

29 DEC 1999

An encoding/decoding device for generating a Run Length Limited and DC-free data sequence

5 **Description**

BACKGROUND OF THE INVENTION

1. Field of the Invention

10 This invention relates to a device for encoding an arbitrary bit sequence into a constraint channel sequence. The channel sequence has the constraint that it is run length limited, satisfies the parameters (d,k) where $d=1$, $k=7$ and the code rate is $2/3$. The invention also relates to a decoding device for recovering data from the recorded channel sequence.

15

2. Description of the Related Art

Run Length Limited codes, generically referred as (d, k) or RLL codes, have been widely and successfully applied in modern magnetic and optical recording systems. A (d, k) code sequence satisfies the constraint, which is imposed from the recording channel, that at least d "zeros" are recorded between successive data "ones", and no more than k "zeros" are recorded between successive data "ones". In optical recordings, the recorded signals have to comply with an additional constraint, namely that the signals should have limited power density at the low-frequency end. This condition is imposed in order not to disturb operation of servo systems and to eliminate some effects due to defects such as scratches, dust or fingerprints on the disc. A sequences with limited power at the low frequency end is usually termed a DC-free sequence. Well-known examples include EFM and EFMPlus, which are widely used in the Compact Disc and its high density successor the DVD, respectively.

The rate $2/3$ (1,7) code is a very popular (d,k) code used in magnetic recording. Several variations of such code schemes have already been proposed. One

example is known from the text book entitled 'Coding techniques for digital recorders' by K. A. Schouhamer Immink,, pp. 129 to 130, Prentice Hall (1991). However this encoding scheme suffers from the presence of a DC component which may become excessively large and may therefore introduce distortion in systems such as an optical recording channel that cannot handle the DC component.

There is a straightforward method for generating a DC-free sequence by controlling the value of a running digital sum (RDS) of it. The RDS is defined as the sum of a data bit sequence, when the sequence is expressed in non-return-to-zero(NRZ) format taking the bit value of $\{+1, -1\}$. If the RDS is bounded, the data sequence is DC-free. An explicit proof of this key conclusion can be found in the above mentioned text book by K. A. Schouhamer Immink,, pp. 162 to 163, Prentice Hall (1991). Therefore two approaches have emerged with which a given data sequence can be easily transformed into DC-free sequence. Either the source data or the encoded (d,k) sequence can be multiplexed with DC control bits.

Another encoding method is proposed by J. A. H. Kahlman and K. A. S. Immink in U.S. Patent No. 5,477,222. In the latter reference, a conversion means is described wherein the channel sequence satisfies parity preserving with the parameters: rate $2/3$, as well as the (d,k) constraints, where $d=1$, $k=8$. With this coding means, together with DC control in the source data level, the channel sequence can be generated without a DC component. However, the k value is still larger than the conventional value, which is equal to 7. The present invention proposes a encoding/decoding rule based on the abovementioned work. According to the present invention, a DC-free sequence can be generated with the same parameters as a conventional scheme, that is $d=1$, $k=7$. Such a sequence has better performance, because a smaller k can guarantee better characteristics of clock recovery. Also the relative experience of (1, 7) code from previous data recording applications can be used.

SUMMARY OF THE INVENTION

The present invention may provide a device for encoding n-bit source words into corresponding m-bit channel words, such that with simple DC control in the source bit stream, it may generate a DC-free data sequence in the channel. The encoded sequence may also satisfy (d,k) constraints where d=1, k=7. In the present invention, m and n are preferably chosen with the value of n=2, m=3.

According to one aspect of the present invention there is provided a device for encoding a sequence of data words of a binary source signal into a run length limited sequence that satisfies the constraints of (d,k) wherein d=1, k=7 and the code rate is 2/3, wherein the bit stream of the source signal is divided into n-bit source words, said device including means for converting said source words into corresponding m-bit channel words, such that the conversion is parity preserving and including means for inserting a DC control bit into the sequence of data words, such that the encoded sequence is DC free.

In U.S. patent No. 5477 222, a parity preserving code means is provided, in order to generate DC-free code sequence. In the US patent, one basic coding table and two substitute coding tables are provided, and encoded data complies with the (d,k) constraints where d=1, k=8. The parity, P, of an n-bit word $(x_1, \dots, x_n)$, $x_i \in \{0,1\}$, is given by

$$P = \sum_{i=1}^n x_i \bmod 2.$$

A code with a parity preserving property has the characteristic that the parity of the source word to be converted equals the parity of the corresponding code word in which they are converted. As a result, the encoder does not influence the polarity of the signal. When it is used in conjunction with DC-control bits, setting an even (or odd) number of "ones" in a source data segment will result in an even (or odd) number of "ones" in an encoded data segment. Thus the RDS of channel code sequence can be controlled at source data level.

Table I, II and III show coding means with a mapping rule similar to the rules in the US patent, and are used as reference code tables in the present invention.

TABLE I

5

source word (x_1, x_2) channel word (y_1, y_2, y_3)

10

SW_1	00	CW_1	000
SW_2	01	CW_2	001
SW_3	10	CW_3	100
SW_4	11	CW_4	101

TABLE II

15

source words unmodified coding modified coding

20

01 10	001 100	001 010
01 11	001 101	000 010
11 10	101 100	101 010
11 11	101 101	100 010

TABLE III

25

source words unmodified coding modified coding

30

00 00 00	000 000 000	000 010 010
00 00 01	000 000 001	001 010 010
10 00 00	100 000 000	100 010 010
10 00 01	100 000 001	101 010 010

With these three conversion tables, the encoded channel words forms a sequence with (d, k) parameters in which $d=1$, $k=8$. With the further analysis, there is only one source words combination that has the possibility of generating a sequence with run length greater than 7. This special word pattern has a fixed length of four 2-bit source words. In order to control the k constraint that is equal to the conventional value which is 7, a further conversion means has to be provided. With this means, a detector is provided. When the special source data pattern occurs, in the meantime, another condition is also satisfied by the converting requirement, namely that the encoder will use the new conversion method to output the channel words. That is, in this invention, the fourth converting table may be used to convert four consecutive 2-bit source words into four 3-bit channel words. In addition, such a coding rule satisfies the property that it is parity preserving. Similarly on detection, the decoder is capable of decoding a block of four 3-bit channel words into the corresponding block of four 2-bit source words using the inverse conversion of the means mentioned above.

To realize the encoding of blocks of four 2-bit source words, a device in accordance with a preferred aspect of the invention may be characterized in that the converting means is adapted to convert blocks of four consecutive 2-bit source words into blocks of four consecutive 3-bit channel words in accordance with the coding given in the following table:

Table IV

block of 4 source words	block of 4 channel words
00 01 11 00	000 010 010 010
00 01 11 01	001 010 010 010
00 01 11 10	100 010 010 010
00 01 11 11	101 010 010 010

According to a further aspect of the present invention there is provided a device for decoding a channel signal sequence into a source signal sequence, wherein the bit sequence of the channel signal is generated by means of a device as
5 described above, the decoding device being adapted for mapping four, three, two or one 3-bit channel words into four, three, two and one 2-bit source words respectively, and wherein the conversion is based on Table I, II, III or IV respectively as herein disclosed.

10 According to a further aspect of the present invention, there is provided a device for decoding a sequence of data bits of a binary channel signal into a sequence of data bits of a binary source signal, wherein the bit sequence of the channel signal is divided into m-bit channel words, said device including deconverting means for deconverting the m-bit channel words into corresponding n-bit source
15 words, where n and m are integers and $m > n$.

Preferred embodiments of the present invention will now be described with reference to the accompanying drawings wherein:

20 FIG. 1 shows an embodiment of the present invention;

FIG. 2 shows a method for detecting the last bit of a channel word;

FIG. 3 shows power spectral density of a channel sequence with DC control
25 based on tables I, II, III and IV; and

FIG. 4 shows an embodiment of a decoding device.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

30

Fig. 1 shows an embodiment of present invention. A sequence of data bits of a binary source signal S is received from input terminal 1 and is coupled to an input

of a shift register 2, which has eight cells from X_1 to X_8 . The shift register 2 receives eight consecutive bits from the serial bit sequence in the source signal S. The shift register 2 performs serial-to-parallel conversion so that the parallel output of the source word are coupled to inputs i_1 to i_8 of a logic circuit LC in coding module CM.

The logic circuit LC is designed to convert the source word into a channel word which is parity preserving. The conversion rules are based on coding tables I, II, III and IV respectively. The coding module CM includes pattern detectors D1, D2 and D3 to detect code patterns from the source data sequence and output control signals to make the logic circuit LC select the relative coding table. If the signal output 10 from D1 which is coupled to input 11 of LC is valid, LC will convert the source word based on table II. If the control signal 20 from D2 which is coupled to LC in 21, is valid, LC will convert source word based on table III. If the control signal 30 from D3 which is coupled to LC in 31, is valid, LC will convert the source word based on table IV.

There is also a register T to record the last bit from Channel C. Its output 41 is coupled to input 32 of D3, so that D3 may make decision to output the control signal 30.

Coding module CM also includes a shift register 5, which has 12 cells Y_1 to Y_{12} . Outputs o_1 to o_{12} of logic circuit LC are coupled to these 12 cells respectively. The Shift register 5 performs parallel-to-serial conversion. It converts the channel words supplied by LC into a serial sequence to form a binary channel signal C.

The coding module CM converts the source word to a channel word such that the conversion is parity preserving. Normally the conversion is based on the basic table I, that is the 2-bit source word is converted into a 3-bit channel word. When the bit patterns in the source data sequence are the combinations "11.11", "11.10", "01.11" and "01.10", the output may violate the constraint $d=1$ if unmodified conversion is used. Detector D1 then outputs control signal 10, so

that LC may use table II instead of table I. That is, coding module CM converts the two consecutive 2-bit source words into two consecutive 3-bit channel words. When the bit patterns in the source data sequence are the combinations "00.00.00", "00.00.01", "10.00.00" and "10.00.01", the output may violate the
5 constraint $k=7$ if unmodified conversion is used. Detector D2 then outputs control signal 20, so that LC may use table III instead of table I. That is, CM convert the three consecutive 2-bit source words into three consecutive 3-bit channel words.

Beside the above mentioned source words combinations, with a further analysis,
10 there is a special sequence of source words "00.01.11" to be considered. With the conversion of table I and table II, the generated channel sequence is "000.000.010". If the last bit of previous channel word is "0", then this sequence will violate the $k=7$ constraint. Because this pattern is the only combination that may result the in the violation of $k=7$ constraint, this k violation can be avoided by
15 adopting another conversion method. Therefore when this special bit combination occurs, a new coding table, being table IV of the present application, is designed to ensure that no k violation occurs whilst retaining the characteristic of parity preservation. On the other hand, direct use of the table may result in a risk of violating the $d=1$ constraint, because the mapping in table IV may generate a
20 sequence starting with "1". If the last bit of the previous channel word is "1" then a violation will occur. Therefore one register T is designed in the coding module CM in Fig.1 to serve as a buffer to keep a record of the last bit of the previous channel word. The conversion table IV may be used depending on the specific pattern of source word sequence and the feedback of previous channel bit value
25 as well.

When detector D3 detects a pattern of three 2-bit source words " $X_1 X_2 X_3 X_4 X_5 X_6$ " which equals "00.01.11", and in the meantime the last bit of the previous channel word from the register T is also "zero", the logic circuit LC converts the
30 block of 4 consecutive 2-bit source words " $X_1 X_2 X_3 X_4 X_5 X_6 X_7 X_8$ " in accordance with the modified coding rule as given in the table IV, so as to obtain a block of four 3-bit channel words.

The detection of last bit value is designed to ensure that the channel sequence satisfies the constraints of $d=1$ and $k=7$. There are two methods to detect the last bit value of the previous coding block. One is shown in Fig.1 and is that the register T in the coding module CM dynamically keeps a record of channel C, so
5 when the last bit of channel coding word outputs in the channel C, the value is recorded in T and next source word can be encoded.

A second method is shown in Fig.2, and is that the last bit of the channel word
10 can be pre-determined. If one of "11", "21" and "31" from the detector D1, D2 and D3 in Fig.1 is valid, then this means last bit of the channel word is "zero". Otherwise, the register T will record the value of input " i_2 " shown in Fig.1, which means that the last bit of source word in the input 40. The output of T, 41, is coupled to the input of D3, 32, in Fig.1. Based on the coding table I, both the last
15 bit of source word and channel word has the same value. This may provide a fast method to detect the last bit of previous channel word.

Since the coding rules are parity preserving, an even (or odd) number of "ones" at source data level will result in an even (or odd) number of "ones" at channel data
20 level. Thus the model described here is very suitable for DC control. The source data is chopped into segments p bit long. One bit is inserted between the segments serving as a control bit. Thus this bit can control the polarity of the segment. By choosing the value of the control bit, the RDS of the whole channel sequence can be controlled. The DC content of the channel data sequence can
25 approach zero. The performance of low frequency in the channel bit sequence has a strong relation to the bits length of p . The smaller the value of p , the smaller amount of low frequency component while the larger the redundancy. The optimum value of p may be decided by practical applications.

30 Fig. 3 shows the frequency performance of the code sequence through computer simulation using the code tables I, II, III and IV. The value of p selected is 50, and the power spectral density is calculated against the normalized frequency,

which is the bit frequency. A good low frequency performance is shown by the curve.

FIG. 4 shows a decoding device for decoding the serial data sequence obtained
5 by the encoding module of FIG. 1 so as to obtain a binary source signal. The decoding device has an input terminal 40 for receiving the channel signal, which input terminal 40 is coupled to an input 46 of a shift register 41, comprising twelve cells Y_1 to Y_{12} . The shift register 41 functions as a serial-to-parallel converter so that blocks of four 3-bit channel words are applied to inputs i_1 to i_9 of a logic circuit
10 42. Logic circuit 42 incorporates the four tables I, II, III and IV. Outputs o_1 to o_8 of logic circuit 42 are coupled to inputs of cells X_1 to X_8 of a shift register 44, which has an output 47 coupled to an output terminal 45. A detector circuit 43 is present having inputs coupled to outputs of cells Y_4 to Y_{12} respectively of shift register 41, and outputs o_1 , o_2 and o_3 coupled to control inputs c_1 , c_2 and c_3 respectively of the
15 logic circuit 42. The detector circuit 43 is capable of detecting a '010' bit pattern in the cells Y_4 , Y_5 and Y_6 of the shift register 41 and is capable of detecting a bit pattern '010010' in the cells Y_4 to Y_9 of the shift register 41. It is also capable of detecting a '010010010' bit pattern in the cells from Y_4 to Y_{12} .

20 Upon detection of the '010010010' bit pattern, detector circuit 43 generates a control signal at its output o_3 and upon detection of the '010010' bit pattern in the cells from Y_4 to Y_9 , whilst there is no '010' bit pattern in the cells Y_{10} , Y_{11} and Y_{12} , it generates a control signal at its output o_2 , and upon detection of a '010' bit pattern in the cells Y_4 , Y_5 and Y_6 , whilst there is no '010' bit pattern in the cells
25 Y_7 , Y_8 and Y_9 , it generates a control signal at its output o_1 .

In the absence of the control signals, logic circuit 42 converts the 3-bit channel word stored in the cells Y_1 , Y_2 and Y_3 into its corresponding 2-bit source word, as per the conversion table I, and supplies the 2-bit source word to the cells X_1 and
30 X_2 . In the presence of the control signal at the input c_1 , logic circuit 42 converts the block of two 3-bit channel words stored in the cells Y_1 to Y_6 into a block of two 2-bit source words, as per the conversion table II, and supplies the two 2-bit

source words to the cells X_1 to X_4 . In the presence of the control signal at the input c_2 , logic circuit 42 converts the block of three 3-bit channel words stored in the cells Y_1 to Y_9 into a block of three 2-bit source words, as per the conversion table III, and supplies the three 2-bit source words to the cells X_1 to X_6 . In the
5 presence of the control signal at the input c_3 , logic circuit 42 converts the block of four 3-bit channel words stored in the cells Y_1 to Y_{12} into a block of four 2-bit source words, as per the conversion table IV, and supplies the four 2-bit source words to the cells X_1 to X_8 . In this way, the serial data stream of the channel signal is converted into the serial data stream of the source signal.

10

Finally, it is to be understood that various alterations, modifications and/or additions may be introduced into the constructions and arrangements of parts previously described without departing from the spirit or ambit of the invention.

15 References

J. A. H. Kahlman and K. A. S. Immink, US Patent 5 477 222, Dec. 1995.

"Coding Techniques for Digital Recorders" by K. A. Schouhamer Immink, Chapter
20 5.6.7, pp. 127 to 131, Prentice Hall (1991).

Claims

1. A device for encoding a sequence of data words of a binary source signal into a run length limited sequence that satisfies the constraints of (d,k) wherein $d=1$, $k=7$ and the code rate is $2/3$, wherein the bit stream of the source signal is divided into n-bit source words, said device including means for converting said source words into corresponding m-bit channel words, such that the conversion is parity preserving and including means for inserting a DC control bit into the sequence of data words, such that the encoded sequence is DC free.
2. A device as claimed in claim 1, characterized in that, the converting means is adapted to convert blocks of four consecutive 2-bit source words into blocks of four consecutive 3-bit channel words in accordance with the following table:

block of 4 source words	block of 4 channel words
00 01 11 00	000 010 010 010
00 01 11 01	001 010 010 010
00 01 11 10	100 010 010 010
00 01 11 11	101 010 010 010

3. Device as claimed in claim 1, including means for dynamically detecting the last bit of an encoded block by means of channel feedback.
4. Device as claimed in claim 1 including means for detecting the last bit of an encoded block by means of a predetermined scheme based on a characteristic of the coding rules.
5. Device for decoding a channel signal sequence into a source signal sequence, wherein the bit sequence of the channel signal is generated by means of a device according to claim 1, the decoding device being adapted for mapping



four, three, two or one 3-bit channel words into four, three, two and one 2-bit source words respectively, and wherein the conversion is based on Table I, II, III or IV respectively as herein disclosed.

- 5 6. A device for decoding a sequence of data bits of a binary channel signal into a sequence of data bits of a binary source signal, wherein the bit sequence of the channel signal is divided into m-bit channel words, said device including means for deconverting the m-bit channel words into corresponding n-bit source words, wherein n and m are integers and $m > n$.

