

We Claim:

1. A method of playing out one or more multimedia objects received over a packet-switched network by a client device comprising an electronic device or system, wherein source data of the one or more multimedia objects is represented by encoded symbols in packets such that the source data is recoverable, at least approximately, from the encoded symbols, the method comprising:

a) receiving encoded symbols via a broadcast channel, wherein a value of each encoded symbol received for a multimedia object comprises either a repair symbol derived from values of source symbols for that multimedia object or comprises a source symbol for that multimedia object;

b) determining an indication of which portions of the multimedia object are for playing out to narrow any additional data requested by the client device to data that is to be played out such that data that is not to be played back is not requested, wherein the indicated portions of the multimedia object for playing out are used to determine an amount of additional data needed beyond the received encoded symbols to request for recovering the indicated portions;

c) determining, using the indication, the amount of additional data needed beyond the received encoded symbols to recover the indicated portions of the multimedia object for playing out;

d) determining a corresponding set of one or more byte ranges of one or more files, wherein the corresponding set corresponds to the additional data within the one or more files needed to recover the indicated portions of the multimedia object for playing out;

e) generating one or more requests for at least a portion of the corresponding set, using one or more requests directed to a server, wherein each request specifies one or more byte ranges;

f) sending the one or more requests;

g) receiving at least some of the requested additional data in response to the sent one or more requests; and

h) using the received additional data in combination with the encoded symbols received via the broadcast channel in recovering indicated portions of the multimedia object for playing out with the client device.

2. The method as claimed in claim 1, wherein the additional data needed beyond the received encoded symbols comprises source symbols of the indicated portions of the multimedia object for playing out.

3. The method as claimed in claim 1, wherein the additional data needed beyond the received encoded symbols comprises at least some repair symbols of the indicated portions of the multimedia object for playing out.

4. The method as claimed in claim 1, wherein the encoded symbols received via the broadcast channel are all repair symbols.

5. The method as claimed in claim 1, wherein the additional data needed beyond the received encoded symbols comprises source sub-symbols of the indicated portions of the multimedia object for playing out.

6. The method as claimed in claim 1, wherein playing out with the client device comprises playing out based on data received from a unicast repair server, and when the client device determines that enough data is available from a unicast repair server and a broadcast to recover a portion of the multimedia object, playing out the portion while concurrently receiving from a unicast repair server other portions of the multimedia object.

7. The method as claimed in claim 1, wherein determining the indication of which portions of the multimedia object are for playing out is based on the actions taken by an application executing on the client device or the actions taken by a user of the client device to indicate which portions of the multimedia object are to be played out.

8. The method as claimed in claim 7, wherein the actions taken by the application or user are distributed in time and refer to distinct portions of the multimedia object for playing out at different times.

9. The method as claimed in claim 7, wherein generating the one or more requests, sending the one or more requests, receiving at least some of the requested

additional data, and using the received additional data occurs in response to the actions taken by the application or user.

10. The method as claimed in claim 7, wherein the application determines that the entire multimedia object is for playing out, and wherein playing out comprises writing the recovered multimedia object to storage on the client device.

11. The method as claimed in claim 7, wherein the multimedia object is an audio/video object and playing out comprises providing access to portions of the multimedia object to a multimedia player on the client device for display on a display screen.

12. The method as claimed in claim 1, wherein at least one of the one or more files is in an Extended-original-order HTTP file format.

13. The method as claimed in claim 12, further comprising receiving and processing signaling that indicates that a file is in the Extended-original-order HTTP file format, with the signaling comprising an indication of a universal object symbol identifier range.

14. The method as claimed in claim 1, wherein at least one of the one or more files is an original-order HTTP file of the multimedia object, and data in the at least one file is in a native format.

15. The method as claimed in claim 1, wherein at least one of the one or more files is a universal file symbol identifier-order HTTP file of the multimedia object.

16. The method as claimed in claim 1, wherein the server is a HyperText Transport Protocol server that is responsive to file requests and byte range requests and that serves files and byte ranges of files.

17. The method as claimed in claim 16, wherein the HTTP requests are HTTP 1.1 formatted requests, and the HTTP server supports the HTTP/1.1 protocol.

18. The method as claimed in claim 1, wherein the multimedia object is partitioned into source blocks, and wherein a minimal set of source blocks comprising the indicated portions of the multimedia object for playing out is recovered.

19. The method as claimed in claim 18, wherein at least one of the one or more files is an original-order HTTP file of the multimedia object, and wherein at most one byte range request is generated and sent for each source block in the minimal set of source blocks and the byte range request for a source block corresponds to a consecutive prefix of source symbols from the source block.

20. The method as claimed in claim 18, wherein source blocks of the multimedia object are further partitioned into source sub-blocks, and wherein a minimal set of source sub-blocks comprising the indicated portions of the multimedia object for playing out is recovered.

21. The method as claimed in claim 20, wherein at least one of the one or more files is an original-order HTTP file of the multimedia object, and wherein at most one byte range request is generated and sent for each source sub-block in the minimal set of source sub-blocks and the byte range request for a source sub-block corresponds to a consecutive prefix of source sub-symbols from the source sub-block.

22. The method as claimed in claim 1, wherein at least one of the one or more requests includes one or more repair symbols.

23. The method as claimed in claim 1, wherein the amount of additional data needed beyond the received encoded symbols to request for recovering the indicated portions is determined based on an FEC Object Transmission Information provided for each multimedia object to the client device.

24. A client device, capable of playing out one or more multimedia objects received over a packet-switched network, wherein source data of the one or more multimedia objects is represented by encoded symbols in packets such that the source data is recoverable, at least approximately, from the encoded symbols, the client device comprising:

a) a broadcast receiver input for receiving encoded symbols via a broadcast channel, wherein a value of each encoded symbol received for a multimedia object comprises either a repair symbol derived from values of source symbols for that multimedia object or comprises a source symbol for that multimedia object;

b) logic for determining an indication of which portions of the multimedia object are for playing out to narrow any additional data requested by the client device to data that is to be played out such that data that is not to be played back is not requested, wherein the indicated portions of the multimedia object for playing out are used to determine an amount of additional data needed beyond the received encoded symbols to request for recovering the indicated portions;

c) logic for determining, using the indication, the amount of additional data needed beyond the received encoded symbols to recover the indicated portions of the multimedia object for playing out;

d) logic for determining a corresponding set of one or more byte ranges of one or more files, wherein the corresponding set corresponds to the additional data within the one or more files needed to recover the indicated portions of the multimedia object for playing out;

e) logic for generating one or more requests for at least a portion of the corresponding set, using one or more requests directed to a server, wherein each request specifies one or more byte ranges;

f) a network interface for sending the one or more requests on the packet-switched network;

g) logic for receiving at least some of the requested additional data in response to the sent one or more requests; and

h) a processor for processing the received additional data in combination with the encoded symbols received via the broadcast channel in recovering indicated portions of the multimedia object for playing out with the client device.

25. The client device as claimed in claim 24, wherein the additional data needed beyond the received encoded symbols comprises source symbols of the indicated portions of the multimedia object for playing out.

26. The client device as claimed in claim 24, wherein the additional data needed beyond the received encoded symbols comprises at least some repair symbols of the indicated portions of the multimedia object for playing out.

27. The client device as claimed in claim 24, wherein the encoded symbols received via the broadcast channel are all repair symbols.

28. The client device as claimed in claim 24, wherein the additional data needed beyond the received encoded symbols comprises source sub-symbols of the indicated portions of the multimedia object for playing out.

29. The client device as claimed in claim 24, further comprising an application executable by the client device or executable by a user of the client device to indicate which portions of the multimedia object are for playing out.

30. The client device as claimed in claim 29, further comprising storage for storing a recovered multimedia object.

31. The client device as claimed in claim 24, wherein the multimedia object is an audio/video object and the client device further comprises a display screen for displaying the multimedia object.

32. The client device as claimed in claim 24, wherein at least one of the one or more files is in an Extended-original-order HTTP file format.

33. The client device as claimed in claim 32, wherein received signals indicate that a file is in the Extended-original-order HTTP file format, the signaling comprising an indication of a universal object symbol identifier range.

34. The client device as claimed in claim 24, wherein at least one of the one or more files is an original-order HTTP file of the multimedia object, and data in the at least one file is in a native format.

35. The client device as claimed in claim 24, wherein at least one of the one or more files is a universal file symbol identifier-order HTTP file of the multimedia object.

36. The client device as claimed in claim 24, wherein the server is a HyperText Transport Protocol server that is responsive to file requests and byte range requests and that serves files and byte ranges of files.

37. The client device as claimed in claim 36, wherein the HTTP requests are HTTP 1.1 formatted requests, and the HTTP server supports the HTTP/1.1 protocol.

38. The client device as claimed in claim 24, wherein at least one of the one or more requests includes one or more repair symbols.

39. The client device as claimed in claim 24, wherein the amount of additional data needed beyond the received encoded symbols to request for recovering the indicated portions is determined based on an FEC Object Transmission Information provided for each multimedia object to the client device.

40. A method of receiving, over at least a packet-switched network, by a client device comprising an electronic device or system, one or more multimedia objects, wherein source data of the one or more multimedia objects is represented by encoded symbols in packets such that the source data is recoverable, at least approximately, from the encoded symbols, the method comprising:

- a) receiving a signal indicative of a type of transmission for symbols, wherein at least one type of transmission is such that symbols are to be received over a broadcast channel and additional data can be received over a unicast channel, wherein each of the symbols received over the broadcast channel is a repair symbol;

- b) receiving repair symbols via the broadcast channel, wherein a value of each repair symbol is a value derived from values of source symbols for the multimedia object;

- c) determining an indication of which portions of the multimedia object are for playing out to narrow any additional data requested by the client device to data that is to be played out such that data that is not to be played back is not requested, wherein the indicated portions of the multimedia object for playing out are used to determine an

amount of additional data needed beyond the received encoded symbols to request for recovering the indicated portions;

d) determining, using the indication, the amount of additional data beyond the received repair symbols to recover the indicated portions of the multimedia object for playing out;

e) determining a corresponding set of one or more byte ranges of one or more files, wherein the corresponding set corresponds to the additional data;

f) generating one or more requests for at least a portion of the corresponding set, using one or more requests directed to a server, wherein each request specifies one or more byte ranges;

g) sending the one or more requests;

h) receiving at least some of the requested additional data in response to the sent one or more requests; and

i) using the received additional data in combination with the repair symbols received over the broadcast channel in recovering the indicated portions of the multimedia object for playing out with the client device.

41. The method as claimed in claim 40, further comprising receiving a plurality of broadcast streams comprising the repair symbols, wherein the plurality of broadcast streams are independent.

42. A method of sending metadata to a client device wherein the metadata provides information about a multimedia object to be received by the client device as broadcasted encoded symbols combined with, as needed, unicast encoded symbols, wherein the encoded symbols are such that the multimedia object is recoverable, at least approximately, from received encoded symbols, the method comprising:

a) sending a first signal indicative of broadcast information for a transmission over a broadcast channel, wherein each symbol sent over the broadcast channel is a repair symbol, wherein a value of each repair symbol is a value derived from values of source symbols for the multimedia object; and


b) sending a second signal indicative of information for making one or more requests for additional symbols needed beyond received repair symbols to recover portions of the multimedia object indicated for playing out such that data that is not to be played back is not requested, wherein a request for additional symbols comprises an

indication of a file and a byte range request, the file corresponding to portions of the multimedia object received, at least in part, over the broadcast channel.

43. The method as claimed in claim 42, wherein the request for additional symbols comprises an HTTP/1.1 protocol request including a URL for the file and the byte range within that file.

44. The method as claimed in claim 43, further comprising sending a plurality of broadcast streams from a plurality of transmitters, wherein the plurality of broadcast streams are non-overlapping and comprise repair symbols.

Dated 21/04/2014



[P. SEETHA]
OF REMFRY & SAGAR
ATTORNEY FOR THE APPLICANT[S]

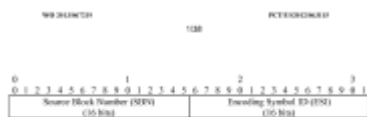


Fig. 1A. Raptor-RFC 5053 Payload ID (prior art)



Fig. 1B. RaptorQ-Spec Payload ID (prior art)



Fig. 2

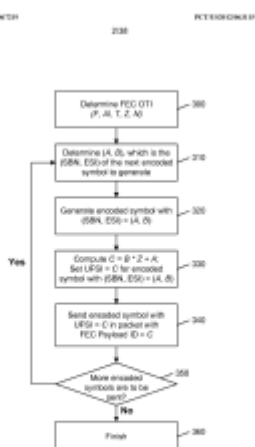


Fig. 3

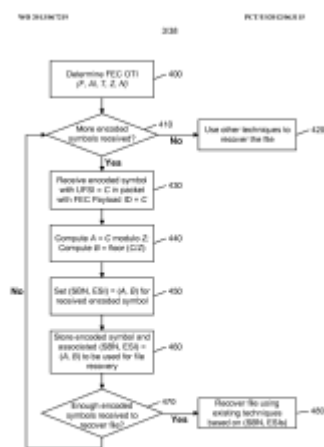


Fig. 4

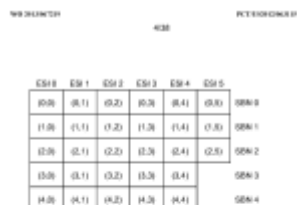


Fig. 5A



Fig. 5B

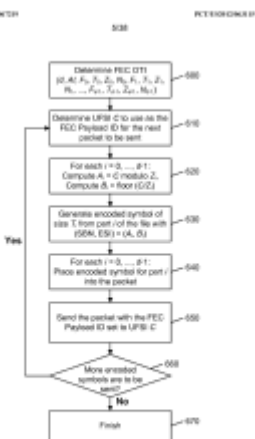


Fig. 6

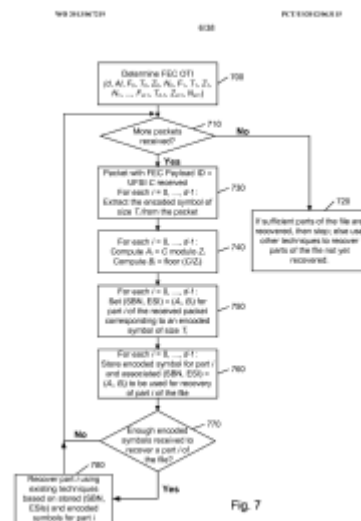


Fig. 7

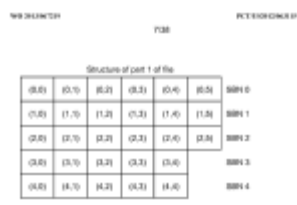


Fig. 8A

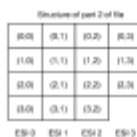


Fig. 8B



Fig. 9

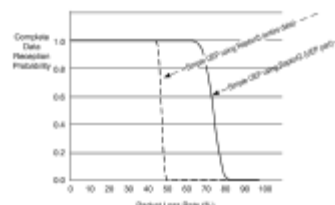


Fig. 10

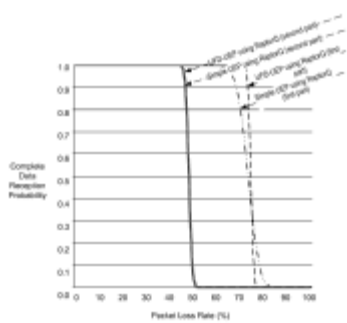


Fig. 11

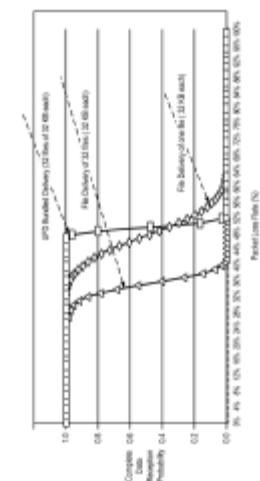


FIG. 12

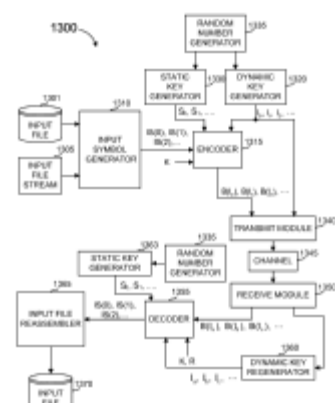


Fig. 13

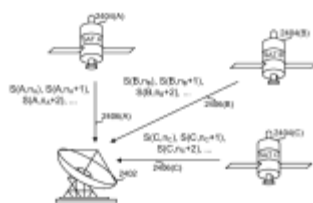


Fig. 14

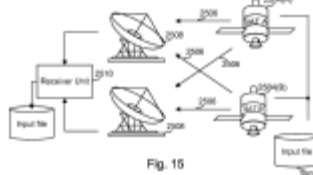


Fig. 15

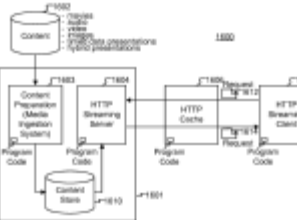


Fig. 16

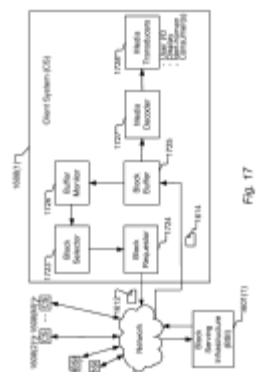


Fig. 17

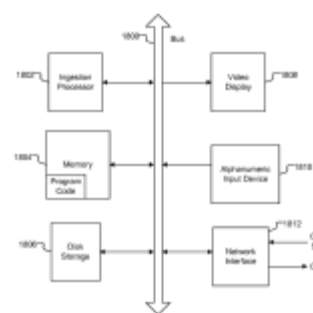


Fig. 18

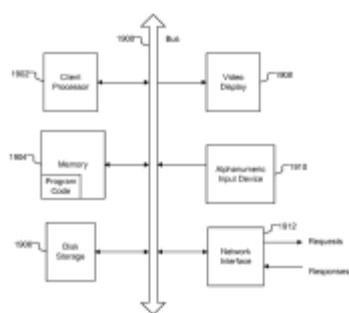


Fig. 19

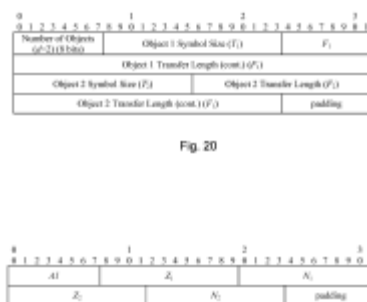


Fig. 20

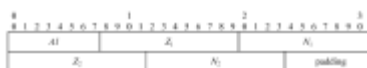


Fig. 21

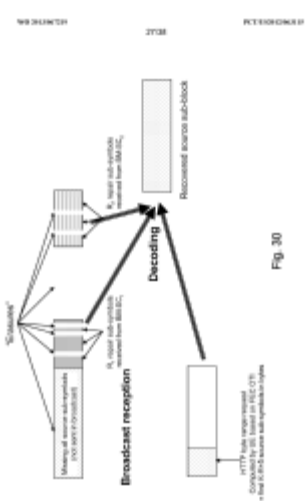
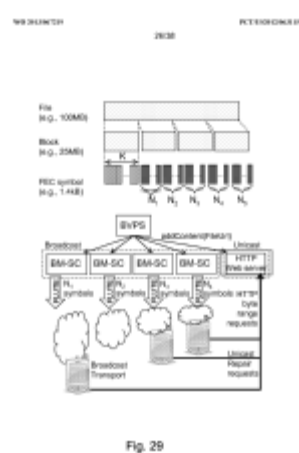
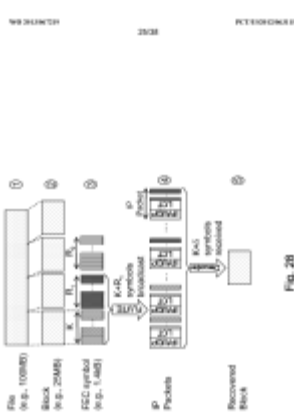
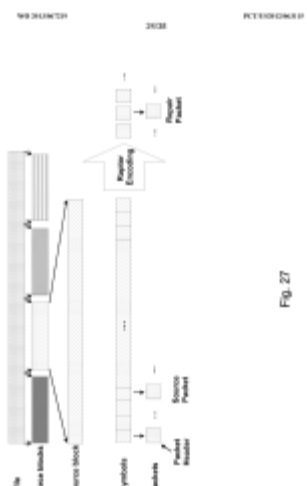
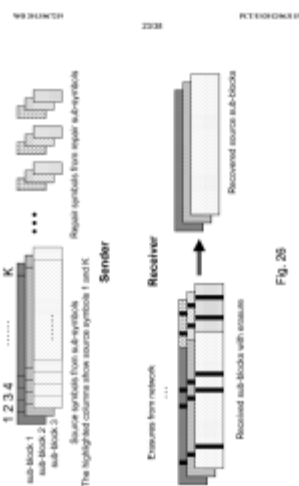
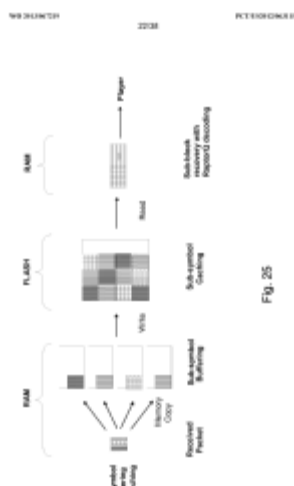
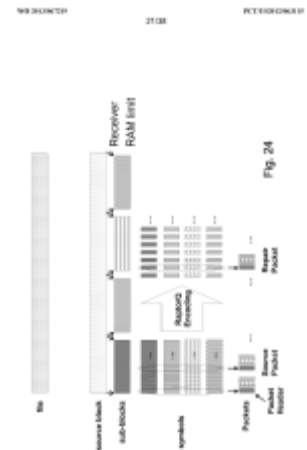
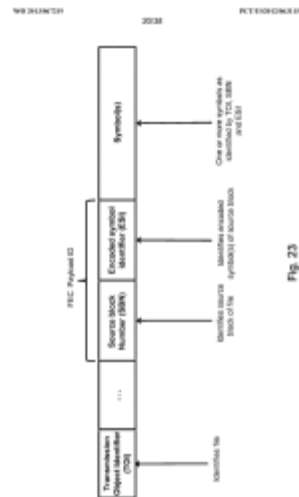




Fig. 37

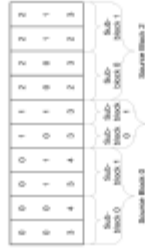


Fig. 38

Sub-Block Number (N)	Start Bit (S)	End Bit (E)	Start Bit (S)	End Bit (E)	Start Bit (S)	End Bit (E)
1	1	16	17	32	33	48
2	1	16	17	32	33	48
3	1	16	17	32	33	48
4	1	16	17	32	33	48
5	1	16	17	32	33	48
6	1	16	17	32	33	48
7	1	16	17	32	33	48
8	1	16	17	32	33	48
9	1	16	17	32	33	48
10	1	16	17	32	33	48
11	1	16	17	32	33	48
12	1	16	17	32	33	48
13	1	16	17	32	33	48
14	1	16	17	32	33	48
15	1	16	17	32	33	48
16	1	16	17	32	33	48

Fig. 39

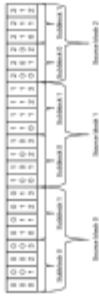


Fig. 34



Fig. 35



Fig. 36

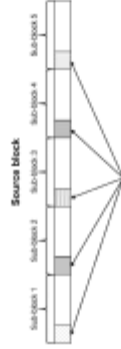


Fig. 31

Sub-Block Number (N)	Start Bit (S)	End Bit (E)	Start Bit (S)	End Bit (E)	Start Bit (S)	End Bit (E)
1	1	16	17	32	33	48
2	1	16	17	32	33	48
3	1	16	17	32	33	48
4	1	16	17	32	33	48
5	1	16	17	32	33	48
6	1	16	17	32	33	48
7	1	16	17	32	33	48
8	1	16	17	32	33	48
9	1	16	17	32	33	48
10	1	16	17	32	33	48
11	1	16	17	32	33	48
12	1	16	17	32	33	48
13	1	16	17	32	33	48
14	1	16	17	32	33	48
15	1	16	17	32	33	48
16	1	16	17	32	33	48

Fig. 32

Sub-Block Number (N)	Start Bit (S)	End Bit (E)	Start Bit (S)	End Bit (E)	Start Bit (S)	End Bit (E)
1	1	16	17	32	33	48
2	1	16	17	32	33	48
3	1	16	17	32	33	48
4	1	16	17	32	33	48
5	1	16	17	32	33	48
6	1	16	17	32	33	48
7	1	16	17	32	33	48
8	1	16	17	32	33	48
9	1	16	17	32	33	48
10	1	16	17	32	33	48
11	1	16	17	32	33	48
12	1	16	17	32	33	48
13	1	16	17	32	33	48
14	1	16	17	32	33	48
15	1	16	17	32	33	48
16	1	16	17	32	33	48

Fig. 33

Sub-Block Number	End Date	Start Date (Frequency)	Start Date (Frequency)	End Date	End Date (Frequency)
1	10/1	10/1	10/1	10/1	10/1
2	10/2	10/2	10/2	10/2	10/2
3	10/3	10/3	10/3	10/3	10/3
4	10/4	10/4	10/4	10/4	10/4
5	10/5	10/5	10/5	10/5	10/5
6	10/6	10/6	10/6	10/6	10/6
7	10/7	10/7	10/7	10/7	10/7
8	10/8	10/8	10/8	10/8	10/8
9	10/9	10/9	10/9	10/9	10/9
10	10/10	10/10	10/10	10/10	10/10
11	10/11	10/11	10/11	10/11	10/11
12	10/12	10/12	10/12	10/12	10/12
13	10/13	10/13	10/13	10/13	10/13
14	10/14	10/14	10/14	10/14	10/14
15	10/15	10/15	10/15	10/15	10/15
16	10/16	10/16	10/16	10/16	10/16
17	10/17	10/17	10/17	10/17	10/17
18	10/18	10/18	10/18	10/18	10/18
19	10/19	10/19	10/19	10/19	10/19
20	10/20	10/20	10/20	10/20	10/20
21	10/21	10/21	10/21	10/21	10/21
22	10/22	10/22	10/22	10/22	10/22
23	10/23	10/23	10/23	10/23	10/23
24	10/24	10/24	10/24	10/24	10/24
25	10/25	10/25	10/25	10/25	10/25
26	10/26	10/26	10/26	10/26	10/26
27	10/27	10/27	10/27	10/27	10/27
28	10/28	10/28	10/28	10/28	10/28
29	10/29	10/29	10/29	10/29	10/29
30	10/30	10/30	10/30	10/30	10/30
31	10/31	10/31	10/31	10/31	10/31

Fig. 40

Sub-Block Number	End Date	Start Date (Frequency)	Start Date (Frequency)	End Date	End Date (Frequency)
1	10/1	10/1	10/1	10/1	10/1
2	10/2	10/2	10/2	10/2	10/2
3	10/3	10/3	10/3	10/3	10/3
4	10/4	10/4	10/4	10/4	10/4
5	10/5	10/5	10/5	10/5	10/5
6	10/6	10/6	10/6	10/6	10/6
7	10/7	10/7	10/7	10/7	10/7
8	10/8	10/8	10/8	10/8	10/8
9	10/9	10/9	10/9	10/9	10/9
10	10/10	10/10	10/10	10/10	10/10
11	10/11	10/11	10/11	10/11	10/11
12	10/12	10/12	10/12	10/12	10/12
13	10/13	10/13	10/13	10/13	10/13
14	10/14	10/14	10/14	10/14	10/14
15	10/15	10/15	10/15	10/15	10/15
16	10/16	10/16	10/16	10/16	10/16
17	10/17	10/17	10/17	10/17	10/17
18	10/18	10/18	10/18	10/18	10/18
19	10/19	10/19	10/19	10/19	10/19
20	10/20	10/20	10/20	10/20	10/20
21	10/21	10/21	10/21	10/21	10/21
22	10/22	10/22	10/22	10/22	10/22
23	10/23	10/23	10/23	10/23	10/23
24	10/24	10/24	10/24	10/24	10/24
25	10/25	10/25	10/25	10/25	10/25
26	10/26	10/26	10/26	10/26	10/26
27	10/27	10/27	10/27	10/27	10/27
28	10/28	10/28	10/28	10/28	10/28
29	10/29	10/29	10/29	10/29	10/29
30	10/30	10/30	10/30	10/30	10/30
31	10/31	10/31	10/31	10/31	10/31

Fig. 41