the switch 330 precisely at the end of the trailing DC-free sequence 530 to thereby store an accurate estimate of the DC offset for reception and detection of the incoming digital data packet." Haartsen, 14:3–14. To the extent that Plaintiff contends that the Additional Prior Art References do not explicitly or inherently disclose this element, or render obvious this element, this element is obvious in combination with one or more references and/or with the knowledge of a person of ordinary skill in the art, as set forth in Defendant's Invalidity Contentions.
Claim 23		
23[p]	The communications device of claim 19	The Additional Prior Art References disclose, expressly or inherently, the transmitting device of claim 19 under any proposed construction. *See* exemplary disclosures and corresponding discussion for claim 19.
23	wherein the packet has a header with a field that indicates packet type and the identification component checks the field	Chen discloses and/or renders obvious this limitation. *See, e.g.*: "FIG. 4 is a block diagram of an Insertion Processing Unit (IPU) of a Digital Ad Insertion Module in accordance with the present invention. The IPU, shown generally at 400, receives the main stream at a packet framer 405, while the insertion stream is received at another packet framer 410. After processing by the packet framer 405, the main stream is provided to a Main Stream Parser (MSP) 415."

	of the header that indicates packet type.	Chen, 5:62–6:1.  FIG.4 Chen, Fig. 4. "The main stream parser 415 provides the main stream data to a main stream buffer 480, which communicates with a syntax processor 470. The main stream buffer 480 is part of the output buffer 475. The insertion stream parser 420 provides the insertion stream data to a Program Specific Information (PSI)/Program Identifier (PID) Replacer 435, which replaces the PSI tables and PIDs of the insertion stream with those of the main stream. The PSI/PID replacer 435 retrieves the relevant information from the main stream using a communication path which is not shown. The PSI and PID data of the main stream provide the information to connect the various audio, video, and data packets to a particular programming service. Specifically, this is accomplished using a Program Association Table (PAT) and a Program Map Table (PMT). The insertion stream is then provided to an Insertion Stream Buffer (ISB) 490, which is part of the output buffer 475. The ISB 490 communicates with a syntax processor 470. Furthermore, the output buffer 475 receives

		commands from a buffer controller 465 to provide an output stream with the insertion stream seamlessly spliced into the main stream." Chen, 6:33–54. "FIG. 5 illustrates the processing flow of the Insertion Processing Unit of FIG. 4 in accordance with the present invention. Note that the steps of FIG. 5 are shown as occurring in a sequential manner for clarity, but many of the steps may occur concurrently and/or sequentially. At block 502, the IPU outputs the main transport stream from the main stream buffer (MSB) 480. The IPU is essentially in a bypass mode at this time since the insertion stream has not yet been inserted. At block 506, the main stream parser parses the main stream to locate the splicing signal T-- in. Note that T-- in need not be carried in the main stream, but may be provided locally, for example, at a cable system headend. When T-- in is detected, the IPU continues to output the main transport stream from the main stream buffer. At block 508, the main stream parser parses the main stream to locate the closest packet with an anchor picture (e.g., I or P picture) start code and passes that packet to the syntax processor 470. This packet will be the last packet of the main stream. To ensure that the resulting data stream is MPEG compliant, no data from the next frame or sequence of frames may be included in this last main stream packet. At block 512, the syntax processor adds a sequence end code field to the last packet of the main stream. The sequence end code is a twenty-four bit code which indicates the end of a sequence of pictures. At block 514, the PSI/PID replacer 435 starts to replace the PSI/PID of the insertion stream." Chen, 12:28–53.

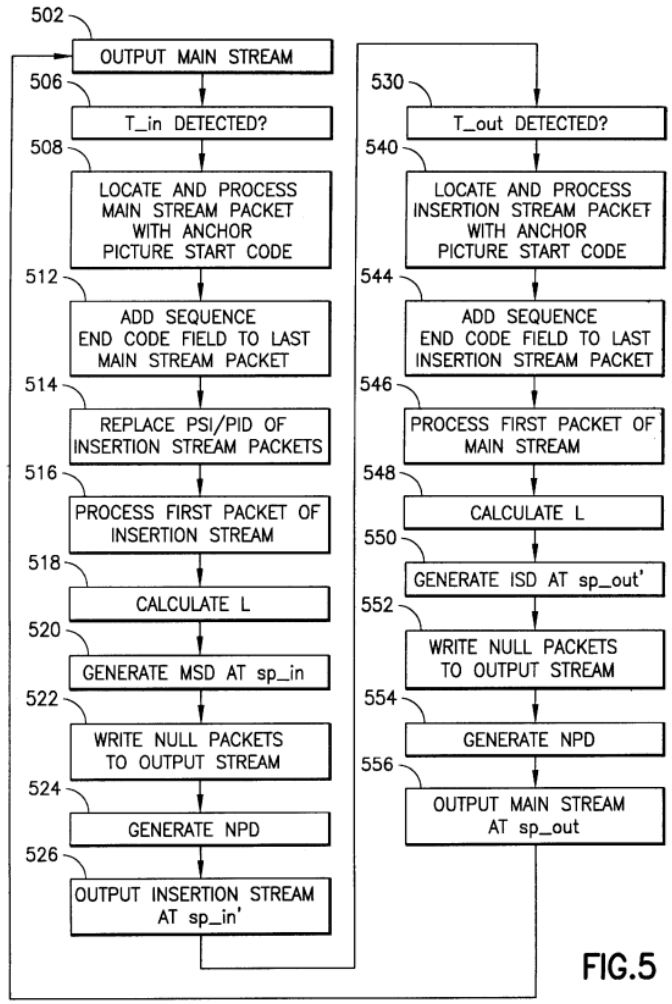
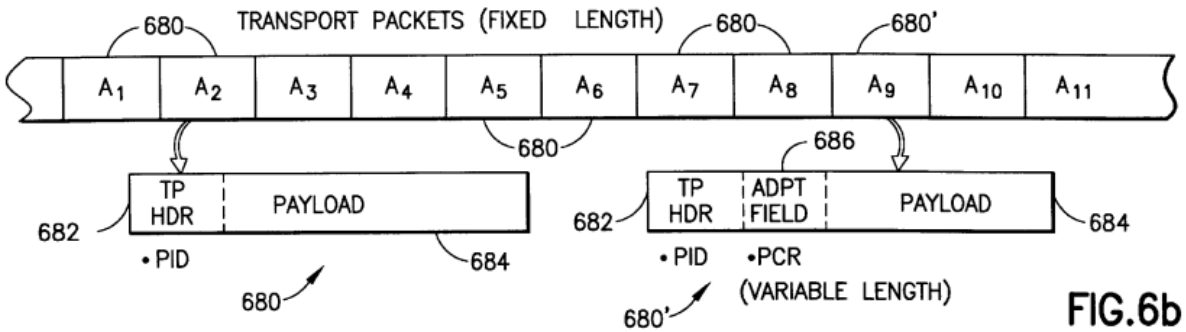


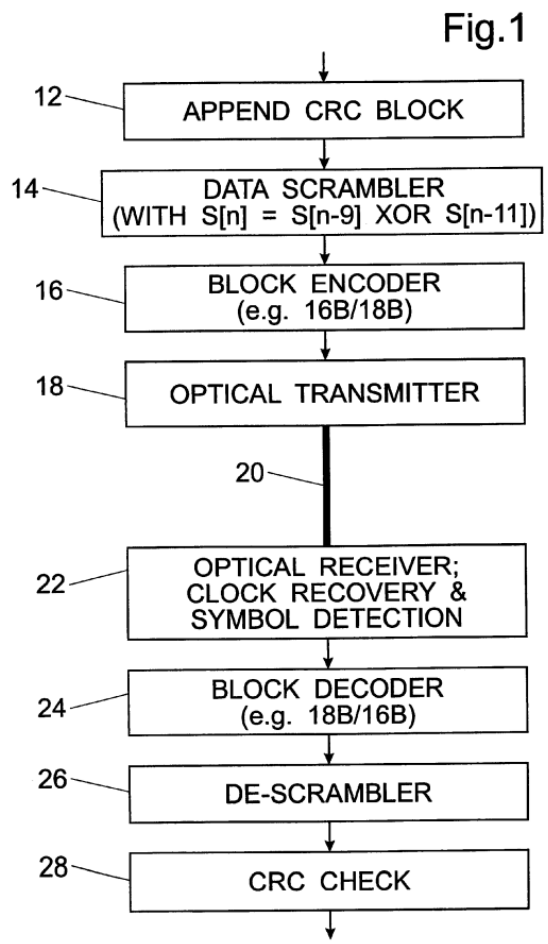
FIG.5

Chen, Fig. 5.

"In the MPEG-2 system (and the similar DigiCipher® II system proprietary to General Instrument Corporation, the assignee hereof) a transport stream, or transport multiplex is made up of a

		contiguous set of fixed length packets. Each packet is 188 total bytes in length, with the first four of those bytes being defined as the packet header. The payload portion of each packet is thus normally 184 bytes. However, a variable length adaptation field may be provided to extend the header, when required. When an adaptation field is present, the payload portion of the packet will be correspondingly shorter. Various timing and identification information is provided in different portions of the transport stream. These include a packet identifier (PID) found in the transport header of each transport packet to provide a reference number for identifying the transport packets carrying a specific service component. This number is included in a service definition or "service map" used by the receiver to identify those transport packets required to reconstruct a television program signal. The PID may also be referenced for various grooming and remultiplexing functions. In the case of video, audio or isochronous control data, the stream of packets labeled with a single PID represents a single video, audio or isochronous data service elementary stream, respectively. Each type of packet will have a different PID identifying the packet type. Timing information carried by the transport stream includes a program clock reference (PCR) which effectively represents a sample of the system time clock (STC) time base that underlies the service composed of the PIDs referenced in the service map. The PID carrying the packet with the PCR is also referenced in the service map. The video, audio and isochronous data components of a service are locked through a defined relationship to the system time clock. The PCR serves to define the transport rate, in the sense that between any two successive PCRs in one PID, the transport rate is constant and nominally equal to the system time clock rate times the ratio of the total number of transport bytes between the PCRs divided by the difference in the PCRs in units of system time clock ticks." Chen, 14:45–15:17. "As indicated in FIG. 6b, the transport header of each transport packet includes the PID which identifies the particular service component carried by the transport packet. The PCR will be carried by an adaptation field to provide timing information for a desired service. At a receiver, the PCR for the desired service is detected from the adaptation field. The PIDs of the transport packets will

		then be monitored in accordance with the timing established by the PCR to recover those transport packets carrying a particular component of the service to be processed." Chen, 16:41–50.  Chen, Fig. 6b. Cunningham discloses and/or renders obvious this limitation. *See, e.g.*: "Referring to FIG. 1, a data frame intended to be communicated over an optical fibre channel 20 typically comprises a binary digital message (payload) to be transferred between stations on the network; this message may comprise user data, or control information for managing operation of the network and its components. A circuit 12 generates an associated cyclic redundancy check (CRC) block containing check data derived from this message in known manner in accordance with a predetermined CRC algorithm, and appends the CRC block to the message." Cunningham, 2:65–3:7.



Cunningham, Fig. 1.

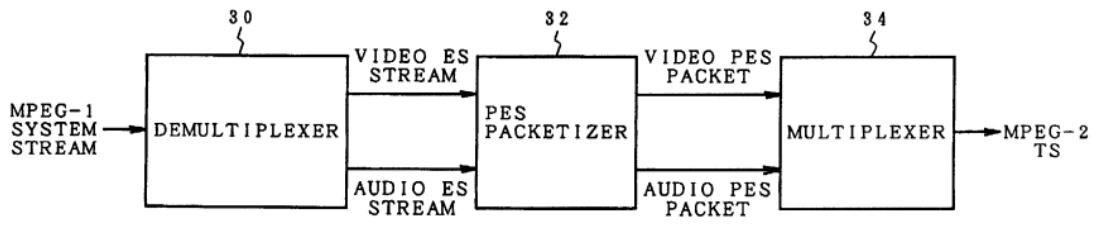
"Control information is conveyed by using selected n-bit data block values, together with a distinctive value of 10 for the pair of label bits a and b to indicate that the associated data block represents control information rather than actual data. The data block values used for this purpose are selected to be values which inherently have a disparity of zero, such as (for 8-bit blocks)

		00001111 or 10101010. It will be noted that the label bit pair value 10 likewise has zero disparity, so the complete encoded block also has zero disparity." Cunningham, 4:23–31. "The encoded blocks are transmitted over the optical fibre channel 20 (leftmost symbol first), and upon receipt at their destination they are detected by a receiver 22 where compensation is made in known manner for the effects of timing drift and other signal degradation to produce a restored digital data stream." Cunningham, 4:45–50. "There are many potential ways to achieve code block synchronization (i.e. correct alignment in the receiver of the label and payload fields). For example, one technique would be to transmit a 'comma sequence' during start-up. As an example of this technique consider a code with n=8 using the simple control payload scheme. A training control state of 1111 1111 would be transmitted as: [10 1111 1111 0000 0000][10 1111 1111 0000 0000][10 1111 1111 0000 0000] The receiver searches for the bit pair 10. This is found in only two locations: the correct location (the label field), and at the end of the run of eight 1's. However, if the receiver (incorrectly) identifies the label field to be the 10 bit pair at the end of the run of 1's, the following 16 bits are not a valid control payload (first 8 bits are not the complement of the second 8 bits). The receiver can therefore detect the mistake. It is also noted that this sequence contains a regular 010 pattern, which aids in phase-locked loop (PLL) clock acquisition." Cunningham, 7:5–23. Hiroshima discloses and/or renders obvious this limitation. *See, e.g*: "FIG. 4 is a block diagram of the first embodiment of the invention in which the moving picture stream conversion in mode 1 is executed. A moving picture stream converting apparatus of FIG. 4 comprises a demultiplexer 30, a packetizer 32, and a multiplexer 34. The demultiplexer 30 receives an MPEG1 system stream as a conversion source and outputs elementary streams which were

separated into video and audio streams. The packetizer 32 forms packets constructing a packetized elementary stream of MPEG2 with respect to each of the video and audio elementary streams which are supplied from the demultiplexer 30. The packet is usually called a PES packet. The multiplexer 34 divides each of the video and audio PES packets into a TS packet having a fixed length of 188 bytes and subsequently multiplexes, and outputs the resultant data as an MPEG2-TS."

Hiroshima, 7:59–8:7.

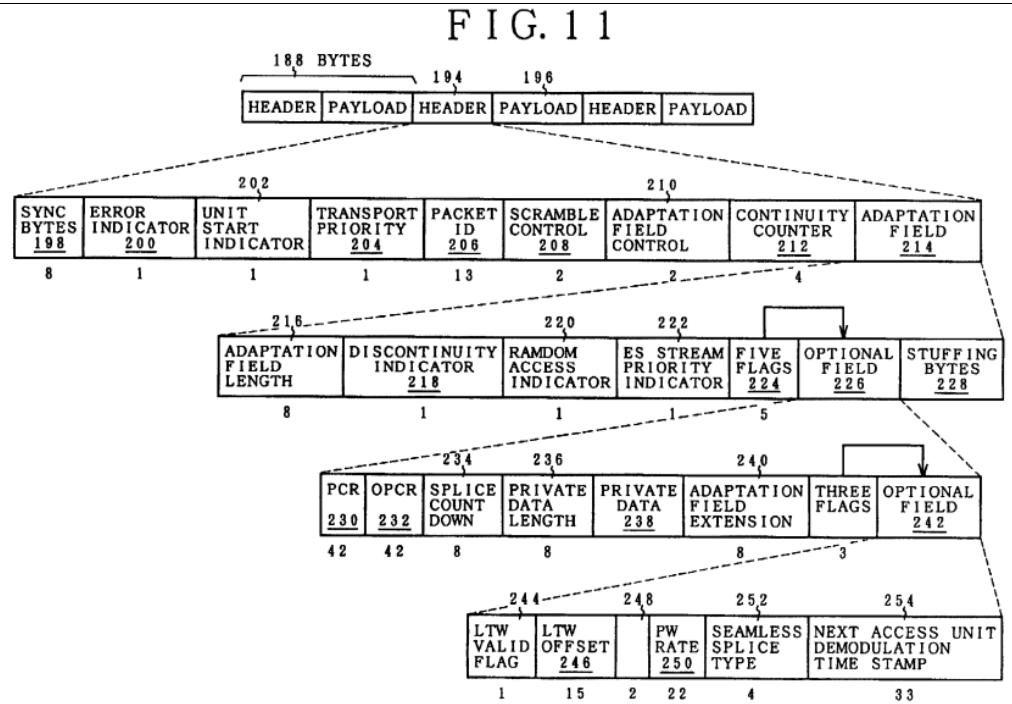
FIG. 4



Hiroshima, Fig. 4.

"FIG. 11 shows the details of the MPEG2-TS. Each packet of the MPEG2-TS has a prefix length of 188 bytes. The header 194 consists of four bytes and the payload 196 consists of 184 bytes. The header 194 is constructed by sync bytes 198, an error indicator 200, a unit start indicator 202, a transport priority 204, a packet ID 206, a scramble control 208, an adaptation field control 210, a continuity counter 212, and an adaptation field 214. The adaptation field 214 is constructed by an adaptation field length 216, a discontinuity indicator 218, a random access indicator 220, an elementary stream priority indicator 222, a 5-flag 224, an optional field 226, and stuffing bytes 228. Among them, the optional field 226 is constructed by a program clock reference 230, an original program clock reference field 232, a splice count down 234, a private data length 236, private data 238, an adaptation field extension 240, 3-flag 241, and an optional field 242."

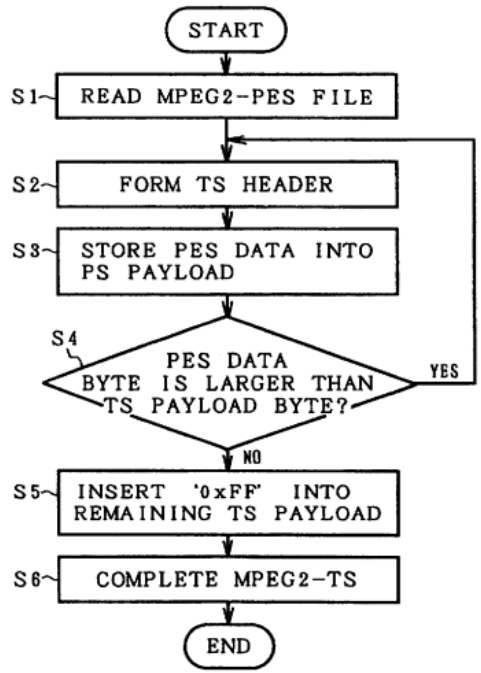
Hiroshima, 10:51–67.



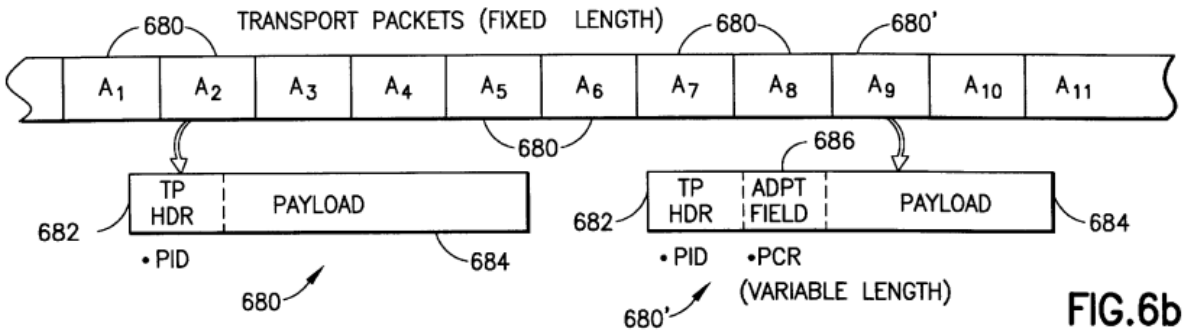
Hiroshima, Fig. 11.

"FIG. 15 shows a process for forming the MPEG2-TS from the PES packet 82 by a multiplexing process in the multiplexer 34 in FIG. 5. First in step S1, the MPEG2-PES file is read. The TS header 194 in FIG. 11 is formed in step S2. The TS header 194 is formed in a manner such that the parameters such as error indicator 200 indicative of the presence or absence of the bit error in the packet, unit start indicator 202 indicating whether a new PES packet appears in the TS payload or not, packet ID 206 indicating whether the packet in the payload is video or audio, adaptation field 214, and adaptation field control 210 indicative of the presence or absence of the payload 196 are set in accordance with the state of the payload storage data of the 184-byte unit divided from the MPEG2-PES packet which had been read in step S1."

Hiroshima, 11:13–27.

		FIG. 15  Hiroshima, Fig. 15. To the extent that Plaintiff contends that the Additional Prior Art References do not explicitly or inherently disclose this element, or render obvious this element, this element is obvious in combination with one or more references and/or with the knowledge of a person of ordinary skill in the art, as set forth in Defendant's Invalidity Contentions.

Claim 24		
24[p]	The communications device of claim 19	Chen discloses, expressly or inherently, the transmitting device of claim 19 under any proposed construction. *See* exemplary disclosures and corresponding discussion for claim 19.
24	wherein the packet types include a data packet.	Chen discloses and/or renders obvious this limitation. *See, e.g.*: "Various timing and identification information is provided in different portions of the transport stream. These include a packet identifier (PID) found in the transport header of each transport packet to provide a reference number for identifying the transport packets carrying a specific service component. This number is included in a service definition or "service map" used by the receiver to identify those transport packets required to reconstruct a television program signal. The PID may also be referenced for various grooming and remultiplexing functions. In the case of video, audio or isochronous control data, the stream of packets labeled with a single PID represents a single video, audio or isochronous data service elementary stream, respectively. Each type of packet will have a different PID identifying the packet type." Chen, 14:56–15:3. "The PES payload contains the information data that is desired to be transmitted to a receiver. Thus, the payload includes all of the video, audio and control information necessary for the receiver to decode and reconstruct, e.g., a digital television signal). In order to meet the requirements of robustness and simplicity, a fixed packet length approach is preferred to the variable length PES packets. Thus, as illustrated in FIG. 6b, the packet elementary stream containing the PES packets 670 is reconfigured into a stream of fixed length transport packets 680. The transport packets illustrated in FIG. 6b all correspond to the same service component, such as the video component of a digital television transmission. In the MPEG-2 and DigiCipher® II embodiments, each packet is 188 total bytes in length, with the first four bytes comprising a transport packet header (TP HDR) 682. The payload portion 684 of each packet 680 is thus normally 184 bytes. However, an adaptation field mechanism is present, as illustrated by transport packet 680', to extend the header when required. The adaptation field 686 provides

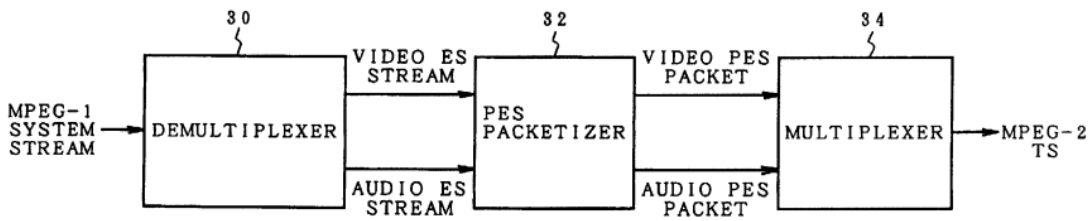
		additional information which is not required for every transport packet. The adaptation field (ADPT FIELD) 686 extends the regular transport header 682 at the expense of payload 684, which will be less than 184 bytes whenever the adaptation is provided. The adaptation field 686 is of variable length, depending on the information it contains. Typically, the adaptation field will support additional information for time base recovery and other functions, and also provides a mechanism for padding the payload when it does not occupy the full 184 bytes. Such padding can be used, for example, to make a variable rate video packetized elementary stream into a constant rate transport stream." Chen, 16:9–40.  Chen, Fig. 6b. Cunningham discloses and/or renders obvious this limitation. *See, e.g.*: "Control information is conveyed by using selected n-bit data block values, together with a distinctive value of 10 for the pair of label bits a and b to indicate that the associated data block represents control information rather than actual data. The data block values used for this purpose are selected to be values which inherently have a disparity of zero, such as (for 8-bit blocks) 00001111 or 10101010. It will be noted that the label bit pair value 10 likewise has zero disparity, so the complete encoded block also has zero disparity." Cunningham, 4:23–31.

Hiroshima discloses and/or renders obvious this limitation. *See, e.g*:

"FIG. 4 is a block diagram of the first embodiment of the invention in which the moving picture stream conversion in mode 1 is executed. A moving picture stream converting apparatus of FIG. 4 comprises a demultiplexer 30, a packetizer 32, and a multiplexer 34. The demultiplexer 30 receives an MPEG1 system stream as a conversion source and outputs elementary streams which were separated into video and audio streams. The packetizer 32 forms packets constructing a packetized elementary stream of MPEG2 with respect to each of the video and audio elementary streams which are supplied from the demultiplexer 30. The packet is usually called a PES packet. The multiplexer 34 divides each of the video and audio PES packets into a TS packet having a fixed length of 188 bytes and subsequently multiplexes, and outputs the resultant data as an MPEG2-TS."

Hiroshima, 7:59–8:7.

FIG. 4



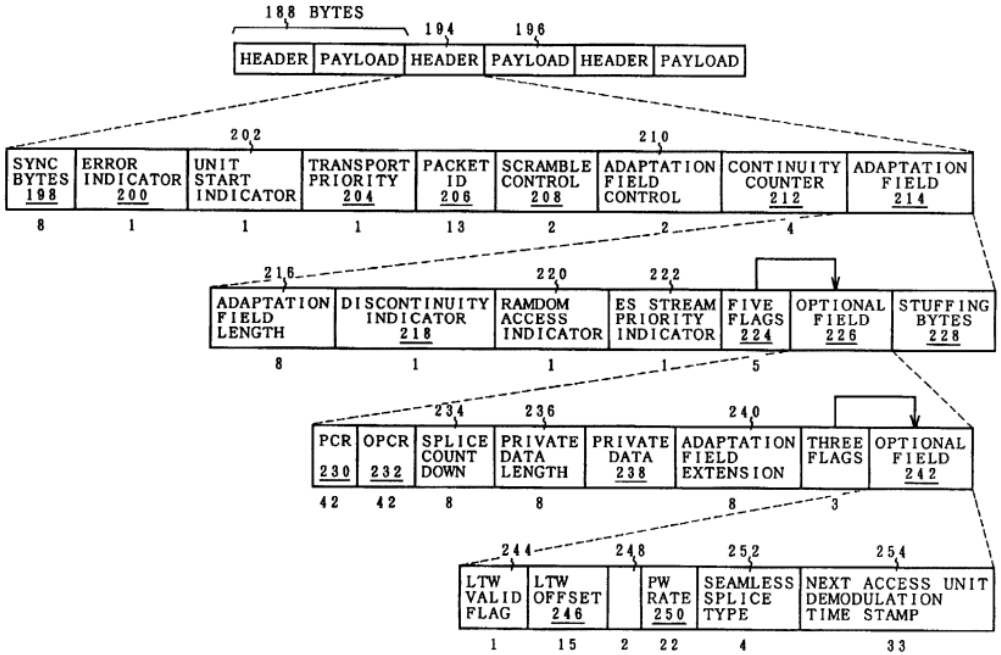
Hiroshima, Fig. 4.

"FIG. 11 shows the details of the MPEG2-TS. Each packet of the MPEG2-TS has a prefix length of 188 bytes. The header 194 consists of four bytes and the payload 196 consists of 184 bytes. The header 194 is constructed by sync bytes 198, an error indicator 200, a unit start indicator 202, a transport priority 204, a packet ID 206, a scramble control 208, an adaptation field control 210, a continuity counter 212, and an adaptation field 214. The adaptation field 214 is constructed by an adaptation field length 216, a discontinuity indicator 218, a random access indicator 220, an

elementary stream priority indicator 222, a 5-flag 224, an optional field 226, and stuffing bytes 228. Among them, the optional field 226 is constructed by a program clock reference 230, an original program clock reference field 232, a splice count down 234, a private data length 236, private data 238, an adaptation field extension 240, 3-flag 241, and an optional field 242."

Hiroshima, 10:51–67.

FIG. 11



Hiroshima, Fig. 11.

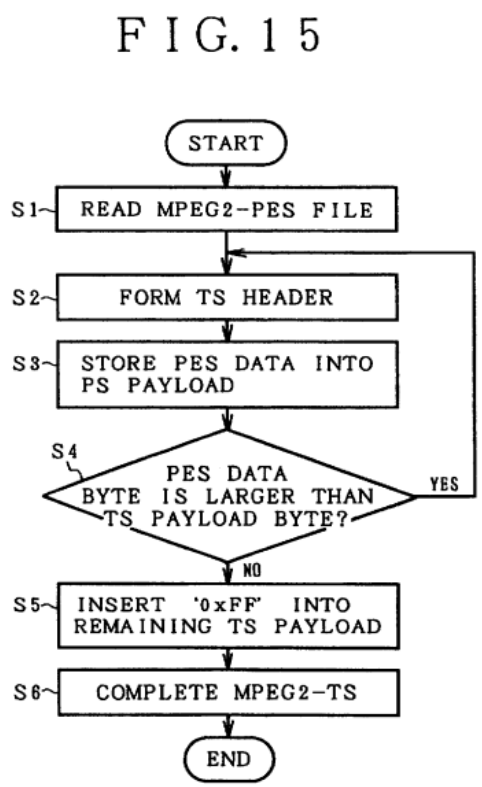
FIG. 12

NO	SYNTAX	BITS	DAVID	EXPLANATION/REMARKS
1	SYSC BYTE	8	'0x47' (PREFIX)	SYNC BYTES
2	TRANSPORT ERROR IND.	1	'0' :NONE (PREFIX)	PRESENCE OR ABSENCE OF BIT ERROR IN PACKET
3	PAYLOAD UNIT START IND.	1	'1' / '0'	WHETHER NEW PES PACKET APPEARS IN THIS TS PAYLOAD OR NOT?
4	TRANSPORT PRIORITY	1	'0' :LOW (PREFIX)	PRIORITY OF THIS TS PACKET IN THE SAME PID
5	PID	13	'0x21' :VIDEO '0x20' :AUDIO (SPECIFIED TO TS PARA. FIRST)	'0x00' :PAT '0x01' :CAT '0x02' ~ '0x0f' :RESERVATION '0x1fff' :NULL PACKET
6	TRANSPORT SCRBL CTRL	2	'00' :OFF (PREFIX)	SCRAMBLE ON/OFF
7	ADAPTATION FIELD CTRL	2	'11' '01'	PRESENCE OR ABSENCE OF ADAPTATION FIELD/PAYLOAD '10' / '11' :ADAPTATION FIELD EXISTS '01' / '11' :PAYLOAD EXISTS '00' :IGNORED (NULL PACKET: '01')
8	CONTINUITY COUNTER	4	'0000' (PREFIX)	INCREMENT (CONTINUITY COUNTER)

Hiroshima, Fig. 12.

"FIG. 15 shows a process for forming the MPEG2-TS from the PES packet 82 by a multiplexing process in the multiplexer 34 in FIG. 5. First in step S1, the MPEG2-PES file is read. The TS header 194 in FIG. 11 is formed in step S2. The TS header 194 is formed in a manner such that the parameters such as error indicator 200 indicative of the presence or absence of the bit error in the packet, unit start indicator 202 indicating whether a new PES packet appears in the TS payload or not, packet ID 206 indicating whether the packet in the payload is video or audio, adaptation field 214, and adaptation field control 210 indicative of the presence or absence of the payload 196 are set in accordance with the state of the payload storage data of the 184-byte unit divided from the MPEG2-PES packet which had been read in step S1."

Hiroshima, 11:13–27.



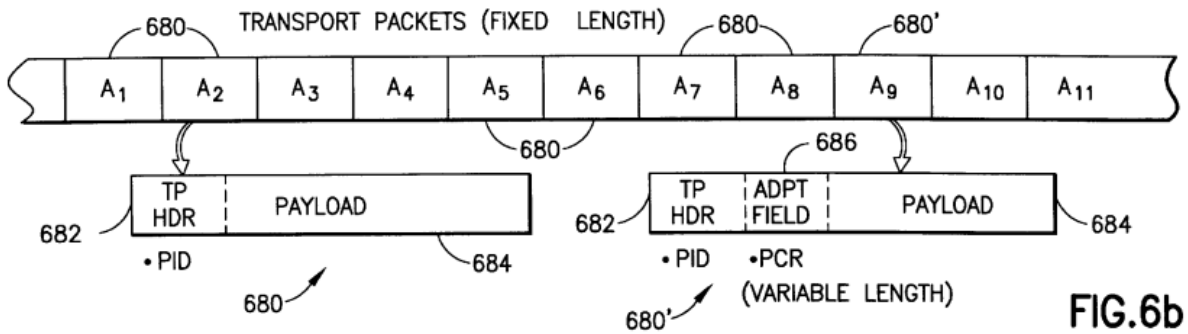
Hiroshima, Fig. 15.

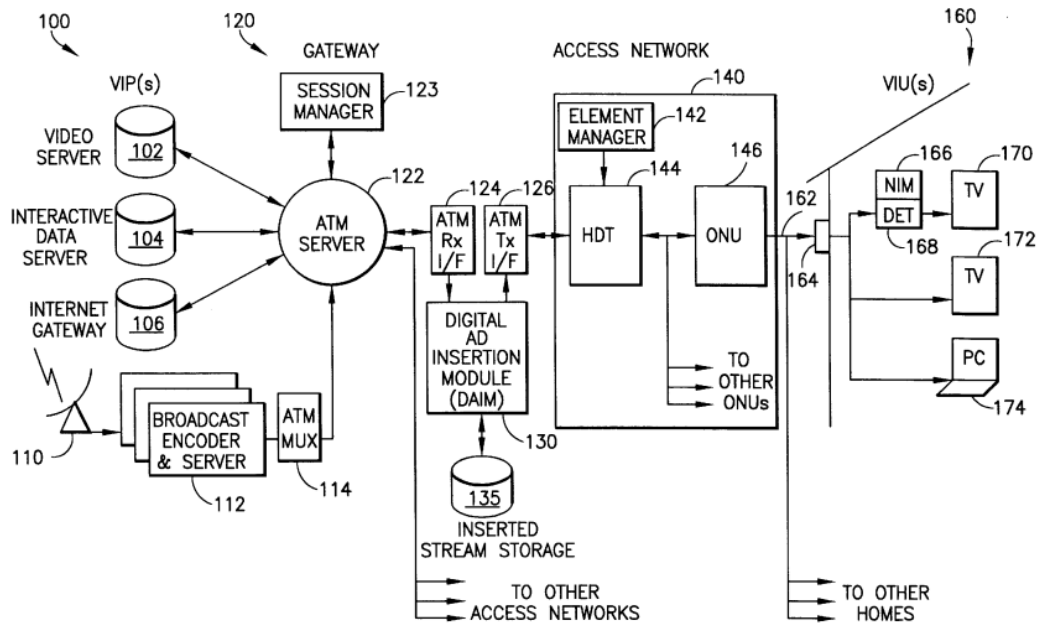
Haartsen discloses and/or renders obvious this limitation. *See, e.g.*:

"The present invention fulfills the above-described and other needs by providing a multi-part digital preamble for use in transmitting digital data packets. Advantageously, a preamble according to the invention substantially reduces the overall length of DC-free sequences that are transmitted with each data packet in a digital FM system. According to exemplary embodiments, a multi-part preamble includes a short, substantially DC-free leading part which enables coarse estimation of a prevailing DC level at an FM receiver. The exemplary preamble also includes a synchronization

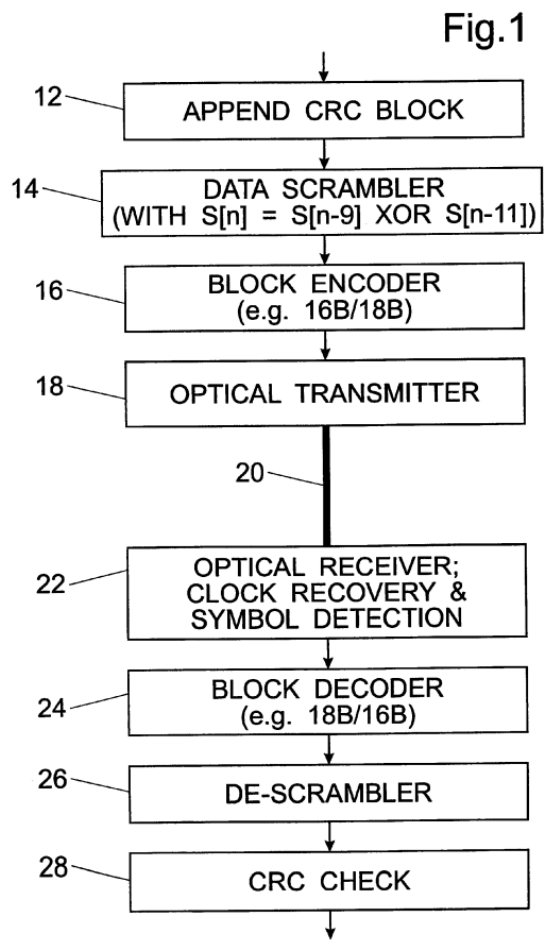
		part which is not necessarily substantially DC-free and which carries timing and/or other useful information (e.g. , channel identification, destination address, etc.) that can vary from data packet to data packet (e.g. , depending on sender, recipient, time of day, etc.).” Haartsen, 5:2–15. Austin discloses and/or renders obvious this limitation. *See, e.g.*: “This system uses up to 5 pre-defined data packet sizes and up to 16 user defined data message types within each packet. Each packet has Forward Error Connection (FEC) and bit interleaving applied to it on transmission. Error checking is performed using a standard CRC. The system uses standard run in and byte synchronisation techniques. The byte synchronisation word used is 32 _bits long and there are 5_different words used, one to identify each of the different packet types.” Austin, 4:37-45. To the extent that Plaintiff contends that the Additional Prior Art References do not explicitly or inherently disclose this element, or render obvious this element, this element is obvious in combination with one or more references and/or with the knowledge of a person of ordinary skill in the art, as set forth in Defendant’s Invalidity Contentions.
Claim 25		
25[p]	The communications device of claim 19	Chen discloses, expressly or inherently, the transmitting device of claim 19 under any proposed construction. *See* exemplary disclosures and corresponding discussion for claim 19.
25	wherein the packet types include a control packet.	Chen discloses and/or renders obvious this limitation. *See, e.g.*: “Various timing and identification information is provided in different portions of the transport stream. These include a packet identifier (PID) found in the transport header of each transport packet to provide a reference number for identifying the transport packets carrying a specific service component. This number is included in a service definition or "service map" used by the receiver to identify those transport packets required to reconstruct a television program signal. The PID may

		also be referenced for various grooming and remultiplexing functions. In the case of video, audio or isochronous control data, the stream of packets labeled with a single PID represents a single video, audio or isochronous data service elementary stream, respectively. Each type of packet will have a different PID identifying the packet type." Chen, 14:56–15:3. "The PES payload contains the information data that is desired to be transmitted to a receiver. Thus, the payload includes all of the video, audio and control information necessary for the receiver to decode and reconstruct, e.g., a digital television signal). In order to meet the requirements of robustness and simplicity, a fixed packet length approach is preferred to the variable length PES packets. Thus, as illustrated in FIG. 6b, the packet elementary stream containing the PES packets 670 is reconfigured into a stream of fixed length transport packets 680. The transport packets illustrated in FIG. 6b all correspond to the same service component, such as the video component of a digital television transmission. In the MPEG-2 and DigiCipher® II embodiments, each packet is 188 total bytes in length, with the first four bytes comprising a transport packet header (TP HDR) 682. The payload portion 684 of each packet 680 is thus normally 184 bytes. However, an adaptation field mechanism is present, as illustrated by transport packet 680', to extend the header when required. The adaptation field 686 provides additional information which is not required for every transport packet. The adaptation field (ADPT FIELD) 686 extends the regular transport header 682 at the expense of payload 684, which will be less than 184 bytes whenever the adaptation is provided. The adaptation field 686 is of variable length, depending on the information it contains. Typically, the adaptation field will support additional information for time base recovery and other functions, and also provides a mechanism for padding the payload when it does not occupy the full 184 bytes. Such padding can be used, for example, to make a variable rate video packetized elementary stream into a constant rate transport stream." Chen, 16:9–40.

		 Chen, Fig. 6b. Cunningham discloses and/or renders obvious this limitation. *See, e.g.*: "Control information is conveyed by using selected n-bit data block values, together with a distinctive value of 10 for the pair of label bits a and b to indicate that the associated data block represents control information rather than actual data. The data block values used for this purpose are selected to be values which inherently have a disparity of zero, such as (for 8-bit blocks) 00001111 or 10101010. It will be noted that the label bit pair value 10 likewise has zero disparity, so the complete encoded block also has zero disparity." Cunningham, 4:23–31. To the extent that Plaintiff contends that the Additional Prior Art References do not explicitly or inherently disclose this element, or render obvious this element, this element is obvious in combination with one or more references and/or with the knowledge of a person of ordinary skill in the art, as set forth in Defendant's Invalidity Contentions.
Claim 26		
26[p]	The communications device of claim 19	Chen discloses, expressly or inherently, the transmitting device of claim 19 under any proposed construction. *See* exemplary disclosures and corresponding discussion for claim 19.

26	wherein the symbols are transmitted to a switch network.	Chen discloses and/or renders obvious this limitation. *See, e.g.*: "FIG. 1 is a block diagram of a Fiber-to-the-Curb Digital Video Distribution Network in accordance with the present invention. The distribution network includes Video Information Providers (VIPs) shown generally at 100, a gateway, shown generally at 120, an access network, shown generally at 140, and the home of a Video Information User (VIU), shown generally at 160. The gateway 120 may comprise an Ll Gateway manufactured by Teleco. The VIPs may include a video server 102, an interactive data server 104, and an internet gateway 106, all of which communicate with an asynchronous transfer mode (ATM) network 122 via, for example, an OC-3 system, where optical fiber is used to transmit data at 155.52 Mbit/sec. The ATM network 122 may also receive data via a satellite receiving antenna 110, broadcast encoder and server 112, and ATM multiplexer 114. The ATM network 122 communicates with a session manager 123." Chen, 4:11–28.  FIG.1

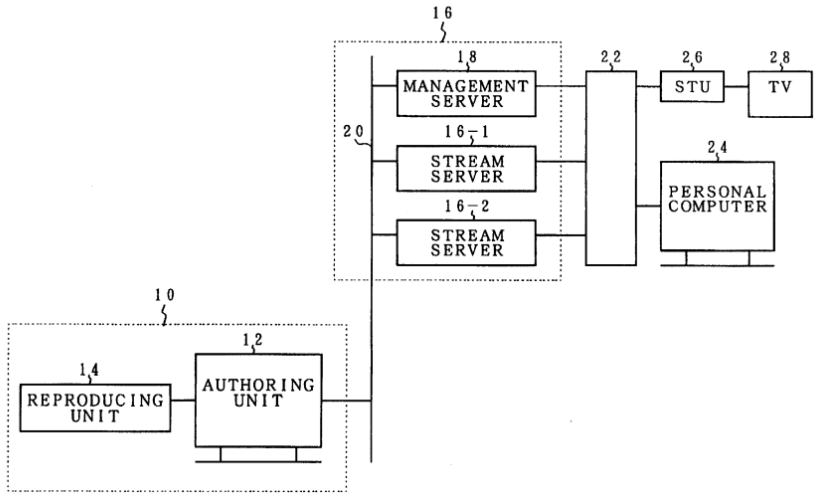
		Chen, Fig. 1. "A compressed digital packetized data stream which includes the inserted message combined with the main program is output from the DAIM 130 to an ATM transmitter (Tx) interface 126, and then provided to an access network (e.g., cable system headend) 140. In the access network 140, a Host Digital Terminal (HDT) 144 receives the data stream and also receives control signals from an element manager 142, which manages the cable network. The HDT 144, which is a specialized digital switch which is typically located in a head office of the cable system, communicates with one or more Optical Node Units (ONUs), including ONU 146. The ONU is located in the local curb, and supports an optical to electrical conversion and modulation/demodulation. The ONU 146 then provides the data stream over a coax distribution network to one or more subscriber's homes or offices. At each home, a drop box 164 located near the home receives the combined digital signal and provides it to a Digital Entertainment Terminal (DET) (e.g., decoder) 168 and a Network Interface Module (NIM) 166. The DET decodes the packetized data stream, and the signal is seamlessly delivered for display to the Video Information User (VIU) (e.g., subscriber) via televisions 170, 172 and/or personal computer 174." Chen, 4:63–5:17. Cunningham discloses and/or renders obvious this limitation. *See, e.g.*: "Referring to FIG. 1, a data frame intended to be communicated over an optical fibre channel 20 typically comprises a binary digital message (payload) to be transferred between stations on the network; this message may comprise user data, or control information for managing operation of the network and its components. A circuit 12 generates an associated cyclic redundancy check (CRC) block containing check data derived from this message in known manner in accordance with a predetermined CRC algorithm, and appends the CRC block to the message." Cunningham, 2:65–3:7.



Cunningham, Fig. 1.

Hiroshima discloses and/or renders obvious this limitation. *See, e.g.*:

"The memory medium is set in a drive of each of the stream servers 16-1 and 16-2 in the center 16 and is reproduced. A management server 18 is connected to the stream servers 16-1 and 16-2

		through the network 20. The stream servers 16-1 and 16-2 and management server 18 are connected to a television apparatus 28 as a user terminal through an external network 22 and a set top unit 26 and are also connected to a personal computer 24 of the user through the network 22. The video-on-demand system corresponds to MPEG2 having a transmission rate of 6.144 Mbps and transmits data by a prefix bit rate of the MPEG2-TS through the network 22. An ATM network is generally used as a network 22. The ATM network operates in an asynchronous transfer mode (ATM) in which a transmission rate is equal to 156 Mbps." Hiroshima, 5:31–45. FIG. 1  Hiroshima, Fig. 1. To the extent that Plaintiff contends that Chen does not explicitly or inherently disclose this element, or render obvious this element, this element is obvious in combination with one or more references and/or with the knowledge of a person of ordinary skill in the art, as set forth in Defendant's Invalidity Contentions.

		Claim 28
28[p]	The communications device of claim 19	Chen discloses, expressly or inherently, the transmitting device of claim 19 under any proposed construction. *See* exemplary disclosures and corresponding discussion for claim 19.
28	wherein the synchronization primitive is made up of a sequence of bit-string synchronization symbols.	Cunningham discloses and/or renders obvious this limitation. *See, e.g.*: "There are many potential ways to achieve code block synchronization (i.e. correct alignment in the receiver of the label and payload fields). For example, one technique would be to transmit a 'comma sequence' during start-up. As an example of this technique consider a code with n=8 using the simple control payload scheme. A training control state of 1111 1111 would be transmitted as: [10 1111 1111 0000 0000][10 1111 1111 0000 0000][10 1111 1111 0000 0000] The receiver searches for the bit pair 10. This is found in only two locations: the correct location (the label field), and at the end of the run of eight 1's. However, if the receiver (incorrectly) identifies the label field to be the 10 bit pair at the end of the run of 1's, the following 16 bits are not a valid control payload (first 8 bits are not the complement of the second 8 bits). The receiver can therefore detect the mistake. It is also noted that this sequence contains a regular 010 pattern, which aids in phase-locked loop (PLL) clock acquisition." Cunningham, 7:5–23. Shin discloses and/or renders obvious this limitation. *See, e.g.*: "During each preambling period, three repetitions of the same synchronization character are produced by the encoder 18. As is described herein, by processing the 21 bits most recently produced by the encoder 14, synchronization module 114 is able to detect at least one of the three repetitions of the synchronization character transmitted during a given preambling period. This advantageously allows synchronization to be achieved within a relatively short preambling period." Shin, 15:20–28.

		"In the exemplary embodiment a preambling sequence (i.e., three repetitions of the same out-of-band SYNC character) is sent upon system power-up as well as during lapses in data transmission over the serial link 30. This allows timing synchronization to be maintained between the encoder 18 and decoder 44 even in the absence of data transmission for extended periods." Shin, 18:19–25. Austin discloses and/or renders obvious this limitation. *See, e.g.*: "This system uses up to 5 pre-defined data packet sizes and up to 16 user defined data message types within each packet. Each packet has Forward Error Connection (FEC) and bit interleaving applied to it on transmission. Error checking is performed using a standard CRC. The system uses standard run in and byte synchronisation techniques. The byte synchronisation word used is 32 _bits long and there are 5_different words used, one to identify each of the different packet types." Austin, 4:37-45. To the extent that Plaintiff contends that Chen does not explicitly or inherently disclose this element, or render obvious this element, this element is obvious in combination with one or more references and/or with the knowledge of a person of ordinary skill in the art, as set forth in Defendant's Invalidity Contentions.