Description

FIELD OF INVENTION

Present invention is concerned with data coding for systems using partial-response signaling with maximum-likelihood sequence detection (PRML). Such a signaling and detection scheme is useful e.g. for magnetic or optic storage of data, and for data transmission over cables. In particular, the invention is concerned with data coding in PRML systems to shorten the time needed for recognizing timing preambles that precede the data sequences to ensure proper timing and gain acquisition in the receiver, and to achieve a more reliable distinction between arbitrary data sequences and the timing preamble.

BACKGROUND

Rate 8/9 constrained codes for partial-response class-IV signaling with maximum-likelihood sequence detection (PRML) have been designed to achieve proper operation of gain and timing control loops and reduction of path-memory length of Viterbi detectors in PRML receivers. In the presence of a precoder between partial-response class IV (PR-IV) channel and rate 8/9 encoder, these two objectives are met by restricting the maximum run-length of 0's, denoted by G_0, in encoded binary sequences $\{b_n\}$ and limiting the maximum number of consecutive 0's, denoted by I_0, in the two subsequences of $\{b_n\}$ with even or odd time indices. An example of a system where G_0 and I_0 are both limited by appropriate coding (either to 4 and 4, or to 3 and 6, respectively) is disclosed in U.S. Patent 4,707,681 to Eggenberger.

It would be desirable to have coding schemes which can accomplish a third task, i.e. to speed up the start-up procedure of PRML receivers by shortening the minimum observation length required to distinguish reliably encoded user data from the timing preamble $\{1,1,1,...,1,1\}$, which in many systems is part of training sequences preceding encoded data sequences $\{b_n\}$.

OBJECTS OF THE INVENTION

It is an object of the invention to devise a coding scheme for PRML systems by which the time required for distinguishing timing preambles from coded data sequences is reduced.

It is another object to provide a coding scheme allowing to use shorter timing preambles.

A further object of the invention is a PRML coding scheme allowing the selection of code mappings which enable simple implementation of encoders and decoders by logic gate arrays.

Another object of the invention is to provide a coding scheme for systems using the PRML technique and using a timing preamble of consecutive ones which allows to maintain code constraints that are useful for the proper operation of gain and timing control loops and for limiting the path memory-length of Viterbi decoders in the receiver.

SUMMARY OF THE INVENTION

These objects are achieved by an encoding method and by encoding apparatus as defined in claims 1 and 7. Preferred particular embodiments of this method and of encoding and decoding apparatus are defined in the dependent claims.

The invention attains the objects by limiting the number G_1 of consecutive ones in the encoded data sequence to a relatively low value, while maintaining the limitation in the numbers of consecutive zeros in the full encoded sequence as well as in the even and odd numbered partial sequences. The smaller the maximum run-length of 1's, denoted G_1, in encoded sequences $\{b_n\}$, the faster data can be discriminated from the preamble. Since the timing preamble has to be long enough to be reliably detected by PRML receivers during start-up, smaller values of G_1 allow shorter training sequences leading to faster synchronization of PRML receivers. The presented 8/9 codes, referred to by the 3-tuple (G_0, I_0, G_1), permit reliable, fast start-up of PRML receivers and the length of the preamble detection window needs typically to be less than one half of the window required with earlier codes presented in U.S.Patent 4,707,681.

Several coding schemes are disclosed which allow to select different combinations of code constraints so that a combination can be chosen which best suits the respective application. These schemes have furthermore the property that decoding can be achieved without knowledge of previously encoded data, thereby avoiding decoding error propagation in case of transmission errors.

Further disclosed are several code mapping schemes which allow to implement the encoders (and decoders) for the presented coding schemes with a small number of logic gates.

These and other advantages will become more apparent from the following description of preferred embodiments of the invention with reference to the drawings.

LIST OF DRAWINGS

Fig.1 is a schematic representation of the structure of stored data sequences which comprise a timing preamble;
Fig.2A is a block diagram representation of a system in which present invention finds application;
Fig.2B is a more detailed block diagram of the encoder section of the system of Fig.2A;
Figs.3A, 3B and 3C show for one specific code a pattern, in the matrix of all 9-bit words, which indicates the preselection of all potential codewords; the partitioning of the matrix of all possible 8-bit input words into nine subsets, to facilitate assignment between coder input and output words; and the assignment between input and output subsets;
Fig.4 shows for another code the pattern, in the matrix of all 9-bit words, which indicates the preselection of all potential codewords;
Figs.5A, 5B and 5C show similar selections and assignments as in Figs.3A, 3B and 3C, for a further code;
Figs.6A and 6B are block diagrams of a finite-state encoder and of a decoder for the implementation of the invention which is represented by the selection and assignment matrices of Figs.5A, 5B and 5C.
Tables 1 through 6 show the encoder logic and the decoder logic for three different implementations of the invention.

DETAILED DESCRIPTION

DATA FORMAT WITH PREAMBLES:

Fig.1 shows an example of the structure of a data sequence when stored in a magnetic recording medium (magnetic disk). As can be seen, each data record is stored separately. The data section of each record is preceded by a timing preamble and a synchronizing pattern; these are necessary for initiating the receiver to properly read the recorded data.

It is important that the timing preamble is correctly distinguished from the data. If the data can comprise any arbitrary bit patterns, including such which resemble the used timing preamble (i.e. {1,1,1, ... }), relatively long timing preambles must be used and relatively much time is required for reliable detection and discrimination.

To shorten this time, the present invention suggests to add a constraint to the coded data sequences such that bit patterns corresponding to such a timing preamble will only occur in limited length. This allows a faster distinction between the data and the preamble, and as a consequence allows to use shorter timing preambles (which require less storage space).

BASIC MODEL:

A system in which present invention is used can be represented by the following three main blocks depicted in Fig. 2A: rate 8/9 encoder 11, precoder 13, and PR-IV channel 15. In the following, input-output relationship of these blocks are discussed in detail. B denotes the binary alphabet {0,1}.

Magnetic-recording channels are baseband channels with a transfer characteristic exhibiting a spectral null at frequency f=0, a spectral maximum at a frequency that depends on the particular recording-system parameters and an exponential decay at higher frequencies. Therefore, they are particularly well suited to be treated as a PR-IV system after the channel-output signal is filtered in an appropriate manner. The PR-IV channel 15 is preceded by a precoder 13 whose input sequence is the encoded sequence $\{b_n\}$, $b_n \in B$, and whose output sequence $\{c_n\}$, $c_n \in B$, is given by

$$c_n = b_n \oplus c_{n-2} = \begin{cases} c_{n-2} & \text{, if } b_n = 0 \\ \bar{c}_{n-2} & \text{, if } b_n = 1 \end{cases} \qquad (1)$$

where $\oplus$ denotes mod 2 addition and overbar stands for negation. The input symbols of the PR-IV channel $d_n \in \{+1, -1\}$ and the precoder-output symbols c_n are related to each other (as shown by block 17) by the correspondence $d_n=2c_n - 1$, where c_n have to be interpreted as the real numbers 0 or 1. The ternary output sequence

of the PR-IV channel $\{y_n\}$ can then be written as

$$y_n = d_n - d_{n-2} = 2\,(c_n - c_{n-2}) = \begin{cases} 0 & \text{, if } b_n = 0 \\ \pm 2 & \text{, if } b_n = 1 \end{cases} \qquad (2)$$

where (1) was used to obtain the relationship between b_n and y_n. Note that the all-zero sequence and the timing preamble (the all-one sequence) at the input of the precoder result in the all-zero sequence and {..., +2,+2,-2,-2,+2,+2,-2,-2,...}, respectively, at the output of the PR-IV channel.

The run-length of 0's in the output sequence $\{y_n\}$ of the PR-IV channel and in its two subsequences with even or odd time indices have to be limited in order to aid the automatic gain control and timing recovery, and reduce the length of path memory in the Viterbi detector. This can be achieved by a method and apparatus as disclosed in U.S.Patent 4,707,681 to Eggenberger. The maximum run-length of channel outputs y_n identical to the timing preamble at the output of the PR-IV channel {..., +2,+2,-2,-2,...} must also be reduced to permit reliable, fast start-up of PRML receivers. These goals can be achieved efficiently by present invention using a rate 8/9 encoder 11 followed by a precoder 13 that translates the desired run-length constraints originally defined at the output of the PR-IV channel into simple run-length constraints (G_0, I_0, G_1) at the output of the rate 8/9 encoder.

Precoding does not affect the performance of data communication systems employing binary PR-IV signaling and maximum-likelihood sequence-detection. However, the presence of a precoder can lead to smaller run-length constraints at the output of the PR-IV channel if encoders are restricted to a specific class. For example, in the absence of a precoder, the class of rate 8/9 block encoders with output block-size equal to 9 cannot produce channel-output sequences $\{y_n\}$ which have at most u=4 consecutive 0's and whose two subsequences with even or odd time indices have at most v=4 consecutive 0's. This can be explained as follows. There are no more than 216 binary 9-tuples which lead to PV-IV sequences $\{y_n\}$ satisfying the constraints u=4 and v=4 when they are transmitted over the PR-IV channel in any order. On the other hand, when the PR-IV channel is preceded by a precoder, there are 279 binary 9-tuples which can be juxtaposed in any order, and rate 8/9 block codes with output block-size equal to 9 that give rise to channel-output sequences constrained by u=4 and v=4 can be constructed (cf. U.S.Patent 4,707,681 mentioned above).

The input of the rate 8/9 encoder $\{a_n\}$ $a_n \in B$, is an unconstrained binary sequence and the encoder output $\{b_n\}$, $b_n \in B$, satisfies the following Boolean equations

$$\sum_{k=0}^{G_0} b_{n+k} \equiv b_n + b_{n+1} + b_{n+2} + \cdots + b_{n+G_0} = 1 \quad , \qquad (3)$$

$$\sum_{k=0}^{I_0} b_{n+2k} \equiv b_n + b_{n+2} + b_{n+4} + \cdots + b_{n+2I_0} = 1 \quad , \qquad (4)$$

$$b_n \cdot b_{n+1} \cdot b_{n+2} \cdot \ldots \cdot b_{n+G_1} = 0 \, , \qquad (5)$$

where n is any integer, '+' denotes the Boolean OR operation and '·' stands for the Boolean AND operation. Equation (3) limits the maximum run-length of 0's in the encoder-output sequence to G_0. Equation (4) does not allow the maximum number of consecutive 0's in the two encoder-output subsequences with even or odd time indices to exceed I_0. Finally, equation (5) ensures that the maximum run-length of 1's in the encoder-output sequence is not greater than G_1. In the following, the encoder output will be designated as a (G_0,I_0,G_1)-constrained sequence.

ENCODER INPUT AND OUTPUT:

Fig.2B shows more detail of the encoder section 11 of Fig.2A. The encoder proper 11B which accepts 8-bit input blocks X and furnishes 9-bit output codewords Y, is preceded by a serial-to-parallel converter 11A which converts the serial stream of bits a_n into the input blocks X. At the output side of the encoder, a parallel-to-serial converter 11C converts the codewords Y into a serial bit stream of bits b_n.

ENCODERS AND DECODERS FOR SELECTED BLOCK CODES:

The encoder for an (n,k) block code with rate equal to k/n assigns to every block of k input symbols a block of n output symbols. The motivation for the use of (9,8) block codes as opposed to non-block codes is the simplicity of the encoder/decoder hardware associated with block codes. The nonlinear binary (9,8) block codes designed so far were constructed with the objective of minimizing the constraints G_0 and I_0 (cf. e.g. U.S.Patent 4,707,681). In the following, simple encoder/decoder realizations of two optimum (9,8) block codes which minimize the parameter G_1 in addition to G_0 and I_0 are presented. In terms of the 3-tuple (G_0,I_0,G_1), these codes are characterized by (4,6,5) and (4,4,9). The smallest sum of constraints $G_0+I_0+G_1$ achievable with a (9,8) block code is 15. This minimum is only attained by (4,6,5) and (4,5,6) among all constraints (G_0,I_0,G_1).

ENCODER/DECODER FOR (4,6,5) CODE:

An encoder/decoder realization of the (4,6,5) code is described first. The encoder design starts by selecting all 9-tuples $Y = (Y_1,Y_2...,Y_9) \in B^9$ which satisfy the following Boolean equations

$$(Y_1 + Y_2 + Y_3)\cdot(Y_2 + Y_3 + Y_4 + Y_5 + Y_6)\cdot(Y_3 + Y_4 + Y_5 + Y_6 + Y_7)\cdot(Y_4 + Y_5 + Y_6 + Y_7 + Y_8)\cdot(Y_7 + Y_8 + Y_9) = 1 \quad (6)$$

$$(Y_1 + Y_3 + Y_6 + Y_7)\cdot(Y_3 + Y_5 + Y_7 + Y_9)\cdot(Y_2 + Y_4 + Y_6 + Y_8) = 1 \quad (7)$$

and either

$$Y_1\cdot Y_2\cdot Y_3 + Y_2\cdot Y_3\cdot Y_4\cdot Y_5\cdot Y_6\cdot Y_7 + Y_3\cdot Y_4\cdot Y_5\cdot Y_6\cdot Y_7\cdot Y_8 + Y_6\cdot Y_7\cdot Y_8\cdot Y_9 = 0 \quad (8a)$$

or

$$Y_1\cdot Y_2\cdot Y_3\cdot Y_4 + Y_2\cdot Y_3\cdot Y_4\cdot Y_5\cdot Y_6\cdot Y_7 + Y_3\cdot Y_4\cdot Y_5\cdot Y_6\cdot Y_7\cdot Y_8 + Y_7\cdot Y_8\cdot Y_9 = 0 \quad (8b)$$

where (6), (7) and (8a) are used to generate a list L_1 of 263 codewords and (6), (7) and (8b) another list L_2 of 263 codewords. By discarding all 9-tuples Y which begin or end with more than two 0's or have a run of five or more 0's, equation (6) ensures that the constraint $G_0 = 4$ is satisfied. Equation (7) guarantees $I_0 = 6$ by limiting the run-length of 0's within $Y_1Y_3Y_5Y_7Y_9$ and $Y_2Y_4Y_6Y_8$ to three. Finally, equation (8a) ((8b)) ensures $G_1 = 5$ by eliminating all Y which begin with more than two (three) 1's or end with more than three (two) 1's or have a run of six or more 1's. Note that if $Y = (Y_1,Y_2,...,Y_9) \in L_1$ then $Y' = (Y_9,Y_8,..., Y_1) \in L_2$. Hence the codewords in one list can be simply obtained by reversing the codewords in the other list. The (4,6,5) block code presented here comprises 256 codewords and was arbitrarily chosen to be a subset of L_1 as opposed to L_2. The 263 potential codewords in L_1 will be illustrated as 263 specific locations marked "■" within the 16 × 32 array shown in Fig.3A, whose (i,j)-th location at the intersection of the i-th row, i=0,1,...,15, and the j-th column, j=0,1,...,31, corresponds to the 9-tuple $Y=(Y_1,Y_2,...,Y_9)$ where $Y_9Y_8Y_7Y_6$ and $Y_5Y_1Y_2Y_3Y_4$ are the binary representations of the decimal numbers i and j, respectively. Note that the left half of the above array corresponds to $Y_5=0$ whereas the right half corresponds to $Y_5=1$. The locations in $L_1(4,6,5)$ marked "-" represent 9-tuples which are not allowed to be used as codewords in the (4,6,5) block code presented here.

There are numerous block codes that can be derived from the array $L_1(4,6,5)$, and the codewords in each code can be assigned to 256 8-bit bytes in 256! ways. Thus, the number of all encoder mappings associated with the array L_1 (4,6,5) is very large. Thus, it is desirable to choose an encoder mapping that leads to simple encoder/decoder realizations.

The space of all encoder input bytes B^8 is divided into 9 pairwise disjoint sets A_k, k=1,2,...,9. The elements of the subset A_k will be illustrated as specific locations marked "k" within the 16 × 16 array shown in Fig.3B, whose (i,j)-th location at the intersection of the i-th row, i=0,1,...,15, and the j-th column, j=0,1,...,15, corresponds to the byte $X=(X_1,X_2,...,X_8)$ where $X_8X_7X_6X_5$ and $X_1X_2X_3X_4$ are the binary representations of the decimal numbers i and j, respectively.

Similarly, the elements of D_k, k=1,2,...9, the images of A_k under the encoder mapping, are illustrated as specific locations marked "k" within the 16 × 32 array shown in Fig.3C, whose (i,j)-th location at the intersection of the i-th row, i=0,1,...,15, and the j-th column, j=0,1,...,31, corresponds to the 9-tuple $Y=(Y_1,Y_2,...,Y_9)$ where $Y_9Y_8Y_7Y_6$ and $Y_5Y_1Y_2Y_3Y_4$ are the binary representations of the decimal numbers i and j, respectively.

The pattern of _'s in D (4,6,5) represent all 9-tuples which cannot be used as codewords and is the same as the pattern of -'s in L_1 (4,6,5). The seven 9-tuples in L_1 which are not used as codewords in the (4,6,5) block

code presented here correspond to locations marked "■" in D (4,6,5). The partitions A_1 and D_1 are as follows: the patterns of 1's in A (4,6,5) and in the right half of D (4,6,5) are the same as the pattern of ■'s in the right half of L_1 (4,6,5). The other subsets A_k, k=2,3,...,9, and their images under the encoder mapping D_k, k=2,3,...,9, are simultaneously chosen such that the number of logic gates needed is as small as possible. Note that the choice of partitions A_k and D_k and of the one-to-one correspondences between A_k and D_k does not follow any rules except for k=1.

The encoder and decoder logic of the (4,6,5) code is shown in Tables 1 and 2, respectively. In the sequel, the overbar used in the tables denotes negation in the Boolean algebra. Intermediate variables V_i, i=1,2,...,7, for the encoder and W_i, i=1,2,...,9, for the decoder were used to reduce the number of logic gates. For the same purpose, superpartitions S_i, i=1,2,...,9, for the encoder and T_i, i=1,2,...,6, for the decoder were introduced as the union of partitions. The decoder includes an optional error flag F which is raised (i.e., F=1) when the decoder input is an illegal codeword.

MINIMUM VALUE OF CONSTRAINTS SUM:

In the following, σ denotes the sum of all three constraints, i.e.,

$$\sigma = G_0 + I_0 + G_1 . \quad (9)$$

The block codes presented so far achieve σ=15. To find out the smallest σ achievable with a (9,8) block code, the maximum number of freely concatenatable 9-tuples, n_{max}, was determined for all relevant constraints. The smallest σ achievable with an encoder for a (9,8) block code is only attained by (4,5,6) and (4,6,5).

ENCODER/DECODER FOR (4,4,9) CODE:

An encoder/decoder realization of the (4,4,9) block code was obtained by modifying the encoder/decoder logic associated with one block code given in U.S.Patent 4,707,681.

The conditions for the selection of output codewords are given by equation (6) and the following two equations:

$$(Y_1 + Y_3 + Y_5)\cdot(Y_5 + Y_7 + Y_9)\cdot(Y_2 + Y_4 + Y_6)\cdot(Y_4 + Y_6 + Y_8) = 1 , \quad (10)$$

and either

$$Y_1\cdot Y_2\cdot Y_3\cdot Y_4\cdot Y_5\cdot Y_6 + Y_5\cdot Y_6\cdot Y_7\cdot Y_8\cdot Y_9 = 0 , \quad (11a)$$

or

$$Y_1\cdot Y_2\cdot Y_3\cdot Y_4\cdot Y_5 + Y_4\cdot Y_5\cdot Y_6\cdot Y_7\cdot Y_8\cdot Y_9 = 0 , \quad (11b)$$

where (6), (10) and (11a) are used to generate a list L_1 of 259 codewords and (6), (12) and (11b) another list L_2 of 259 codewords. Equations (6) and (10) ensure G_0=4 and I_0=4, whereas (11a) or (11b) guarantees G_1=9. Note that the codewords in L_2 can be simply obtained by reversing the codewords in L_1. The (4,4,9) block code presented here comprises 256 codewords and was arbitrarily chosen to be a subset of L_1.

The 259 codewords in L_1 will be again illustrated as locations marked "■" within a 16 × 32 array L_1 (4,4,9) shown in Fig.4.

Tables 3 and 4 show encoder and decoder logic of the (4,4,9) code including an optional error detection flag F, which is raised when the decoder input is an illegal codeword. Note that the notation associated with the encoder/decoder partitions in U.S.Patent 4,707,681 has been adopted. The codeword assignments were chosen such that the logic realizations of the encoder and decoder are simple.

The two encoder/decoder realizations presented above correspond to optimum (9,8) block codes, i.e., none of the constraints G_0, I_0 and G_1 can be decreased further without relaxing another one.

FINITE-STATE ENCODERS:

The block encoders discussed in the previous section break the information sequence into blocks and handle them independently. In the following, a further encoder will be described which is a finite-state machine, i.e. in this encoder the output is not only dependent on the present input but also depends on previous encoded inputs. This encoder allows an even better reduction in run-lengths. Decoders which are used with the finite-state encoder need not be finite-state machines but can be block decoders.

ENCODER AND DECODER FOR (4,6,4) CODE:

The encoder for the (4,6,4) code is a finite-state machine where the encoder state is defined as the last bit of the preceding codeword. Similar to the decoders discussed above, the decoder associated with the (4,6,4) code can be realized as a block decoder which maps every 9-bit block at its input into a block of 8 bits. The

(4,6,4) code performs better than the block codes described above, i.e., it achieves $G_0=4$, $G_1=4$ and $\sigma=14$, and yet its encoder/decoder implementation is almost as simple as the ones for block codes.

In the following, the (4,6,4) code and its encoder/decoder realization are described. A list L of 230 codewords is obtained by selecting all 9-tuples $Y=(Y_1,Y_2,...,Y_9) \in B^9$ which satisfy (6),(7) and

$$Y_1 \cdot Y_2 \cdot Y_3 + Y_2 \cdot Y_3 \cdot Y_4 \cdot Y_5 \cdot Y_6 + Y_3 \cdot Y_4 \cdot Y_5 \cdot Y_6 \cdot Y_7 \cdot + Y_4 \cdot Y_5 \cdot Y_6 \cdot Y_7 \cdot Y_8 + Y_7 \cdot Y_8 \cdot Y_9 = 0 . \quad (12)$$

Conditions (6) and (7) ensure $G_0=4$ and $I_0=6$, whereas (12) guarantees $G_1=4$. The codewords in L do not begin with more than two 0's or 1's. However, if the last bit of the preceding codeword, henceforth denoted Z, is known, the next codeword may start with three or four 0's in the case Z=1 and three or four 1's in the case Z=0. There are 26 binary 9-tuples which begin with three or four 0's and do not violate (7), (12) and a modified version of (6) with $(Y_1+Y_2+Y_3)$ replaced by $(Y_1+Y_2+Y_3+Y_4+Y_5)$ to allow the codewords to begin with three or at most four 0's. Similarly, there are 35 binary 9-tuples which begin with three or four 1's and do not violate (6), (7) and a modified version of (12) with $(Y_1 \cdot Y_2 \cdot Y_3)$ replaced by $(Y_1 \cdot Y_2 \cdot Y_3 \cdot Y_4 \cdot Y_5)$ to allow the codewords to begin with three or at most four 1's.

The 230 codewords in L, the 26 codewords following Z=1 and the 35 binary 9-tuples following Z=0 will be illustrated as specific locations marked "■", "1" and "0", respectively, within the 16 × 32 array shown in Fig.5A, where the definition with respect to the correspondence between an array location and a 9-tuple in Fig.3A applies. The pattern of _'s in L (4,6,4) represents all 9-tuples which cannot be used as codewords. The space of 256 input bytes B^8 is divided into 13 pairwise disjoint sets A_k, k=1,2,...,9,a,b,c,d, where the subscript k is a hexadecimal number.

The elements of the subsets A_k, k=1,2,...,d, will be illustrated as specific locations marked "k" within the 16 × 16 array shown in Fig.5B where the definition with respect to the correspondence between an array location and an 8-tuple in Fig.3B applies. The 230 bytes in A(4,6,4) marked "k", k=1,2,...,9,a,b, are mapped independent of the encoder state Z into the 230 binary 9-tuples in L (4,6,4) marked "■" where the images of A_k, k=1,2,...,b, are D_k, k=1,2,...,b. The remaining 26 bytes in A(4,6,4) marked "c" and "d" are mapped either into the 26 codewords in L (4,6,4) marked "1" if Z=1 or into 26 codewords chosen out of the 35 binary 9-tuples in L(4,6,4) marked "0".

The elements of D_k, k=1,2,...,b, are illustrated as locations marked "k" within the 16 × 32 array shown in Fig.5C, where the definition with respect to the correspondence between an array location and a 9-tuple in Fig.3C applies. The images of the 26 bytes in A_k, k=c,d, depend on the encoder state Z and are specified by the locations marked "c" and "d" in the above array. The pattern of _'s in D (4,6,4) represent all 9-tuples which are not used as codewords and comprises the pattern of -'s in L (4,6,4) in addition to 35-26=9 unused 9-tuples marked by "0" in L(4,6,4).

The (4,6,4) encoder/decoder structure is shown in Fig.6(6A and 6B). The encoder logic 19 has, besides the eight inputs for bits $X_1,X_2,...,X_8$ and the nine outputs for bits $Y_1,Y_2,...,Y_9$, one extra input Z which is connected through a delay element 21 to the bit output Y_9. The delay element represents a time delay corresponding to the period of the byte clock. Thus, an input bit block X may be converted to two different output codewords Y, depending on the last bit of the previous output codeword.

The decoder need not to be a finite-state machine (sequential decoder). A normal block decoder 23 is sufficient which directly converts a codeword Y into an 8-bit block X. For the error detection logic 25, however, an extra input Z is required, receiving the last bit Y_9 of the previous codeword through a delay element 27.

The encoder and decoder logic is shown in Tables 5 and 6, respectively. Logic gate arrays implementing these logic equations are contained in the encoder logic 19, the decoder logic 23, and the error detection logic 25, respectively, of Figs.6A and 6B. Intermediate variables V_i, i=1,2,...,22, for the encoder and W_i, i=1,2,...,17, for the decoder were used to reduce the number of logic gates. An optional error flag F, which is raised when the decoder input is an illegal codeword, is provided.

Table 1. (4,6,5) Encoder Logic.

VARIABLES	PARTITIONS	SUPERPARTITIONS
$V_1 = X_1 + X_2 + X_3$	$P_1 = V_1 \cdot V_2 \cdot V_3 \cdot V_4 \cdot V_5 \cdot V_6$	$S_1 = P_5 + P_8 = \overline{V}_1$
$V_2 = \overline{X_1 \cdot X_2 \cdot X_3}$	$P_2 = \overline{P}_1 \cdot X_3 \cdot X_4 \cdot (\overline{X}_1 + \overline{X}_2)$	$S_2 = P_5 + P_7$
$V_3 = X_5 + X_7 + X_8$	$P_3 = \overline{P}_1 \cdot \overline{X}_2 \cdot \overline{X}_4 \cdot (X_1 + X_3)$	$S_3 = P_1 + S_2$
$V_4 = \overline{X_5 \cdot X_6 \cdot X_7 \cdot X_8}$	$P_4 = \overline{P}_1 \cdot \overline{P}_2 \cdot \overline{P}_3 \cdot V_1 \cdot V_2$	$S_4 = P_2 + P_6 + S_3$
$V_5 = X_2 + X_4 + X_6 + X_7$	$P_5 = \overline{V}_1 \cdot V_7$	$S_5 = P_8 + P_9 + P_9$
$V_6 = \overline{X_3 \cdot X_4 \cdot X_5 \cdot X_6 \cdot (X_2 + X_7)}$	$P_6 = \overline{V}_1 \cdot \overline{V}_7$	$S_6 = P_1 + P_2$
$V_7 = (X_5 + X_6) \cdot V_4$	$P_7 = \overline{V}_2 \cdot V_3 \cdot \overline{X}_5$	$S_7 = P_2 + P_3$
	$P_8 = \overline{V}_2 \cdot V_4 \cdot X_5$	$S_8 = P_4 + P_6$
	$P_9 = \overline{V}_2 \cdot \overline{(P_7 + P_8)}$	$S_9 = P_7 + S_1$

ENCODER OUTPUT

$$Y_1 = P_3 \cdot \overline{X}_1 + P_4 \cdot X_5 + P_9 + S_1 + S_6 \cdot X_1$$

$$Y_2 = P_3 \cdot \overline{X}_3 + P_4 \cdot X_5 \cdot \overline{X}_6 + P_6 + S_6 \cdot X_2 + S_6 \cdot X_4$$

$$Y_3 = P_1 \cdot X_3 + P_7 + S_7 + S_8$$

$$Y_4 = P_1 \cdot X_4 + P_8 \cdot X_4 + S_2 + S_8$$

$$Y_5 = P_1$$

$$Y_6 = P_3 + P_4 + P_6 + P_8 \cdot X_6 + S_3 \cdot X_6$$

$$Y_7 = P_2 \cdot X_6 + P_3 \cdot (X_5 + X_6 \cdot \overline{X}_8) + P_4 \cdot \overline{X}_4 + S_3 \cdot X_6 + S_5 \cdot X_5$$

$$Y_8 = P_4 \cdot (\overline{X}_1 \cdot \overline{X}_3 + \overline{X}_2) + P_6 \cdot X_7 \cdot \overline{X}_6 + P_8 \cdot X_4 + S_4 \cdot X_7 + S_7 \cdot \overline{X}_6$$

$$Y_9 = P_2 \cdot \overline{X}_5 + P_3 \cdot X_5 + P_4 \cdot X_1 + P_6 \cdot \overline{X_5 \cdot X_7} + P_8 + S_4 \cdot X_8$$

Table 2. (4,6,5) Decoder Logic.

VARIABLES	PARTITIONS	SUPERPARTITIONS
$W_1 = \overline{Y}_4 \cdot \overline{Y}_5$	$Q_1 = Y_5$	$T_1 = Q_1 + Q_2$
$W_2 = Y_4 \cdot \overline{Y}_5$	$Q_2 = W_1 \cdot W_5$	$T_2 = Q_7 + Q_8 + Q_9$
$W_3 = Y_3 \cdot Y_6$	$Q_3 = W_1 \cdot W_3$	$T_3 = Q_1 + Q_5 + Q_7$
$W_4 = Y_1 \cdot \overline{Y}_3$	$Q_4 = W_2 \cdot W_3$	$T_4 = Q_8 + T_3$
$W_5 = Y_3 \cdot \overline{Y}_6$	$Q_5 = W_2 \cdot W_4$	$T_5 = Q_6 + Q_8 + Q_9$
$W_6 = \overline{Y}_1 \cdot \overline{Y}_5$	$Q_6 = W_1 \cdot W_4$	$T_6 = Q_5 + Q_6 + Q_7$
$W_7 = \overline{Y_8 + Y_9}$	$Q_7 = W_6 \cdot W_5 \cdot Y_4$	
$W_8 = Y_7 \cdot Y_8$	$Q_8 = W_6 \cdot \overline{Y}_3$	
$W_9 = Y_7 \cdot Y_9$	$Q_9 = W_2 \cdot W_5 \cdot Y_1$	

DECODER OUTPUT

$$X_1 = Q_3 \cdot \overline{Y}_1 + Q_4 \cdot Y_9 + T_1 \cdot Y_1 + T_2$$

$$X_2 = Q_4 \cdot \overline{Y_8 \cdot Y_9} + T_1 \cdot Y_2 + T_2$$

$$X_3 = Q_1 \cdot Y_3 + Q_2 + Q_3 \cdot \overline{Y}_2 + Q_4 \cdot W_7 + T_2$$

$$X_4 = Q_1 \cdot Y_4 + Q_2 + Q_4 \cdot \overline{Y}_7 + Q_8 \cdot Y_4 + Q_9 \cdot Y_8 + T_6 \cdot Y_2$$

$$X_5 = Q_2 \cdot (Y_7 + \overline{Y}_9) + Q_3 \cdot (W_8 + W_9) + Q_4 \cdot (Y_1 + Y_2) + T_3 \cdot Y_6 + T_5 \cdot Y_7$$

$$X_6 = (Q_2 + T_3) \cdot Y_7 + Q_3 \cdot \overline{Y}_8 + Q_4 \cdot Y_1 + Q_6 \cdot W_7 + Q_8 \cdot Y_8 + Q_9 \cdot Y_7$$

$$X_7 = Q_2 \cdot W_8 + Q_3 \cdot W_9 + Q_4 \cdot Y_1 + Q_6 \cdot (Y_8 + W_7) + Q_8 \cdot Y_7 + T_4 \cdot Y_8$$

$$X_8 = Q_2 \cdot W_9 + (Q_3 + T_4) \cdot Y_9 + Q_4 \cdot Y_1 + Q_6 \cdot W_7 + Q_9 \cdot Y_7$$

ERROR DETECTION FLAG

$$\overline{F} = (Y_1 + Y_2 + Y_3) \cdot \overline{Y_1 \cdot Y_2 \cdot Y_3} \cdot (Y_7 + Y_8 + Y_9) \cdot \overline{Y_8 \cdot Y_7 \cdot Y_8 \cdot Y_9} \cdot (Y_2 + Y_4 + Y_8 + Y_9) \cdot$$
$$\{Y_5 \cdot \overline{Y_3 \cdot Y_4 \cdot Y_8 \cdot Y_7 \cdot (Y_2 + Y_9)} + \overline{Y}_5 \cdot (Y_1 + Y_3 + Y_7) \cdot (Y_8 + Y_7 + Y_3) \cdot (\overline{Y}_1 + Y_3 + Y_4 + Y_8) \cdot$$
$$(\overline{Y}_1 + \overline{Y}_3 + \overline{Y}_4 + Y_8 + Y_9) \cdot (Y_4 + Y_8 + Y_7 + Y_9)\}$$

Table 3. (4,4,9) Encoder Logic.

VARIABLES

$V_1 = X_1 \cdot X_2 \cdot X_3 \cdot X_4 \cdot X_5$

$V_2 = X_5 \cdot X_6 \cdot X_7 \cdot X_8$

$V_3 = X_4 \cdot V_2$

$V_4 = \overline{X}_4 \cdot V_2$

$V_5 = X_1 + X_3$

$V_6 = X_6 + X_8$

PARTITIONS

$M = (X_2 + V_5) \cdot (X_7 + V_6) \cdot (X_4 + X_5 + X_2 \cdot X_7) \cdot \overline{V}_1 \cdot \overline{V}_3 \cdot \overline{V}_4$

$H = \overline{X}_2 \cdot V_5 \cdot V_4$

$T_1 = V_1 \cdot V_6$ $\quad$ $T_2 = V_3 \cdot V_5$

$M_1 = M + V_5 \cdot X_4 \cdot \overline{T}_1$ $\quad$ $M_2 = M + V_6 \cdot X_5 \cdot \overline{T}_2 \cdot \overline{H}$

$N_1 = \overline{M} \cdot V_5 \cdot \overline{X}_4 \cdot \overline{H}$ $\quad$ $N_2 = \overline{M} \cdot V_6 \cdot \overline{X}_5$

$R_1 = \overline{M} \cdot \overline{V}_5 \cdot X_2$ $\quad$ $R_2 = \overline{M} \cdot \overline{V}_6 \cdot X_7$

$S_1 = \overline{M} \cdot \overline{V}_5 \cdot \overline{X}_2$ $\quad$ $S_2 = \overline{M} \cdot \overline{V}_6 \cdot \overline{X}_7$

ENCODER OUTPUT

$Y_1 = M_1 \cdot X_1 + N_1 \cdot X_1 + R_1 + S_1 \cdot X_4 + T_1 + H$

$Y_2 = M_1 \cdot X_2 + R_1 + S_1 + T_1 + H$

$Y_3 = M_1 \cdot X_3 + N_1 \cdot X_3 + R_1 + S_1 \cdot \overline{X}_4 + T_1 + H$

$Y_4 = M_1 \cdot X_4 + N_1 \cdot \overline{X}_2 + R_1 \cdot \overline{X}_4 + S_1 \cdot (\overline{X}_5 + S_2) + H$

$Y_5 = M$

$Y_6 = M_2 \cdot X_6 + N_2 \cdot \overline{X}_7 + R_2 \cdot \overline{X}_5 + S_2 \cdot (\overline{X}_4 + S_1)$

$Y_7 = M_2 \cdot X_6 + N_2 \cdot X_6 + R_2 + S_2 \cdot \overline{X}_5 + T_2 + H \cdot X_3$

$Y_8 = M_2 \cdot X_7 + R_2 + S_2 + T_2$

$Y_9 = M_2 \cdot X_6 + N_2 \cdot X_6 + R_2 + S_2 \cdot X_5 + T_2 + H \cdot X_1$

Table 4. (4,4,9) Decoder Logic.

VARIABLES

$W_1 = Y_1 \bullet Y_2 \bullet Y_3$

$W_2 = W_1 \bullet Y_4$

$W_3 = Y_7 \bullet Y_8 \bullet Y_9$

$W_4 = W_3 \bullet Y_6$

$W_5 = Y_4 \bullet Y_8 \bullet \overline{Y}_6$

$W_6 = Y_6 \bullet Y_2 \bullet \overline{Y}_4 + W_4 \bullet (\overline{Y}_2 \bullet \overline{Y}_4 + W_2)$

$W_7 = Y_4 \bullet Y_8 + W_4 + \overline{Y}_5 \bullet Y_8 \bullet \overline{Y_7 \bullet Y_9}$

$W_8 = Y_6 + W_2 + \overline{Y}_4 \bullet Y_2 \bullet \overline{Y_1 \bullet Y_3}$

PARTITIONS

$M_1 = Y_5 + W_3$

$N_1 = \overline{M}_1 \bullet \overline{Y}_2$

$R_1 = \overline{M}_1 \bullet W_1 \bullet W_7$

$T_1 = \overline{M}_1 \bullet W_1 \bullet \overline{Y}_4 \bullet \overline{R}_1$

$S_1 = \overline{M}_1 \bullet Y_2 \bullet \overline{Y_1 \bullet Y_3}$

$M_2 = Y_5 + W_6$

$N_2 = \overline{M}_2 \bullet \overline{Y}_8$

$R_2 = \overline{M}_2 \bullet W_3 \bullet W_8$

$T_2 = \overline{M}_2 \bullet W_3 \bullet \overline{R}_2$

$S_2 = \overline{M}_2 \bullet Y_8 \bullet \overline{Y_7 \bullet Y_9}$

$H = \overline{M}_1 \bullet W_2 \bullet \overline{Y}_6$

DECODER OUTPUT

$X_1 = M_1 \bullet Y_1 + N_1 \bullet Y_1 + T_1 + H \bullet Y_3$

$X_2 = M_1 \bullet Y_2 + N_1 \bullet \overline{Y}_4 + R_1 + T_1$

$X_3 = M_1 \bullet Y_3 + N_1 \bullet Y_3 + T_1 + H \bullet Y_7$

$X_4 = M_1 \bullet Y_4 + R_1 \bullet \overline{Y}_4 + T_1 + S_1 \bullet \overline{Y}_3$

$X_5 = M_2 \bullet Y_5 + R_2 \bullet \overline{Y}_6 + T_2 + S_2 \bullet \overline{Y}_7 + H$

$X_6 = M_2 \bullet Y_7 + N_2 \bullet Y_7 + T_2 + H$

$X_7 = M_2 \bullet Y_8 + N_2 \bullet \overline{Y}_8 + R_2 + T_2 + H$

$X_8 = M_2 \bullet Y_9 + N_2 \bullet Y_9 + T_2 + H$

ERROR DETECTION FLAG

$$\overline{F} = Y_5 \bullet (Y_1 + Y_2 + Y_3) \bullet (Y_4 + Y_6 + Y_2 \bullet Y_8) \bullet (Y_7 + Y_8 + Y_9) \bullet \overline{W}_4 \bullet \overline{W_2 \bullet Y_6} + \overline{Y}_5 \bullet (Y_1 + Y_3) \bullet (Y_7 + Y_9) \bullet (Y_4 + Y_6 + Y_2 \bullet W_3 + W_1 \bullet Y_8)$$

Table 5. (4,6,4) Encoder Logic.

VARIABLES

$V_1 = X_1 + X_2$

$V_2 = V_1 + X_3$

$V_3 = X_6 + X_7 + X_8$

$V_4 = \overline{X_1 \cdot X_2 \cdot X_3}$

$V_5 = \overline{X_6 \cdot X_7 \cdot X_8}$

$V_6 = X_5 + X_7$

$V_7 = V_6 + X_2 + X_4$

$V_8 = \overline{X_3 \cdot X_4}$

$V_9 = \overline{X_5 \cdot X_6}$

$V_{10} = \overline{X}_7 + V_9$

$V_{11} = (V_8 + V_9) \cdot (\overline{X}_2 + V_8 + \overline{X}_5) \cdot (V_{10} + \overline{X}_4)$

$V_{12} = V_2 \cdot V_4$

$V_{13} = V_3 \cdot V_5$

$V_{14} = V_{12} \cdot V_{13}$

$V_{15} = V_8 \cdot V_{12}$

$V_{16} = \overline{V}_4 \cdot V_{13}$

$V_{17} = V_2 + X_4$

$V_{18} = \overline{V}_2 \cdot X_4$

$V_{19} = \overline{V}_3 \cdot V_1$

$V_{20} = \overline{X}_2 \cdot (X_3 + \overline{X}_4)$

$V_{21} = X_4 + X_3 \cdot \overline{X}_5$

$V_{22} = X_4 + X_3 \cdot X_5$

PARTITIONS

$P_1 = V_7 \cdot V_{11} \cdot V_{14}$

$P_2 = V_{14} \cdot \overline{V}_{11}$

$P_3 = \overline{V}_5 \cdot V_{15}$

$P_4 = V_{16} \cdot V_9$

$P_5 = \overline{V}_5 \cdot \overline{V}_{15}$

$P_6 = \overline{V}_{17} \cdot \overline{V_6 \cdot V_3}$

$P_7 = \overline{V}_7 \cdot V_{14}$

$P_8 = V_{19} \cdot V_4 \cdot \overline{V}_{20}$

$P_9 = V_{19} \cdot V_{20}$

$P_{10} = \overline{V}_3 \cdot \overline{V}_1 \cdot V_{17} + V_{18} \cdot \overline{V}_{10} \cdot \overline{X}_6$

$P_{11} = \overline{V}_4 \cdot \overline{V}_3$

$P_{12} = V_{13} \cdot (\overline{V}_{17} \cdot V_6 + V_{18} \cdot V_{10})$

$P_{13} = V_{18} \cdot \overline{V}_9$

SUPERPARTITIONS

$S_1 = P_1 + P_2$

$S_2 = S_1 + P_3 + P_8$

$S_3 = S_1 + P_4 + P_{12}$

$S_4 = S_3 + P_{13}$

$S_5 = P_9 + P_{11}$

$S_6 = P_{12} + P_{13}$

$S_7 = P_6 + P_7$

$S_8 = P_3 + P_8$

ENCODER OUTPUT

$Y_1 = S_2 \cdot X_1 + P_4 \cdot X_4 + P_5 \cdot (X_1 + \overline{X}_3) + P_6 \cdot X_8 + P_7 \cdot \overline{X}_1 + S_5 \cdot X_5 + P_{10} \cdot X_3 + S_8 \cdot \overline{Z}$

$Y_2 = S_2 \cdot X_2 + P_3 \cdot X_2 + P_5 \cdot \overline{X}_6 + P_7 \cdot \overline{X}_3 + S_3 \cdot V_{21} + P_{10} \cdot X_4 + S_6 \cdot \overline{Z}$

$Y_3 = S_2 \cdot X_3 + P_4 + P_5 \cdot V_{12} + P_6 \cdot X_6 + P_7 + P_9 \cdot \overline{V}_{22} + P_{10} \cdot \overline{X}_3 + P_{11} \cdot \overline{X}_5 + S_8 \cdot \overline{Z}$

$Y_4 = S_2 \cdot X_4 + P_4 + P_5 \cdot \overline{X}_4 + P_6 \cdot X_8 + P_{10} + P_{11} \cdot V_{22} + P_{12} \cdot X_4 + P_{13} \cdot Z$

$Y_5 = P_1 + P_{12} \cdot Z + P_{13} \cdot \overline{Z}$

$Y_6 = S_3 \cdot X_6 + P_3 + P_6 + P_7 \cdot \overline{X}_6 + P_9 + P_{13} \cdot X_4$

$Y_7 = S_3 \cdot X_8 + S_6 + P_6 + S_8 + P_{13}$

$Y_8 = S_4 \cdot X_7 + P_3 \cdot X_6 + S_5 + S_7 + P_{10} \cdot X_6$

$Y_9 = S_4 \cdot X_8 + S_6 \cdot X_5 + P_6 + P_7 \cdot \overline{X}_8 + P_{10} \cdot \overline{X}_6$

Table 6. (4,6,4) Decoder Logic

VARIABLES	PARTITIONS	SUPERPARTITIONS
$W_1 = \overline{Y_1 \cdot Y_2 \cdot Y_3}$	$Q_1 = Y_5 \cdot W_1$	$T_1 = Q_1 + Q_2$
$W_2 = \overline{Y}_3 \cdot W_1$	$Q_2 = W_{10} \cdot Y_6 + W_6 \cdot (Y_8 \cdot W_{12} + W_3 \cdot \overline{Y}_8)$	$T_2 = Q_3 + Q_8$
$W_3 = Y_6 \cdot Y_7$	$Q_3 = W_2 \cdot \overline{Y}_8 \cdot W_3 \cdot \overline{W}_4 \cdot W_{12}$	$T_3 = T_1 + T_2$
$W_4 = Y_3 \cdot Y_4$	$Q_4 = \overline{Y}_3 \cdot \overline{Y}_2 \cdot W_4 \cdot \overline{W}_3$	$T_4 = Q_4 + Q_{11} + Q_{12}$
$W_5 = \overline{Y}_2 \cdot Y_3$	$Q_5 = W_2 \cdot \overline{Y}_7 \cdot \overline{W}_4 \cdot W_{13}$	$T_5 = T_1 + Q_4 + Q_{12}$
$W_6 = Y_4 \cdot W_3$	$Q_6 = \overline{Y}_5 \cdot W_8 \cdot (\overline{Y}_4 \cdot \overline{Y_2 \cdot Y_3} + \overline{Y}_1 \cdot W_9)$	$T_6 = Q_5 + Q_{11}$
$W_7 = Y_7 \cdot \overline{Y}_8$	$Q_7 = W_2 \cdot \overline{Y}_4 \cdot \overline{Y}_7 \cdot \overline{W}_{13}$	$T_7 = Q_5 + Q_{10}$
$W_8 = Y_2 \cdot W_7$	$Q_8 = W_2 \cdot W_{12} \cdot \overline{Y}_8 \cdot W_7 \cdot \overline{W}_5$	$T_8 = Q_3 + Q_5$
$W_9 = Y_2 \cdot \overline{Y}_3$	$Q_9 = W_2 \cdot \overline{Y}_4 \cdot Y_8 \cdot W_3$	$T_9 = T_5 + Q_{13}$
$W_{10} = \overline{Y}_1 \cdot Y_2 \cdot W_4$	$Q_{10} = \overline{Y}_5 \cdot \overline{Y}_6 \cdot \overline{Y}_7 \cdot (\overline{Y}_3 \cdot Y_4 + W_{10})$	
$W_{11} = Y_2 + Y_3$	$Q_{11} = \overline{Y}_5 \cdot W_8 \cdot (\overline{Y}_1 \cdot Y_2 \cdot Y_3 + Y_1 \cdot \overline{Y}_3 \cdot Y_4)$	
$W_{12} = W_{11} + Y_1$	$Q_{12} = \overline{Y}_5 \cdot \overline{W}_1$	
$W_{13} = Y_6 \cdot Y_8$	$Q_{13} = \overline{Y}_5 \cdot \overline{W}_{12} + Y_5 \cdot \overline{W}_1$	
$W_{14} = Y_4 + Y_8$		
$W_{15} = Y_3 + Y_7$		
$W_{16} = W_{14} + Y_8$		
$W_{17} = W_1 + \overline{Y}_4$		

DECODER OUTPUT

$$X_1 = T_3 \cdot Y_1 + T_4 + Q_5 \cdot Y_1 \cdot W_{11} + Q_7 \cdot \overline{Y}_1 + Q_8$$

$$X_2 = (T_3 + Q_5) \cdot Y_2 + T_4$$

$$X_3 = T_3 \cdot Y_3 + T_4 + Q_5 \cdot W_{11} + Q_7 \cdot \overline{Y}_2 + Q_9 \cdot \overline{W}_5 + Q_{10} \cdot Y_1$$

$$X_4 = (T_3 + Q_{12}) \cdot Y_4 + Q_4 \cdot Y_1 + Q_5 \cdot \overline{Y}_4 + Q_9 \cdot W_9 + Q_{10} \cdot Y_2 + Q_{11} \cdot Y_2 \cdot Y_4 + Q_{12} \cdot Y_6$$

$$X_5 = T_5 \cdot Y_5 + T_2 \cdot Y_9 + T_7 \cdot Y_8 + Q_6 \cdot Y_4 + T_8 \cdot Y_1 + Q_{13}$$

$$X_6 = T_5 \cdot Y_7 + T_8 + Q_6 \cdot Y_3 + Q_7 \cdot \overline{Y}_6 + Q_{10} \cdot \overline{Y}_9 + Q_{13}$$

$$X_7 = T_9 \cdot Y_9 + T_8 + Q_{10} \cdot \overline{Y}_9$$

$$X_8 = T_9 \cdot Y_9 + T_8 + Q_6 \cdot Y_1 + Q_7 \cdot \overline{Y}_9$$

ERROR DETECTION FLAG

$$\overline{F} = (Y_7 + Y_8 + Y_9) \cdot \overline{Y_7 \cdot Y_8 \cdot Y_9} \cdot (Y_2 + W_{16}) \cdot (Z + W_{12}) \cdot (\overline{Z} + W_1) \cdot \{Y_5 \cdot W_{17} \cdot \overline{W_{10} \cdot Y_5} \cdot \overline{W_5 \cdot Y_8} \cdot$$
$$(\overline{W}_4 + \overline{W}_7) \cdot (W_1 + Y_7) + \overline{Y}_5 \cdot (W_{12} + Y_4) \cdot (Y_1 + W_{15}) \cdot (Y_3 + W_{15}) \cdot (W_{12} + Y_7) \cdot (W_{11} + W_{14}) \cdot$$
$$(W_{11} + W_{15}) \cdot (Y_7 + W_{15}) \cdot (W_1 + W_{15}) \cdot (W_{17} + \overline{W_3 \cdot Y_5})\}$$

Claims

1. Method for encoding binary data being partitioned into blocks each having a first given number k of bits, into codewords each having a second given number n = k+ 1 of bits, said method comprising the steps of:
   - receiving the binary data;
   - producing sequences of fixed-length codewords,
     - said sequences having no more than a first preselected number (G_0) of consecutive zeros therein;
     - said sequences comprising two subsequences, one consisting only of odd bit positions, and another consisting only of even bit positions, each of said subsequences having no more than a second preselected number (l_0) of consecutive zeros therein;

   characterized in that
   - codewords are selected such that the maximum number of consecutive ones which can occur in any codeword sequence does not exceed a third preselected number (G1) which is a minimum possible in connection with the given combination of first preselected number (G_0) and second preselected number (l_0).

2. A method according to Claim 1, for encoding data from which by partitioning 8-bit data blocks were formed, into 9-bit codewords, comprising the steps of:
   - selecting from the set of all possible 9-bit words $Y=(Y_1,Y_2,...,Y_9)$a full subset (L_1) of 263 potential codewords satisfying the logic conditions

$$(Y_1+Y_2+Y_3)\cdot(Y_2+Y_3+Y_4+Y_5+Y_6)\cdot(Y_3+Y_4+Y_5+Y_6+Y_7)\cdot(Y_4+Y_5+Y_6+Y_7+Y_8)\cdot(Y_7+Y_8+Y_9)=1,$$
$$(Y_1+Y_3+Y_5+Y_7)\cdot(Y_3+Y_5+Y_7+Y_9)\cdot(Y_2+Y_4+Y_6+Y_8)=1,$$

   and

$$Y_1\cdot Y_2\cdot Y_3+Y_2\cdot Y_3\cdot Y_4\cdot Y_5\cdot Y_6\cdot Y_7+Y_3\cdot Y_4\cdot Y_5\cdot Y_6\cdot Y_7\cdot Y_8+Y_6\cdot Y_7\cdot Y_8\cdot Y_9=0\,;$$

   - selecting from the full subset of potential codewords, a partial subset of 256 actual codewords;
   - assigning to each one of the actual codewords of the partial subset, one of the 256 possible 8-bit data blocks; and
   - for encoding, replacing each data block by the assigned codeword;

   such that for the resulting sequence of codewords, said first preselected number (G_0) is four, said second preselected number (l_0) is six, and said third preselected number (G_1) is five.

3. A method according to Claim 1, for encoding data from which by partitioning 8-bit data blocks were formed, into 9-bit codewords, comprising the steps of:
   - selecting from the set of all possible 9-bit words $Y=(Y_1,Y_2,...,Y_9)$ a full subset (L_2) of 263 potential codewords satisfying the logic conditions

$$(Y_1+Y_2+Y_3)\cdot(Y_2+Y_3+Y_4+Y_5+Y_6)\cdot(Y_3+Y_4+Y_5+Y_6+Y_7)\cdot(Y_4+Y_5+Y_6+Y_7+Y_8)\cdot(Y_7+Y_8+Y_9)=1,$$
$$(Y_1+Y_3+Y_5+Y_7)\cdot(Y_3+Y_5+Y_7+Y_9)\cdot(Y_2+Y_4+Y_6+Y_8)=1\,,$$

   and

$$Y_1\cdot Y_2\cdot Y_3\cdot Y_4+Y_2\cdot Y_3\cdot Y_4\cdot Y_5\cdot Y_6\cdot Y_7+Y_3\cdot Y_4\cdot Y_5\cdot Y_6\cdot Y_7\cdot Y_8+Y_7\cdot Y_8\cdot Y_9=0\,;$$

   - selecting from the full subset of potential codewords, a partial subset of 256 actual codewords;
   - assigning to each one of the actual codewords of the partial subset, one of the 256 possible 8-bit data blocks; and
   - for encoding, replacing each data block by the assigned codeword;

   such that for the resulting sequence of codewords, said first preselected number (G_0) is four, said second preselected number (l_0) is six, and said third preselected number (G_1) is five.

4. A method according to Claim 1, for encoding data from which by partitioning 8-bit data blocks were formed, into 9-bit codewords, comprising the steps of:
   - selecting from the set of all possible 9-bit words $Y=(Y_1,Y_2,..., Y_9)$ a full subset (L_1) of 259 potential codewords satisfying the logic conditions

$$(Y_1+Y_2+Y_3)\cdot(Y_2+Y_3+Y_4+Y_5+Y_6)\cdot(Y_3+Y_4+Y_5+Y_6+Y_7)\cdot(Y_4+Y_5+Y_6+Y_7+Y_8)\cdot(Y_7+Y_8+Y_9)=1,$$
$$(Y_1+Y_3+Y_5)\cdot(Y_5+Y_7+Y_9)\cdot(Y_2+Y_4+Y_6)\cdot(Y_4+Y_6+Y_8)=1\,,$$

   and

$$Y_1\cdot Y_2\cdot Y_3\cdot Y_4\cdot Y_5\cdot Y_6+Y_5\cdot Y_6\cdot Y_7\cdot Y_8\cdot Y_9=0\,;$$

   - selecting from the full subset of potential codewords, a partial subset of 256 actual codewords;

- assigning to each one of the actual codewords of the partial subset, one of the 256 possible 8-bit data blocks; and
- for encoding, replacing each data block by the assigned codeword;

such that for the resulting sequence of codewords, said first preselected number (G_0) is four, said second preselected number (I_0) is four, and said third preselected number (G_1) is nine.

5. A method according to Claim 1, for encoding data from which by partitioning 8-bit data blocks were formed, into 9-bit codewords, comprising the steps of:
   - selecting from the set of all possible 9-bit word $Y=(Y_1,Y_2,...,Y_9)$ a full subset (L_2) of 259 potential codewords satisfying the logic conditions

$$(Y_1+Y_2+Y_3)\cdot(Y_2+Y_3+Y_4+Y_5+Y_6)\cdot(Y_3+Y_4+Y_5+Y_6+Y_7)\cdot(Y_4+Y_5+Y_6+Y_7+Y_8)\cdot(Y_7+Y_8+Y_9)=1,$$
$$(Y_1+Y_3+Y_5)\cdot(Y_5+Y_7+Y_9)\cdot(Y_2+Y_4+Y_6)\cdot(Y_4+Y_6+Y_8)=1,$$

   and

$$Y_1\cdot Y_2\cdot Y_3\cdot Y_4\cdot Y_5+Y_4\cdot Y_5\cdot Y_6\cdot Y_7\cdot Y_8\cdot Y_9=0;$$

   - selecting from the full subset of potential codewords, a partial subset of 256 actual codewords;
   - assigning to each one of the actual codewords of the partial subset, one of the 256 possible 8-bit blocks; and
   - for encoding, replacing each data block by the assigned codeword;

such that for the resulting sequence of codewords, said first preselected number (G_0) is four, said second preselected number (I_0) is four, and said third preselected number (G_1) is nine.

6. A method according to Claim 1, for encoding data from which by partitioning a sequence of 8-bit data block was formed, into a sequence of 9-bit codewords, comprising the steps of:
   - selecting from the set of all possible 9-bit words $Y=(Y_1,Y_2...,Y_9)$
   - a first full subset (L) of 230 actual codewords satisfying the logic conditions

$$(Y_1+Y_2+Y_3)\cdot(Y_2+Y_3+Y_4+Y_5+Y_6)\cdot(Y_3+Y_4+Y_5+Y_6+Y_7)\cdot(Y_4+Y_5+Y_6+Y_7+Y_8)\cdot(Y_7+Y_8+Y_9)=1,$$
$$(Y_1+Y_3+Y_5+Y_7)\cdot(Y_3+Y_5+Y_7+Y_9)\cdot(Y_2+Y_4+Y_6+Y_8)=1,$$

   and

$$Y_1\cdot Y_2\cdot Y_3\cdot+Y_2\cdot Y_3\cdot Y_4\cdot Y_5\cdot Y_6+Y_3\cdot Y_4\cdot Y_5\cdot Y_6\cdot Y_7+Y_4\cdot Y_5\cdot Y_6\cdot Y_7\cdot Y_8+Y_7\cdot Y_8\cdot Y_9=0;$$

   - a second subset (1) of 26 actual first conditional codewords with $Y_1=Y_2=Y_3=0$, satisfying the logic conditions

$$(Y_4+Y_5)\cdot(Y_4+Y_5+Y_6)\cdot(Y_4+Y_5+Y_6+Y_7)\cdot(Y_4+Y_5+Y_6+Y_7+Y_8)\cdot(Y_7+Y_8+Y_9)=1,$$
$$(Y_5+Y_7)\cdot(Y_5+Y_7+Y_9)\cdot(Y_4+Y_6+Y_8)=1,$$

   and

$$Y_4\cdot Y_5\cdot Y_6\cdot Y_7\cdot Y_8+Y_7\cdot Y_8\cdot Y_9=0;$$

   - a third subset (0) of 35 potential second conditional codewords with $Y_1=Y_2=Y_3=1$, satisfying the logic conditions

$$(Y_4+Y_5+Y_6+Y_7+Y_8)\cdot(Y_7+Y_8+Y_9)=1,$$

   and

$$Y_4\cdot Y_5+Y_4\cdot Y_5\cdot Y_6+Y_4\cdot Y_5\cdot Y_6\cdot Y_7+Y_4\cdot Y_5\cdot Y_6\cdot Y_7\cdot Y_8+Y_7\cdot Y_8\cdot Y_9=0;$$

   - selecting from the third subset of potential second conditional codewords, a partial subset of 26 actual conditional codewords;
   - assigning to each one of the actual codewords of the subsets, one of the 256 possible 8-bit data blocks, with 26 of said possible 8-bit data blocks each being assigned to one first conditional codeword and to one second conditional codeword; and
   - for encoding, replacing each data block by the assigned codeword, the selection of the conditional codewords depending on the binary value of the last bit of the previously generated codeword;

such that for the resulting sequence of codewords, said first preselected number (G_0) is four, said second preselected number (I_0) is six, and said third preselected number (G_1) is four.

7. Apparatus for encoding binary data into codewords, said apparatus comprising:
   - receiver means for receiving the binary data and for partitioning them into blocks each comprising a first given number k of bits:
   - encoder means, coupled to the receiver means, for producing sequences of fixed-length codewords each comprising a second given number n = k+1 of bits;
   - said sequences having no more than a first preselected number (G_0) of consecutive zeros therein;
   - said sequences comprising two subsequences, one consisting only of odd bit positions, and another consisting only of even bit positions, each of said subsequences having no more than a second pre-

selected number (I_0) of consecutive zeros therein; characterized by

- said sequences further having no more than a third preselected number (G_1) of consecutive ones therein;
- said third preselected number (G_1) having the minimum value possible in connection with the given combination of first preselected number (G_0) and second preselected number (I_0).

8. Apparatus according to Claim 7, comprising an encoder for converting 8-bit input data blocks $X=(X_1,X_2,...,X_8)$ into 9-bit output codewords $Y=(Y_1,Y_2,...,Y_9)$, said encoder being a logic gate array implementing the logic relations

$$Y_1 = P_3\cdot\overline{X}_1 + P_4\cdot X_8 + P_9 + S_1 + S_6\cdot X_1$$
$$Y_2 = P_3\cdot\overline{X}_3 + P_4\cdot X_5\cdot\overline{X}_6 + P_8 + S_6\cdot X_2 + S_9\cdot X_4$$
$$Y_3 = P_1\cdot X_3 + P_7 + S_7 + S_8$$
$$Y_4 = P_1\cdot X_4 + P_8\cdot X_4 + S_2 + S_8$$
$$Y_5 = P_1$$
$$Y_6 = P_3 + P_4 + P_6 + P_8\cdot X_8 + S_3\cdot X_5$$
$$Y_7 = P_2\cdot X_6 + P_3\cdot(X_5 + X_6\cdot\overline{X}_8) + P_4\cdot\overline{X}_4 + S_3\cdot X_6 + S_5\cdot X_5$$
$$Y_8 = P_4\cdot(\overline{X}_1\cdot\overline{X}_3 + \overline{X}_2) + P_6\cdot X_7\cdot\overline{X}_8 + P_9\cdot X_4 + S_4\cdot X_7 + S_7\cdot\overline{X}_5$$
$$Y_9 = P_2\cdot\overline{X}_5 + P_3\cdot X_8 + P_4\cdot X_1 + P_6\cdot\overline{X}_5\cdot\overline{X}_7 + P_9 + S_4\cdot X_8$$

with

$$V_1 = X_1 + X_2 + X_3 \qquad P_1 = V_1 \bullet V_2 \bullet V_3 \bullet V_4 \bullet V_5 \bullet V_6 \qquad S_1 = P_5 + P_6 = \overline{V}_1$$
$$V_2 = \overline{X_1 \bullet X_2 \bullet X_3} \qquad P_2 = \overline{P}_1 \bullet X_3 \bullet X_4 \bullet (\overline{X}_1 + \overline{X}_2) \qquad S_2 = P_5 + P_7$$
$$V_3 = X_6 + X_7 + X_8 \qquad P_3 = \overline{P}_1 \bullet \overline{X}_2 \bullet \overline{X}_4 \bullet (X_1 + X_3) \qquad S_3 = P_1 + S_2$$
$$V_4 = \overline{X_5 \bullet X_6 \bullet X_7 \bullet X_8} \qquad P_4 = \overline{P}_1 \bullet \overline{P}_2 \bullet \overline{P}_3 \bullet V_1 \bullet V_2 \qquad S_4 = P_2 + P_8 + S_3$$
$$V_5 = X_2 + X_4 + X_5 + X_7 \qquad P_5 = \overline{V}_1 \bullet V_7 \qquad S_5 = P_6 + P_8 + P_9$$
$$V_6 = \overline{X_3 \bullet X_4 \bullet X_5 \bullet X_6 \bullet (X_2 + X_7)} \qquad P_6 = \overline{V}_1 \bullet \overline{V}_7 \qquad S_6 = P_1 + P_2$$
$$V_7 = (X_6 + X_8) \bullet V_4 \qquad P_7 = \overline{V}_2 \bullet V_3 \bullet \overline{X}_6 \qquad S_7 = P_2 + P_3$$
$$P_8 = \overline{V}_2 \bullet V_4 \bullet X_5 \qquad S_8 = P_4 + P_9$$
$$P_9 = \overline{V}_2 \bullet \overline{(P_7 + P_8)} \qquad S_9 = P_7 + S_1$$

such that in response to an arbitrary sequence of input data blocks, for the resulting sequence of output codewords said first preselected number (G_0) is four, said second preselected number (I_0) is six, and said third preselected number (G_1) is five.

9. Apparatus according to Claim 7, comprising an encoder for converting 8-bit input data blocks $X=(X_1,X_2,...,X_8)$ into 9-bit output codewords $Y=(Y_1,Y_2,...Y_9)$, said encoder being a logic gate array implementing the logic relations

$$Y_1 = M_1\cdot X_1 + N_1\cdot X_1 + R_1 + S_1\cdot X_4 + T_1 + H$$
$$Y_2 = M_1\cdot X_2 + R_1 + S_1 + T_1 + H$$
$$Y_3 = M_1\cdot X_3 + N_1\cdot X_3 + R_1 + S_1\cdot\overline{X}_4 + T_1 + H$$
$$Y_4 = M_1\cdot X_4 + N_1\cdot\overline{X}_2 + R_1\cdot\overline{X}_4 + S_1\cdot(\overline{X}_5 + S_2) + H$$
$$Y_5 = M$$
$$Y_6 = M_2\cdot X_5 + N_2\cdot\overline{X}_7 + R_2\cdot\overline{X}_5 + S_2\cdot(\overline{X}_4 + S_1)$$
$$Y_7 = M_2\cdot X_6 + N_2\cdot X_6 + R_2 + S_2\cdot\overline{X}_5 + T_2 + H\cdot X_3$$
$$Y_8 = M_2\cdot X_7 + R_2 + S_2 + T_2$$
$$Y_9 = M_2\cdot X_8 + N_2\cdot X_8 + R_2 + S_2\cdot X_5 + T_2 + H\cdot X_1$$

with

$V_1 = X_1 \cdot X_2 \cdot X_3 \cdot X_4 \cdot X_5$

$V_2 = X_5 \cdot X_6 \cdot X_7 \cdot X_8$

$V_3 = X_4 \cdot V_2$

$V_4 = \overline{X}_4 \cdot V_2$

$V_5 = X_1 + X_3$

$V_6 = X_6 + X_8$

$M = (X_2 + V_5) \cdot (X_7 + V_6) \cdot (X_4 + X_5 + X_2 \cdot X_7) \cdot \overline{V}_1 \cdot \overline{V}_3 \cdot \overline{V}_4$

$H = \overline{X}_2 \cdot V_3 \cdot V_4$

$T_1 = V_1 \cdot V_6$ $\quad$ $T_2 = V_3 \cdot V_5$

$M_1 = M + V_3 \cdot X_4 \cdot \overline{T}_1$ $\quad$ $M_2 = M + V_6 \cdot X_5 \cdot \overline{T}_2 \cdot \overline{H}$

$N_1 = \overline{M} \cdot V_5 \cdot \overline{X}_4 \cdot \overline{H}$ $\quad$ $N_2 = \overline{M} \cdot V_6 \cdot \overline{X}_5$

$R_1 = \overline{M} \cdot \overline{V}_5 \cdot X_2$ $\quad$ $R_2 = \overline{M} \cdot \overline{V}_6 \cdot X_7$

$S_1 = \overline{M} \cdot \overline{V}_5 \cdot \overline{X}_2$ $\quad$ $S_2 = \overline{M} \cdot \overline{V}_6 \cdot \overline{X}_7$

such that in response to an arbitrary sequence of input data blocks, for the resulting sequence of output codewords said first preselected number (G_0) is four, said second preselected number (I_0) is four, and said third preselected number (G_1) is nine.

10. Apparatus according to Claim 7, comprising an encoder for converting a sequence of 8-bit input data blocks $X=(X_1,X_2,...,X_8)$ into 9-bit output codewords $Y=(Y_1,Y_2,...,Y_9)$

said encoder being a sequential encoder (Fig.6A) having nine outputs for furnishing the nine bits of each output codeword; and having nine inputs, eight of them being connected to receive the eight bits of an input data block, and the nineth input (Z) being connected to the nineth output through a delay element (21);

said encoder comprising a logic gate array implementing the logic relations

$$Y_1 = S_2 \cdot X_1 + P_4 \cdot X_4 + P_5 \cdot (X_1 + \overline{X}_3) + P_6 \cdot X_8 + P_7 \cdot \overline{X}_1 + S_5 \cdot X_5 + P_{10} \cdot X_3 + S_6 \cdot \overline{Z}$$
$$Y_2 = S_2 \cdot X_2 + P_5 \cdot X_2 + P_6 \cdot \overline{X}_6 + P_7 \cdot \overline{X}_3 + S_5 \cdot V_{21} + P_{10} \cdot X_4 + S_6 \cdot \overline{Z}$$
$$Y_3 = S_2 \cdot X_3 + P_4 + P_5 \cdot V_{12} + P_6 \cdot X_6 + P_7 + P_9 \cdot \overline{V}_{22} + P_{10} \cdot \overline{X}_3 + P_{11} \cdot \overline{X}_5 + S_6 \cdot \overline{Z}$$
$$Y_4 = S_2 \cdot X_4 + P_4 + P_5 \cdot \overline{X}_4 + P_6 \cdot X_5 + P_{10} + P_{11} \cdot V_{22} + P_{12} \cdot X_4 + P_{13} \cdot Z$$
$$Y_5 = P_1 + P_{12} \cdot Z + P_{13} \cdot \overline{Z}$$
$$Y_6 = S_3 \cdot X_5 + P_3 + P_5 + P_7 \cdot \overline{X}_6 + P_9 + P_{13} \cdot X_4$$
$$Y_7 = S_3 \cdot X_6 + S_8 + P_6 + S_5 + P_{13}$$
$$Y_8 = S_4 \cdot X_7 + P_5 \cdot X_5 + S_5 + S_7 + P_{10} \cdot X_5$$
$$Y_9 = S_4 \cdot X_8 + S_8 \cdot X_5 + P_5 + P_7 \cdot \overline{X}_8 + P_{10} \cdot \overline{X}_6$$

with

$V_1 = X_1 + X_2$

$V_2 = V_1 + X_3$

$V_3 = X_6 + X_7 + X_8$

$V_4 = \overline{X_1 \cdot X_2 \cdot X_3}$

$V_5 = \overline{X_6 \cdot X_7 \cdot X_8}$

$V_6 = X_5 + X_7$

$V_7 = V_6 + X_2 + X_4$

$V_8 = \overline{X_3 \cdot X_4}$

$V_9 = \overline{X_5 \cdot X_6}$

$V_{10} = \overline{X}_7 + V_9$

$V_{11} = (V_8 + V_9) \cdot (\overline{X}_2 + V_8 + \overline{X}_5) \cdot (V_{10} + \overline{X}_4)$

$V_{12} = V_2 \cdot V_4$

$V_{13} = V_3 \cdot V_5$

$V_{14} = V_{12} \cdot V_{13}$

$V_{15} = V_8 \cdot V_{12}$

$V_{16} = \overline{V}_4 \cdot V_{13}$

$V_{17} = V_2 + X_4$

$V_{18} = \overline{V}_2 \cdot X_4$

$V_{19} = \overline{V}_3 \cdot V_1$

$V_{20} = \overline{X}_2 \cdot (X_3 + \overline{X}_4)$

$V_{21} = X_4 + X_3 \cdot \overline{X}_5$

$V_{22} = X_4 + X_3 \cdot X_5$

and

$$P_1 = V_7 \cdot V_{11} \cdot V_{14} \qquad P_8 = V_{19} \cdot V_4 \cdot \overline{V}_{20} \qquad S_1 = P_1 + P_2$$

$$P_2 = V_{14} \cdot \overline{V}_{11} \qquad P_9 = V_{19} \cdot V_{20} \qquad S_2 = S_1 + P_3 + P_8$$

$$P_3 = \overline{V}_5 \cdot V_{15} \qquad P_{10} = \overline{V}_3 \cdot \overline{V}_1 \cdot V_{17} + V_{18} \cdot \overline{V}_{10} \cdot \overline{X}_8 \qquad S_3 = S_1 + P_4 + P_{12}$$

$$P_4 = V_{16} \cdot V_9 \qquad P_{11} = \overline{V}_4 \cdot \overline{V}_3 \qquad S_4 = S_3 + P_{13}$$

$$P_5 = \overline{V}_5 \cdot \overline{V}_{13} \qquad P_{12} = V_{13} \cdot (\overline{V}_{17} \cdot V_6 + V_{18} \cdot V_{10}) \qquad S_5 = P_9 + P_{11}$$

$$P_6 = \overline{V}_{17} \cdot \overline{V_6 \cdot V_3} \qquad P_{13} = V_{18} \cdot \overline{V}_9 \qquad S_6 = P_{12} + P_{13}$$

$$P_7 = \overline{V}_7 \cdot V_{14} \qquad S_7 = P_6 + P_7$$

$$S_8 = P_3 + P_8$$

such that in response to an arbitrary sequence of input data blocks, for the resulting sequence of output codewords said first preselected number (G_0) is four, said second preselected number (I_0) is six, and said third preselected number (G_1) is four.

11. Decoder apparatus for converting 9-bit input codewords $Y=(Y_1,Y_2...,Y_9)$ generated by encoder apparatus according to Claim 8, into 8-bit output data blocks $X=(X_1,X_2...,X_8)$, said decoder apparatus comprising a logic gate array implementing the logic relations

$$X_1 = Q_3 \cdot \overline{Y}_1 + Q_4 \cdot Y_9 + T_1 \cdot Y_1 + T_2$$
$$X_2 = Q_4 \cdot \overline{Y_8 \cdot Y_9} + T_1 \cdot Y_2 + T_2$$
$$X_3 = Q_1 \cdot Y_3 + Q_2 + Q_3 \cdot \overline{Y}_2 + Q_4 \cdot W_7 + T_2$$
$$X_4 = Q_1 \cdot Y_4 + Q_2 + Q_4 \cdot \overline{Y}_7 + Q_8 \cdot Y_4 + Q_9 \cdot Y_8 + T_6 \cdot Y_2$$
$$X_5 = Q_2 \cdot (Y_7 + \overline{Y}_9) + Q_3 \cdot (W_8 + W_9) + Q_4 \cdot (Y_1 + Y_2) + T_3 \cdot Y_6 + T_5 \cdot Y_7$$
$$X_6 = (Q_2 + T_3) \cdot Y_7 + Q_3 \cdot \overline{Y}_8 + Q_4 \cdot Y_1 + Q_6 \cdot W_7 + Q_8 \cdot Y_6 + Q_9 \cdot Y_7$$
$$X_7 = Q_2 \cdot W_8 + Q_3 \cdot W_9 + Q_4 \cdot Y_1 + Q_6 \cdot (Y_8 + W_7) + Q_9 \cdot Y_7 + T_4 \cdot Y_8$$
$$X_8 = Q_2 \cdot W_9 + (Q_3 + T_4) \cdot Y_9 + Q_4 \cdot Y_1 + Q_6 \cdot W_7 + Q_9 \cdot Y_7$$

with

$$W_1 = \overline{Y}_4 \cdot \overline{Y}_5 \qquad Q_1 = Y_5 \qquad T_1 = Q_1 + Q_2$$

$$W_2 = Y_4 \cdot \overline{Y}_5 \qquad Q_2 = W_1 \cdot W_5 \qquad T_2 = Q_7 + Q_8 + Q_9$$

$$W_3 = Y_3 \cdot Y_8 \qquad Q_3 = W_1 \cdot W_3 \qquad T_3 = Q_1 + Q_5 + Q_7$$

$$W_4 = Y_1 \cdot \overline{Y}_3 \qquad Q_4 = W_2 \cdot W_3 \qquad T_4 = Q_8 + T_3$$

$$W_5 = Y_3 \cdot \overline{Y}_8 \qquad Q_5 = W_2 \cdot W_4 \qquad T_5 = Q_6 + Q_8 + Q_9$$

$$W_6 = \overline{Y}_1 \cdot \overline{Y}_5 \qquad Q_6 = W_1 \cdot W_4 \qquad T_6 = Q_5 + Q_6 + Q_7$$

$$W_7 = \overline{Y_8 + Y_9} \qquad Q_7 = W_6 \cdot W_5 \cdot Y_4$$

$$W_8 = Y_7 \cdot Y_8 \qquad Q_8 = W_5 \cdot \overline{Y}_3$$

$$W_9 = Y_7 \cdot Y_9 \qquad Q_9 = W_2 \cdot W_6 \cdot Y_1$$

12. Decoder apparatus for converting 9-bit input codewords $Y=(Y_1,Y_2...,Y_9)$ generated by encoder apparatus according to Claim 9, into 8-bit output data blocks $X=(X_1,X_2,...,X_8)$, said decoder apparatus comprising a logic gate array implementing the logic relations

$$X_1 = M_1 \cdot Y_1 + N_1 \cdot Y_1 + T_1 + H \cdot Y_9$$

$$X_2 = M_1 \cdot Y_2 + N_1 \cdot \overline{Y}_4 + R_1 + T_1$$
$$X_3 = M_1 \cdot Y_3 + N_1 \cdot Y_3 + T_1 + H \cdot Y_7$$
$$X_4 = M_1 \cdot Y_4 + R_1 \cdot \overline{Y}_4 + T_1 + S_1 \cdot \overline{Y}_3$$
$$X_5 = M_2 \cdot Y_6 + R_2 \cdot \overline{Y}_6 + T_2 + S_2 \cdot \overline{Y}_7 + H$$
$$X_6 = M_2 \cdot Y_7 + N_2 \cdot Y_7 + T_2 + H$$
$$X_7 = M_2 \cdot Y_8 + N_2 \cdot \overline{Y}_6 + R_2 + T_2 + H$$
$$X_8 = M_2 \cdot Y_9 + N_2 \cdot Y_9 + T_2 + H$$

with

$$W_1 = Y_1 \bullet Y_2 \bullet Y_3 \qquad M_1 = Y_5 + W_5 \qquad M_2 = Y_5 + W_6$$
$$W_2 = W_1 \bullet Y_4 \qquad N_1 = \overline{M}_1 \bullet \overline{Y}_2 \qquad N_2 = \overline{M}_2 \bullet \overline{Y}_8$$
$$W_3 = Y_7 \bullet Y_8 \bullet Y_9 \qquad R_1 = \overline{M}_1 \bullet W_1 \bullet W_7 \qquad R_2 = \overline{M}_2 \bullet W_3 \bullet W_8$$
$$W_4 = W_3 \bullet Y_6 \qquad T_1 = \overline{M}_1 \bullet W_1 \bullet \overline{Y}_4 \bullet \overline{R}_1 \qquad T_2 = \overline{M}_2 \bullet W_3 \bullet \overline{R}_2$$
$$W_5 = Y_4 \bullet Y_8 \bullet \overline{Y}_6 \qquad S_1 = \overline{M}_1 \bullet Y_2 \bullet \overline{Y_1 \bullet Y_3} \qquad S_2 = \overline{M}_2 \bullet Y_8 \bullet \overline{Y_7 \bullet Y_9}$$
$$W_6 = Y_6 \bullet Y_2 \bullet \overline{Y}_4 + W_4 \bullet (\overline{Y}_2 \bullet \overline{Y}_4 + W_2)$$
$$H = \overline{M}_1 \bullet W_2 \bullet \overline{Y}_6$$
$$W_7 = Y_4 \bullet Y_6 + W_4 + \overline{Y}_6 \bullet Y_8 \bullet \overline{Y_7 \bullet Y_9}$$
$$W_8 = Y_6 + W_2 + \overline{Y}_4 \bullet Y_2 \bullet \overline{Y_1 \bullet Y_3}$$

13. Decoder apparatus for converting 9-bit input codewords $Y=(Y_1,Y_2...,Y_9)$ generated by encoder apparatus according to Claim 10, into 8-bit output data blocks $X=(X_1,X_2...,X_8)$, said decoder apparatus comprising a logic gate array implementing the logic relations

$$X_1 = T_3 \cdot Y_1 + T_4 + Q_5 \cdot Y_1 \cdot W_{11} + Q_7 \cdot \overline{Y}_1 + Q_9$$
$$X_2 = (T_3 + Q_5) \cdot Y_2 + T_4$$
$$X_3 = T_3 \cdot Y_3 + T_4 + Q_5 \cdot W_{11} + Q_7 \cdot \overline{Y}_2 + Q_9 + \overline{W}_5 + Q_{10} \cdot Y_1$$
$$X_4 = (T_3 + Q_{12}) \cdot Y_4 + Q_4 \cdot Y_1 + Q_5 \cdot \overline{Y}_4 + Q_9 \cdot W_9 + Q_{10} \cdot Y_2 + Q_{11} \cdot Y_2 \cdot Y_4 + Q_{13} \cdot Y_6$$
$$X_5 = T_5 \cdot Y_6 + T_2 \cdot Y_9 + T_7 \cdot Y_8 + Q_6 \cdot Y_4 + T_6 \cdot Y_1 + Q_{13}$$
$$X_6 = T_5 \cdot Y_7 + T_8 + Q_6 \cdot Y_3 + Q_7 \cdot \overline{Y}_6 + Q_{10} \cdot \overline{Y}_9 + Q_{13}$$
$$X_7 = T_9 \cdot Y_8 + T_8 + Q_{10} \cdot \overline{Y}_9$$
$$X_8 = T_9 \cdot Y_9 + T_8 + Q_6 \cdot Y_1 + Q_7 \cdot \overline{Y}_9$$

with

$$W_1 = \overline{Y_1 \cdot Y_2 \cdot Y_3}$$
$$W_2 = \overline{Y}_3 \cdot W_1$$
$$W_3 = Y_6 \cdot Y_7$$
$$W_4 = Y_3 \cdot Y_4$$
$$W_5 = \overline{Y}_2 \cdot Y_3$$
$$W_6 = Y_4 \cdot W_3$$
$$W_7 = Y_7 \cdot \overline{Y}_6$$
$$W_8 = Y_6 \cdot W_7$$
$$W_9 = Y_2 \cdot \overline{Y}_3$$
$$W_{10} = \overline{Y}_1 \cdot Y_2 \cdot W_4$$
$$W_{11} = Y_2 + Y_3$$
$$W_{12} = W_{11} + Y_1$$
$$W_{13} = Y_6 \cdot Y_5$$
$$W_{14} = Y_4 + Y_6$$
$$W_{15} = Y_3 + Y_7$$
$$W_{16} = W_{14} + Y_8$$
$$W_{17} = W_1 + \overline{Y}_4$$

$$Q_1 = Y_3 \cdot W_1$$
$$Q_2 = W_{10} \cdot Y_6 + W_6 \cdot (Y_8 \cdot W_{12} + W_5 \cdot \overline{Y}_6)$$
$$Q_3 = W_2 \cdot \overline{Y}_8 \cdot W_3 \cdot \overline{W}_4 \cdot W_{12}$$
$$Q_4 = \overline{Y}_3 \cdot \overline{Y}_2 \cdot W_4 \cdot \overline{W}_2$$
$$Q_5 = W_2 \cdot \overline{Y}_7 \cdot \overline{W}_4 \cdot W_{13}$$
$$Q_6 = \overline{Y}_3 \cdot W_5 \cdot (\overline{Y}_4 \cdot \overline{Y_2 \cdot Y_3} + \overline{Y}_1 \cdot W_3)$$
$$Q_7 = W_2 \cdot \overline{Y}_4 \cdot \overline{Y}_7 \cdot \overline{W}_{13}$$
$$Q_8 = W_2 \cdot W_{12} \cdot \overline{Y}_9 \cdot W_7 \cdot \overline{W}_5$$
$$Q_9 = W_2 \cdot \overline{Y}_4 \cdot Y_6 \cdot W_3$$
$$Q_{10} = \overline{Y}_3 \cdot \overline{Y}_6 \cdot \overline{Y}_7 \cdot (\overline{Y}_3 \cdot Y_4 + W_{10})$$
$$Q_{11} = \overline{Y}_5 \cdot W_9 \cdot (\overline{Y}_1 \cdot Y_2 \cdot Y_3 + Y_1 \cdot \overline{Y}_3 \cdot Y_4)$$
$$Q_{12} = \overline{Y}_5 \cdot \overline{W}_1$$
$$Q_{13} = \overline{Y}_5 \cdot \overline{W}_{12} + Y_6 \cdot \overline{W}_1$$

$$T_1 = Q_1 + Q_2$$
$$T_2 = Q_3 + Q_8$$
$$T_3 = T_1 + T_2$$
$$T_4 = Q_4 + Q_{11} + Q_{13}$$
$$T_5 = T_1 + Q_4 + Q_{12}$$
$$T_6 = Q_9 + Q_{11}$$
$$T_7 = Q_5 + Q_{10}$$
$$T_8 = Q_3 + Q_5$$
$$T_9 = T_5 + Q_{13}$$

Patentansprüche

1. Verfahren zum Codieren binärer Daten, die in Blöcke unterteilt sind, von denen jeder eine erste gegebene Anzahl k an Bits hat, in Codeworte, von denen jedes eine zweite gegebene Anzahl n = k + 1 Bits hat, mit den Schritten:
   - Empfangen der binären Daten;
   - Erzeugen von Folgen von Codeworten fester Länge,
     - die Folgen enthalten nicht mehr als eine erste vorgewählte Anzahl (G_0) aufeinanderfolgender Nullen;
     - die Folgen umfassen zwei Subfolgen, von denen eine nur aus ungeraden Bitpositionen und die andere nur aus geraden Bitpositionen besteht, wobei jede der Subfolgen nicht mehr als eine zweite vorgewählte Anzahl (I_0) aufeinanderfolgender Nullen enthält;

   dadurch gekennzeichnet, daß
   - die Codeworte so ausgewählt werden, daß die maximale Anzahl aufeinanderfolgender Einsen, welche in einer beliebigen Codewortfolge auftreten kann, eine dritte vorgewählte Anzahl (G_1) nicht überschreitet, welche ein Minimum darstellt, das in Verbindung mit der ersten vorgewählten Anzahl (G_0) und der zweiten vorgewählten Anzahl (I_0) möglicherweise auftreten kann.

2. Verfahren nach Anspruch 1 zum Codieren von Daten, von welchen aus den unterteilten 8-Bit-Datenblöcken 9-Bit-Codeworte gebildet werden, umfassend die folgenden Schritte:

$$W_1 = \overline{Y_1 \cdot Y_2 \cdot Y_3}$$
$$W_2 = \overline{Y}_3 \cdot W_1$$
$$W_3 = Y_6 \cdot Y_7$$
$$W_4 = Y_3 \cdot Y_4$$
$$W_5 = \overline{Y}_2 \cdot Y_3$$
$$W_6 = Y_4 \cdot W_2$$
$$W_7 = Y_7 \cdot \overline{Y}_6$$
$$W_8 = Y_8 \cdot W_7$$
$$W_9 = Y_2 \cdot \overline{Y}_3$$
$$W_{10} = \overline{Y}_1 \cdot Y_2 \cdot W_4$$
$$W_{11} = Y_2 + Y_3$$
$$W_{12} = W_{11} + Y_1$$
$$W_{13} = Y_5 \cdot Y_9$$
$$W_{14} = Y_4 + Y_6$$
$$W_{15} = Y_3 + Y_7$$
$$W_{16} = W_{14} + Y_8$$
$$W_{17} = W_1 + \overline{Y}_4$$

$$Q_1 = Y_5 \cdot W_1$$
$$Q_2 = W_{10} \cdot Y_6 + W_6 \cdot (Y_8 \cdot W_{12} + W_5 \cdot \overline{Y}_8)$$
$$Q_3 = W_2 \cdot \overline{Y}_8 \cdot W_3 \cdot \overline{W}_4 \cdot W_{12}$$
$$Q_4 = \overline{Y}_5 \cdot \overline{Y}_2 \cdot W_4 \cdot \overline{W}_3$$
$$Q_5 = W_2 \cdot \overline{Y}_7 \cdot \overline{W}_4 \cdot W_{12}$$
$$Q_6 = \overline{Y}_5 \cdot W_8 \cdot (\overline{Y}_4 \cdot \overline{Y_2 \cdot Y_3} + \overline{Y}_1 \cdot W_9)$$
$$Q_7 = W_2 \cdot \overline{Y}_4 \cdot \overline{Y}_7 \cdot \overline{W}_{12}$$
$$Q_8 = W_2 \cdot W_{12} \cdot \overline{Y}_8 \cdot W_7 \cdot \overline{W}_5$$
$$Q_9 = W_2 \cdot \overline{Y}_4 \cdot Y_8 \cdot W_3$$
$$Q_{10} = \overline{Y}_5 \cdot \overline{Y}_6 \cdot \overline{Y}_7 \cdot (\overline{Y}_3 \cdot Y_4 + W_{10})$$
$$Q_{11} = \overline{Y}_5 \cdot W_9 \cdot (\overline{Y}_1 \cdot Y_2 \cdot Y_3 + Y_1 \cdot \overline{Y}_3 \cdot Y_4)$$
$$Q_{12} = \overline{Y}_5 \cdot \overline{W}_1$$
$$Q_{13} = \overline{Y}_5 \cdot \overline{W}_{12} + Y_6 \cdot \overline{W}_1$$

$$T_1 = Q_1 + Q_2$$
$$T_2 = Q_3 + Q_8$$
$$T_3 = T_1 + T_2$$
$$T_4 = Q_4 + Q_{11} + Q_{13}$$
$$T_5 = T_1 + Q_4 + Q_{12}$$
$$T_6 = Q_9 + Q_{11}$$
$$T_7 = Q_5 + Q_{10}$$
$$T_8 = Q_3 + Q_5$$
$$T_9 = T_5 + Q_{13}$$

Patentansprüche

1. Verfahren zum Codieren binärer Daten, die in Blöcke unterteilt sind, von denen jeder eine erste gegebene Anzahl k an Bits hat, in Codeworte, von denen jedes eine zweite gegebene Anzahl n = k + 1 Bits hat, mit den Schritten:
   - Empfangen der binären Daten;
   - Erzeugen von Folgen von Codeworten fester Länge,
     - die Folgen enthalten nicht mehr als eine erste vorgewählte Anzahl (G_0) aufeinanderfolgender Nullen;
     - die Folgen umfassen zwei Subfolgen, von denen eine nur aus ungeraden Bitpositionen und die andere nur aus geraden Bitpositionen besteht, wobei jede der Subfolgen nicht mehr als eine zweite vorgewählte Anzahl (I_0) aufeinanderfolgender Nullen enthält;

   dadurch gekennzeichnet, daß
   - die Codeworte so ausgewählt werden, daß die maximale Anzahl aufeinanderfolgender Einsen, welche in einer beliebigen Codewortfolge auftreten kann, eine dritte vorgewählte Anzahl (G_1) nicht überschreitet, welche ein Minimum darstellt, das in Verbindung mit der ersten vorgewählten Anzahl (G_0) und der zweiten vorgewählten Anzahl (I_0) möglicherweise auftreten kann.

2. Verfahren nach Anspruch 1 zum Codieren von Daten, von welchen aus den unterteilten 8-Bit-Datenblöcken 9-Bit-Codeworte gebildet werden, umfassend die folgenden Schritte:

* C L M S *

- Auswählen eines vollen Subsatzes (L_1), bestehend aus 263 möglichen Codeworten, aus dem Satz aller möglichen 9-Bit-Worte $Y = (Y_1, Y_2, \ldots, Y_9)$ wobei der Subsatz die folgenden logischen Bedingungen erfüllt:

$$(Y_1 + Y_2 + Y_3)\cdot(Y_2 + Y_3 + Y_4 + Y_5 + Y_6)\cdot(Y_3 + Y_4 + Y_5 + Y_6 + Y_7)\cdot(Y_4 + Y_5 + Y_6 + Y_7 + Y_8)\cdot(Y_7 + Y_8 + Y_9) = 1,$$

$$(Y_1 + Y_3 + Y_5 + Y_7)\cdot(Y_3 + Y_5 + Y_7 + Y_9)\cdot(Y_2 + Y_4 + Y_6 + Y_8) = 1$$

und

$$Y_1\cdot Y_2\cdot Y_3 + Y_2\cdot Y_3\cdot Y_4\cdot Y_5\cdot Y_6\cdot Y_7 + Y_3\cdot Y_4\cdot Y_5\cdot Y_6\cdot Y_7\cdot Y_8 + Y_6\cdot Y_7\cdot Y_8\cdot Y_9 = 0;$$

- Auswählen eines Teil-Subsatzes, bestehend aus 256 aktuellen Codeworten, aus dem vollen Subsatz der möglichen Codeworte;
- Zuordnen von je einem der 256 möglichen 8-Bit-Datenblöcke zu je einem der aktuellen Codeworte des TeilSubsatzes und
- Codieren der Datenblöcke dadurch, daß jeder Datenblock durch das zugeordnete Codewort ersetzt wird;

so daß für die sich ergebende Folge von Codeworten die erste vorgewählte Anzahl (G_0) die Vier ist, die zweite vorgewählte Anzahl (I_0) die Sechs ist und die dritte vorgewählte Anzahl (G_1) die Fünf ist.

3. Verfahren nach Anspruch 1 zum Codieren von Daten, von welchen aus den unterteilten 8-Bit-Datenblöcken 9-Bit-Codeworte gebildet werden, umfassend die folgenden Schritte:
   - Auswählen eines vollen Subsatzes (L_2), bestehend aus 263 möglichen Codeworten, aus dem Satz aller möglichen 9-Bit-Worte $Y = (Y_1, Y_2, \ldots, Y_9)$ wobei der Subsatz die folgenden logischen Bedingungen erfüllt:

$$(Y_1 + Y_2 + Y_3)\cdot(Y_2 + Y_3 + Y_4 + Y_5 + Y_6)\cdot(Y_3 + Y_4 + Y_5 + Y_6 + Y_7)\cdot(Y_4 + Y_5 + Y_6 + Y_7 + Y_8)\cdot(Y_7 + Y_8 + Y_9) = 1,$$

$$(Y_1 + Y_3 + Y_5 + Y_7)\cdot(Y_3 + Y_5 + Y_7 + Y_9)\cdot(Y_2 + Y_4 + Y_6 + Y_8) = 1$$

und

$$Y_1\cdot Y_2\cdot Y_3\cdot Y_4 + Y_2\cdot Y_3\cdot Y_4\cdot Y_5\cdot Y_6\cdot Y_7 + Y_3\cdot Y_4\cdot Y_5\cdot Y_6\cdot Y_7\cdot Y_8 + Y_7\cdot Y_8\cdot Y_9 = 0;$$

   - Auswählen eines Teil-Subsatzes, bestehend aus 256 aktuellen Codeworten, aus dem vollen Subsatz der möglichen Codeworte;
   - Zuordnen von je einem der 256 möglichen 8-Bit-Datenblöcke zu je einem der aktuellen Codeworte des Teil-Subsatzes; und
   - Codieren der Datenblöcke dadurch, daß jeder Datenblock durch das zugeordnete Codewort ersetzt wird;

   so daß für die sich ergebende Folge von Codeworten die erste vorgewählte Anzahl (G_0) die Vier ist, die zweite vorgewählte Anzahl (I_0) die Sechs ist und die dritte vorgewählte Anzahl (G_1) die Fünf ist.

4. Verfahren nach Anspruch 1 zum Codieren von Daten, von welchen aus den unterteilten 8-Bit-Datenblöcken 9-Bit-Codeworte gebildet werden, umfassend die folgenden Schritte:
   - Auswählen eines vollen Subsatzes (L_1), bestehend aus 259 möglichen Codeworten, aus dem Satz aller möglichen 9-Bit-Worte $Y = (Y_1, Y_2, \ldots, Y_9)$ wobei der Subsatz die folgenden logischen Bedingungen erfüllt:

$$(Y_1 + Y_2 + Y_3)\cdot(Y_2 + Y_3 + Y_4 + Y_5 + Y_6)\cdot(Y_3 + Y_4 + Y_5 + Y_6 + Y_7).(Y_4 + Y_5 + Y_6 + Y_7 + Y_8)\cdot(Y_7 + Y_8 + Y_9) = 1,$$

$$(Y_1 + Y_3 + Y_5)\cdot(Y_5 + Y_7 + Y_9)\cdot(Y_2 + Y_4 + Y_6)\cdot(Y_4 + Y_6 + Y_8) = 1$$

und

$$Y_1\cdot Y_2\cdot Y_3\cdot Y_4\cdot Y_5\cdot Y_6 + Y_5\cdot Y_6\cdot Y_7\cdot Y_8\cdot Y_9 = 0;$$

   - Auswählen eines Teil-Subsatzes, bestehend aus 256 aktuellen Codeworten, aus dem vollen Subsatz der möglichen Codeworte;
   - Zuordnen von je einem der 256 möglichen 8-Bit-Datenblöcke zu je einem der aktuellen Codeworte des Teil-Subsatzes; und
   - Codieren der Datenblöcke dadurch, daß jeder Datenblock durch das zugeordnete Codewort ersetzt wird;

   so daß für die sich ergebende Folge von Codeworten die erste vorgewählte Anzahl (G_0) die Vier ist, die zweite vorgewählte Anzahl (I_0) die Vier ist und die dritte vorgewählte Anzahl (G_1) die Neun ist.

5. Verfahren nach Anspruch 1 zum Codieren von Daten, von welchen aus den unterteilten 8-Bit-Datenblöcken 9-Bit-Codeworte gebildet werden, umfassend die folgenden Schritte:
   - Auswählen eines vollen Subsatzes (L_2), bestehend aus 259 möglichen Codeworten, aus dem Satz

aller möglichen 9-Bit-Worte $Y = (Y_1, Y_2, \ldots, Y_9)$ wobei der Subsatz die folgenden logischen Bedingungen erfüllt:

$$(Y_1 + Y_2 + Y_3)\cdot(Y_2 + Y_3 + Y_4 + Y_5 + Y_6)\cdot(Y_3 + Y_4 + Y_5 + Y_6 + Y_7)\cdot(Y_4 + Y_5 + Y_6 + Y_7 + Y_8)\cdot(Y_7 + Y_8 + Y_9) = 1,$$

$$(Y_1 + Y_3 + Y_5)\cdot(Y_5 + Y_7 + Y_9)\cdot(Y_2 + Y_4 + Y_6)\cdot(Y_4 + Y_6 + Y_8) = 1$$

und

$$Y_1\cdot Y_2\cdot Y_3\cdot Y_4\cdot Y_5 + Y_4\cdot Y_5\cdot Y_6\cdot Y_7\cdot Y_8\cdot Y_9 = 0;$$

- Auswählen eines Teil-Subsatzes, bestehend aus 256 aktuellen Codeworten, aus dem vollen Subsatz der möglichen Codeworte;
- Zuordnen von je einem der 256 möglichen 8-Bit-Datenblöcke zu je einem der aktuellen Codeworte des Teil-Subsatzes; und
- Codieren der Datenblöcke dadurch, daß jeder Datenblock durch das zugeordnete Codewort ersetzt wird;

so daß für die sich ergebende Folge von Codeworten die erste vorgewählte Anzahl (G_0) die Vier ist, die zweite vorgewählte Anzahl (I_0) die Vier ist und die dritte vorgewählte Anzahl (G_1) die Neun ist.

6. Verfahren nach Anspruch 1 zum Codieren von Daten, von welchen aus den unterteilten 8-Bit-Datenblöcken 9-Bit-Codeworte gebildet werden, umfassend die folgenden Schritte:
   - Auswählen aus dem Satz aller möglichen 9-Bit-Worte
   $$Y = (Y_1, Y_2, \ldots, Y_9)$$
   - eines ersten vollen Subsatzes (L), bestehend aus 230 aktuellen Codeworten, welche die folgenden logischen Bedingungen erfüllen:
   $$(Y1 + Y_2 + Y_3)\cdot(Y_2 + Y_3 + Y_4 + Y_5 + Y_6)\cdot(Y_3 + Y_4 + Y_5 + Y_6 + Y_7)\cdot(Y_4 + Y_5 + Y_6 + Y_7 + Y_8)\cdot(Y_7 + Y_8 + Y_9) = 1,$$
   $$(Y_1 + Y_3 + Y_5 + Y_7)\cdot(Y_3 + Y_5 + Y_7 + Y_9)\cdot(Y_2 + Y_4 + Y_6 + Y_8) = 1$$
   und
   $$Y_1\cdot Y_2\cdot Y_3 + Y_2\cdot Y_3\cdot Y_4\cdot Y_5\cdot Y_6 + Y_3\cdot Y_4\cdot Y_5\cdot Y_6\cdot Y_7 + Y_4\cdot Y_5\cdot Y_6\cdot Y_7\cdot Y_8 + Y_7\cdot Y_8\cdot Y_9 = 0;$$
   - Auswählen eines zweiten Subsatzes (1), bestehend aus 26 aktuellen Codeworten, die einer ersten Bedingung mit $Y_1 = Y_2 = Y_3 = 0$ unterliegen und welche weiterhin die folgenden logischen Bedingungen erfüllen:
   $$(Y_4 + Y_5)\cdot(Y_4 + Y_5 + Y_6)\cdot(Y_4 + Y_5 + Y_6 + Y_7)\cdot(Y_4 + Y_5 + Y_6 + Y_7 + Y_8)\cdot(Y_7 + Y_8 + Y_9) = 1,$$
   $$(Y_5 + Y_7)\cdot(Y_5 + Y_7 + Y_9)\cdot(Y_4 + Y_6 + Y_8) = 1$$
   und
   $$Y_4\cdot Y_5\cdot Y_6\cdot Y_7\cdot Y_8 + Y_7\cdot Y_8\cdot Y_9 = 0$$
   - Auswählen eines dritten Subsatzes (0) von 35 möglichen Codeworten, die einer zweiten Bedingung mit $Y_1 = Y_2 = Y_3 = 1$ unterliegen und welche weiterhin die folgenden logischen Bedingungen erfüllen:
   $$(Y_4 + Y_5 + Y_6 + Y_7 + Y_8)\cdot(Y_7 + Y_8 + Y_9) = 1$$
   und
   $$Y_4\cdot Y_5 + Y_4\cdot Y_5\cdot Y_6 + Y_4\cdot Y_5\cdot Y_6\cdot Y_7 + Y_4\cdot Y_5\cdot Y_6\cdot Y_7\cdot Y_8 + Y_7\cdot Y_8\cdot Y_9 = 0;$$
   - Auswählen eines Teil-Subsatzes, bestehend aus 26 aktuellen Codeworten, die der zweiten Bedingung unterliegen, aus dem dritten Subsatz der möglichen Codeworte, die der zweiten Bedingung unterliegen;
   - Zuordnen von je einem der 256 möglichen 8-Bit-Datenblöcke zu je einem der aktuellen Codeworte der Subsätze, wobei 26 von den möglichen 8-Bit-Datenblöcken zweifach zugeordnet werden und zwar einem Codewort, das der ersten Bedingung unterliegt und einem Codewort, das der zweiten Bedingung unterliegt; und
   - Codieren der Datenblöcke dadurch, daß jeder Datenblock durch das zugeordnete Codewort ersetzt wird, wobei die Auswahl der bedingten Codeworte von dem binären Wert des letzten Bits des zuletzt erzeugten Codewortes abhängt;

   so daß für die sich ergebende Folge von Codeworten die erste vorgewählte Anzahl (G_0) die Vier ist, die zweite vorgewählte Anzahl (I_0) die Sechs ist und die dritte vorgewählte Anzahl (G_1) die Vier ist.

7. Vorrichtung zum Codieren binärer Daten in Codeworte, wobei die Vorrichtung umfaßt:
   - Empfängermittel zum Empfangen binärer Daten und zur Aufteilung dieser Daten in Blöcke, von denen jeder eine erste gegebene Anzahl k an Bits umfaßt;
   - Codiermittel, die mit den Empfängermitteln verschaltet sind, zum Erzeugen von Folgen von Codeworten fester Länge, von denen jedes Codewort eine zweite gegebene Anzahl $n = k + 1$ Bits umfaßt;
   - die Folgen enthalten nicht mehr als eine erste vorgewählte Anzahl (G_0) aufeinanderfolgender Nullen;

* C L M S *

- die Folgen umfassen zwei Subfolgen, von denen eine nur aus ungeraden Bitpositionen und die andere nur aus geraden Bitpositionen besteht, wobei jede der Subfolgen nicht mehr als eine zweite vorgewählte Anzahl (I_0) aufeinanderfolgender Nullen enthält,

  dadurch gekennzeichnet, daß

- die Folgen nicht mehr als eine dritte vorgewählte Anzahl (G_1) aufeinanderfolgender Einsen enthalten;
- die dritte vorgewählte Anzahl (G_1) der Minimalwert ist, der in Verbindung mit der gegebenen Kombination der ersten vorgewählten Anzahl (G_0) und der zweiten vorgewählten Anzahl (I_0) auftreten kann.

8. Vorrichtung nach Anspruch 7, dadurch gekennzeichnet, daß sie einen Codierer zur Wandlung von 8-Bit Eingangsdatenblöcken $X = (X_1, X_2, \dots, X_8)$ in 9-Bit-Ausgangscodeworte $Y = (Y_1, Y_2, \dots, Y_9)$ umfaßt, wobei der Codierer ein logisches Gate-Array ist, in welchem die folgenden logischen Funktionen implementiert sind:

$$Y_1 = P_3 \cdot \overline{X}_1 + P_4 \cdot X_8 + P_9 + S_1 + S_6 \cdot X_1$$
$$Y_2 = P_3 \cdot \overline{X}_3 + P_4 \cdot X_5 \cdot \overline{X}_6 + P_8 + S_6 \cdot X_2 + S_9 \cdot X_4$$
$$Y_3 = P_1 \cdot X_3 + P_7 + S_7 + S_8$$
$$Y_4 = P_1 \cdot X_4 + P_8 \cdot X_4 + S_2 + S_8$$
$$Y_5 = P_1$$
$$Y_6 = P_3 + P_4 + P_6 + P_8 \cdot X_6 + S_3 \cdot X_5$$
$$Y_7 = P_2 \cdot X_6 + P_3 \cdot (X_5 + X_6 \cdot \overline{X}_8) + P_4 \cdot \overline{X}_4 + S_3 \cdot X_6 + S_5 \cdot X_5$$
$$Y_8 = P_4 \cdot (\overline{X}_1 \cdot \overline{X}_3 + \overline{X}_2) + P_6 \cdot X_7 \cdot \overline{X}_8 + P_9 \cdot X_4 + S_4 \cdot X_7 + S_7 \cdot \overline{X}_5$$
$$Y_9 = P_2 \cdot \overline{X}_5 + P_3 \cdot X_6 + P_4 \cdot X_1 + P_6 \cdot \overline{X_5 \cdot X_7} + P_9 + S_4 \cdot X_8$$

mit

	$P_1 = V_1 \cdot V_2 \cdot V_3 \cdot V_4 \cdot V_5 \cdot V_6$	$S_1 = P_3 + P_6 = \overline{V}_1$
$V_1 = X_1 + X_2 + X_3$	$P_2 = \overline{P}_1 \cdot X_3 \cdot X_4 \cdot (\overline{X}_1 + \overline{X}_2)$	$S_2 = P_5 + P_7$
$V_2 = \overline{X_1 \cdot X_2 \cdot X_3}$	$P_3 = \overline{P}_1 \cdot \overline{X}_2 \cdot \overline{X}_4 \cdot (X_1 + X_2)$	$S_3 = P_1 + S_2$
$V_3 = X_5 + X_7 + X_8$	$P_4 = \overline{P}_1 \cdot \overline{P}_2 \cdot \overline{P}_3 \cdot V_1 \cdot V_2$	$S_4 = P_2 + P_8 + S_3$
$V_4 = \overline{X_5 \cdot X_6 \cdot X_7 \cdot X_8}$	$P_5 = \overline{V}_1 \cdot V_7$	$S_5 = P_6 + P_8 + P_9$
$V_5 = X_2 + X_4 + X_5 + X_7$	$P_6 = \overline{V}_1 \cdot \overline{V}_7$	$S_6 = P_1 + P_2$
$V_6 = \overline{X_3 \cdot X_4 \cdot X_5 \cdot X_6 \cdot (X_2 + X_7)}$	$P_7 = \overline{V}_2 \cdot V_3 \cdot \overline{X}_5$	$S_7 = P_2 + P_3$
$V_7 = (X_6 + X_8) \cdot V_4$	$P_8 = \overline{V}_2 \cdot V_4 \cdot X_5$	$S_8 = P_4 + P_5$
	$P_9 = \overline{V}_2 \cdot \overline{(P_7 + P_8)}$	$S_9 = P_7 + S_1$

so daß in Reaktion auf eine zufällige Folge der Eingangsdatenblöcke für die sich ergebende Folge der Ausgangscodeworte die erste vorgewählte Anzahl (G_0) die Vier ist, die zweite vorgewählte Anzahl (I_0) die Sechs ist und die dritte vorgewählte Anzahl (G_1) die Fünf ist.

9. Vorrichtung nach Anspruch 7, dadurch gekennzeichnet, daß sie einen Codierer zur Wandlung von 8-Bit Eingangsdatenblöcken $X = (X_1, X_2, \dots, X_8)$ in 9-Bit-Ausgangscodeworte $Y = (Y_1, Y_2, \dots, Y_9)$ umfaßt, wobei der Codierer ein logisches Gate-Array ist, in welchem die folgenden logischen Funktionen implementiert sind:

$$Y_1 = M_1 \cdot X_1 + N_1 \cdot X_1 + R_1 + S_1 \cdot X_4 + T_1 + H$$
$$Y_2 = M_1 \cdot X_2 + R_1 + S_1 + T_1 + H$$
$$Y_3 = M_1 \cdot X_3 + N_1 \cdot X_3 + R_1 + S_1 \cdot \overline{X}_4 + T_1 + H$$
$$Y_4 = M_1 \cdot X_4 + N_1 \cdot \overline{X}_2 + R_1 \cdot \overline{X}_4 + S_1 \cdot (\overline{X}_5 + S_2) + H$$
$$Y_5 = M$$
$$Y_6 = M_2 \cdot X_5 + N_2 \cdot \overline{X}_7 + R_2 \cdot \overline{X}_5 + S_2 \cdot (\overline{X}_4 + S_1)$$
$$Y_7 = M_2 \cdot X_5 + N_2 \cdot X_5 + R_2 + S_2 \cdot \overline{X}_5 + T_2 + H \cdot X_3$$
$$Y_8 = M_2 \cdot X_7 + R_2 + S_2 + T_2$$

$$Y_9 = M_2 \cdot X_8 + N_2 \cdot X_8 + R_2 + S_2 \cdot X_5 + T_2 + H \cdot \lambda_1$$

mit

$$M = (X_2 + V_2) \cdot (X_7 + V_6) \cdot (X_4 + X_3 + X_2 \cdot X_7) \cdot \overline{V}_1 \cdot \overline{V}_3 \cdot \overline{V}_4$$

$$H = \overline{X}_2 \cdot V_3 \cdot V_4$$

$$V_1 = X_1 \cdot X_2 \cdot X_3 \cdot X_4 \cdot X_5$$

$$V_2 = X_5 \cdot X_6 \cdot X_7 \cdot X_8$$

$$V_3 = X_4 \cdot V_2$$

$$V_4 = \overline{X}_4 \cdot V_2$$

$$V_5 = X_1 + X_3$$

$$V_6 = X_3 + X_6$$

$$T_1 = V_1 \cdot V_6 \qquad T_2 = V_2 \cdot V_3$$

$$M_1 = M + V_3 \cdot X_4 \cdot \overline{T}_1 \qquad M_2 = M + V_6 \cdot X_3 \cdot \overline{T}_2 \cdot \overline{H}$$

$$N_1 = \overline{M} \cdot V_3 \cdot \overline{X}_4 \cdot \overline{H} \qquad N_2 = \overline{M} \cdot V_6 \cdot \overline{X}_3$$

$$R_1 = \overline{M} \cdot \overline{V}_3 \cdot X_2 \qquad R_2 = \overline{M} \cdot \overline{V}_6 \cdot X_7$$

$$S_1 = \overline{M} \cdot \overline{V}_3 \cdot \overline{X}_2 \qquad S_2 = \overline{M} \cdot \overline{V}_6 \cdot \overline{X}_7$$

so daß in Reaktion auf eine zufällige Folge der Eingangsdatenblöcke für die sich ergebende Folge der Ausgangscodeworte die erste vorgewählte Anzahl (G_0) die Vier ist, die zweite vorgewählte Anzahl (I_0) die Vier ist und die dritte vorgewählte Anzahl (G_1) die Neun ist.

10. Vorrichtung nach Anspruch 7, dadurch gekennzeichnet, daß sie einen Codierer zur Wandlung von 8-Bit Eingangsdatenblöcken $X = (X_1, X_2, \ldots, X_8)$ in 9-Bit-Ausgangscodeworte $Y = (Y_1, Y_2, \ldots, Y_9)$ umfaßt,

wobei der Codierer ein sequentieller Codierer (Figur 6A) ist, der neun Ausgänge zum Bereitstellen der neun Bits jedes Ausgangscodewortes besitzt und ebenfalls neun Eingänge, von denen acht so verschaltet sind, daß sie die acht Bits eines Eingangsdatenblockes empfangen und der neunte Eingang (Z) ist mit dem neunten Ausgang über ein Verzögerungselement (21) verbunden;
der Codierer ist ein logisches Gate-Array, in welchem die folgenden logischen Funktionen implementiert sind:

$$Y_1 = S_2 \cdot X_1 + P_4 \cdot X_4 + P_5 \cdot (X_1 + \overline{X}_3) + P_6 \cdot X_8 + P_7 \cdot \overline{X}_1 + S_5 \cdot X_5 + P_{10} \cdot X_3 + S_6 \cdot \overline{Z}$$
$$Y_2 = S_2 \cdot X_2 + P_5 \cdot X_2 + P_6 \cdot \overline{X}_6 + P_7 \cdot \overline{X}_3 + S_5 \cdot V_{21} + P_{10} \cdot X_4 + S_6 \cdot \overline{Z}$$
$$Y_3 = S_2 \cdot X_3 + P_4 + P_5 \cdot V_{12} + P_6 \cdot X_6 + P_7 + P_9 \cdot \overline{V}_{22} + P_{10} \cdot \overline{X}_3 + P_{11} \cdot \overline{X}_5 + S_6 \cdot \overline{Z}$$
$$Y_4 = S_2 \cdot \overline{X}_4 + P_4 + P_5 \cdot \overline{X}_4 + P_6 \cdot X_5 + P_{10} + P_{11} \cdot V_{22} + P_{12} \cdot X_4 + P_{13} \cdot Z$$
$$Y_5 = P_1 + P_{12} \cdot Z + P_{13} \cdot \overline{Z}$$
$$Y_6 = S_3 \cdot X_5 + P_3 + P_5 + P_7 \cdot \overline{X}_6 + P_9 + P_{13} \cdot X_4$$
$$Y_7 = S_3 \cdot X_6 + S_8 + P_6 + S_5 + P_{13}$$
$$Y_8 = S_4 \cdot X_7 + P_5 \cdot X_5 + S_5 + S_7 + P_{10} \cdot X_5$$
$$Y_9 = S_4 \cdot X_8 + S_8 \cdot X_5 + P_5 + P_7 \cdot \overline{X}_8 + P_{10} \cdot \overline{X}_6$$

mit

$V_1 = X_1 + X_2$

$V_2 = V_1 + X_3$

$V_3 = X_6 + X_7 + X_9$

$V_4 = \overline{X_1 \cdot X_2 \cdot X_3}$

$V_5 = \overline{X_6 \cdot X_7 \cdot X_8}$

$V_6 = X_5 + X_7$

$V_7 = V_6 + X_2 + X_4$

$V_8 = \overline{X_3 \cdot X_4}$

$V_9 = \overline{X_3 \cdot X_8}$

$V_{10} = \overline{X}_7 + V_8$

$V_{11} = (V_6 + V_9) \cdot (\overline{X}_2 + V_8 + \overline{X}_3) \cdot (V_{10} + \overline{X}_4)$

$V_{12} = V_2 \cdot V_4$

$V_{13} = V_3 \cdot V_5$

$V_{14} = V_{12} \cdot V_{13}$

$V_{15} = V_8 \cdot V_{12}$

$V_{16} = \overline{V}_4 \cdot V_{13}$

$V_{17} = V_2 + X_4$

$V_{18} = \overline{V}_3 \cdot X_4$

$V_{19} = \overline{V}_3 \cdot V_1$

$V_{20} = \overline{X}_2 \cdot (X_3 + \overline{X}_4$

$V_{21} = X_4 + X_3 \cdot \overline{X}_5$

$V_{22} = X_4 + X_3 \cdot X_5$

und

$P_1 = V_7 \cdot V_{11} \cdot V_{14}$

$P_2 = V_{14} \cdot \overline{V}_{11}$

$P_3 = \overline{V}_3 \cdot V_{15}$

$P_4 = V_{16} \cdot V_9$

$P_5 = \overline{V}_5 \cdot \overline{V}_{15}$

$P_6 = \overline{V}_{17} \cdot \overline{V_6 \cdot V_3}$

$P_7 = \overline{V}_7 \cdot V_{14}$

$P_8 = V_{19} \cdot V_4 \cdot \overline{V}_{20}$

$P_9 = V_{18} \cdot V_{20}$

$P_{10} = \overline{V}_3 \cdot \overline{V}_1 \cdot V_{17} + V_{18} \cdot \overline{V}_{19} \cdot \overline{X}_6$

$P_{11} = \overline{V}_4 \cdot \overline{V}_2$

$P_{12} = V_{13} \cdot (\overline{V}_{17} \cdot V_8 + V_{13} \cdot V_{10})$

$P_{13} = V_{16} \cdot \overline{V}_9$

$S_1 = P_1 + P_2$

$S_2 = S_1 + P_3 + P_6$

$S_3 = S_1 + P_4 + P_{12}$

$S_4 = S_3 + P_{13}$

$S_5 = P_5 + P_{11}$

$S_6 = P_{12} + P_{13}$

$S_7 = P_6 + P_7$

$S_8 = P_3 + P_8$

so daß in Reaktion auf eine zufällige Folge der Eingangsdatenblöcke für die sich ergebende Folge der Ausgangscodeworte die erste vorgewählte Anzahl (G_0) die Vier ist, die zweite vorgewählte Anzahl (I_0) die Sechs ist und die dritte vorgewählte Anzahl (G_1) die Vier ist.

11. Decodiervorrichtung zur Wandlung von 9-Bit-Eingangscodeworten $Y = (Y_1, Y_2, \ldots, Y_9)$, die von einer Codiervorrichtung nach Anspruch 8 erzeugt wurden, in 8-Bit Ausgangsdatenblöcke $X = (X_1, X_2, \ldots, X_8)$, wobei die Decodiervorrichtung ein logisches Gate-Array umfaßt, in welchem die folgenden logischen Funktionen implementiert sind:

$$X_1 = Q_3 \cdot \overline{Y}_1 + Q_4 \cdot Y_9 + T_1 \cdot Y_1 + T_2$$
$$X_2 = Q_4 \cdot \overline{Y_8 \cdot Y_9} + T_1 \cdot Y_2 + T_2$$
$$X_3 = Q_1 \cdot Y_3 + Q_2 + Q_3 \cdot \overline{Y}_2 + Q_4 \cdot W_7 + T_2$$
$$X_4 = Q_1 \cdot Y_4 + Q_2 + Q_4 \cdot \overline{Y}_7 + Q_5 \cdot Y_4 + Q_9 \cdot Y_8 + T_6 \cdot Y_2$$
$$X_5 = Q_2 \cdot (Y_7 + \overline{Y}_9) + Q_3 \cdot (W_8 + W_9) + Q_4 \cdot (Y_1 + Y_2) + T_3 \cdot Y_6 + T_5 \cdot Y_7$$
$$X_6 = (Q_2 + T_3) \cdot Y_7 + Q_3 \cdot \overline{Y}_8 + Q_4 \cdot Y_1 + Q_6 \cdot W_7 + Q_8 \cdot Y_6 + Q_9 \cdot Y_7$$
$$X_7 = Q_2 \cdot W_8 + Q_3 \cdot W_9 + Q_4 \cdot Y_1 + Q_6 \cdot (Y_8 + W_7) + Q_9 \cdot Y_7 + T_4 \cdot Y_8$$
$$X_8 = Q_2 \cdot W_9 + (Q_3 + T_4) \cdot Y_9 + Q_4 \cdot Y_1 + Q_6 \cdot W_7 + Q_9 \cdot Y_7$$

mit

$$W_1 = \overline{Y}_4 \cdot \overline{Y}_5 \qquad Q_1 = Y_5 \qquad T_1 = Q_1 + Q_2$$
$$W_2 = Y_4 \cdot \overline{Y}_5 \qquad Q_2 = W_1 \cdot W_5 \qquad T_2 = Q_7 + Q_8 + Q_9$$
$$W_3 = Y_2 \cdot Y_8 \qquad Q_3 = W_1 \cdot W_3 \qquad T_3 = Q_1 + Q_3 + Q_7$$
$$W_4 = Y_1 \cdot \overline{Y}_3 \qquad Q_4 = W_2 \cdot W_3 \qquad T_4 = Q_8 + T_3$$
$$W_5 = Y_2 \cdot \overline{Y}_8 \qquad Q_5 = W_2 \cdot W_4 \qquad T_5 = Q_6 + Q_8 + Q_9$$
$$W_6 = \overline{Y}_1 \cdot \overline{Y}_5 \qquad Q_6 = W_1 \cdot W_4 \qquad T_6 = Q_3 + Q_6 + Q_7$$
$$W_7 = \overline{Y_2 + Y_9} \qquad Q_7 = W_6 \cdot W_5 \cdot Y_4$$
$$W_8 = Y_7 \cdot Y_8 \qquad Q_8 = W_6 \cdot \overline{Y}_3$$
$$W_9 = Y_7 \cdot Y_9 \qquad Q_9 = W_2 \cdot W_5 \cdot Y_1$$

12. Decodiervorrichtung zur Wandlung von 9-Bit-Eingangscodeworten $Y = (Y_1, Y_2, \ldots, Y_9)$ die von einer Codiervorrichtung nach Anspruch 9 erzeugt wurden, in 8-Bit Ausgangsdatenblöcke $X = (X_1, X_2, \ldots, X_8)$, wobei die Decodiervorrichtung ein logisches Gate-Array umfaßt, in welchem die folgenden logischen Funktionen implementiert sind:

$$X_1 = M_1 \cdot Y_1 + N_1 \cdot Y_1 + T_1 + H \cdot Y_9$$
$$X_2 = M_1 \cdot Y_2 + N_1 \cdot \overline{Y}_4 + R_1 + T_1$$
$$X_3 = M_1 \cdot Y_3 + N_1 \cdot Y_3 + T_1 + H \cdot Y_7$$
$$X_4 = M_1 \cdot Y_4 + R_1 \cdot \overline{Y}_4 + T_1 + S_1 \cdot \overline{Y}_3$$
$$X_5 = M_2 \cdot Y_6 + R_2 \cdot \overline{Y}_6 + T_2 + S_2 \cdot \overline{Y}_7 + H$$
$$X_6 = M_2 \cdot Y_7 + N_2 \cdot Y_7 + T_2 + H$$
$$X_7 = M_2 \cdot Y_8 + N_2 \cdot \overline{Y}_6 + R_2 + T_2 + H$$
$$X_8 = M_2 \cdot Y_9 + N_2 \cdot Y_9 + T_2 + H$$

mit

$$W_1 = Y_1 \cdot Y_2 \cdot Y_3 \qquad M_1 = Y_3 + W_5 \qquad M_2 = Y_3 + W_6$$
$$W_2 = W_1 \cdot Y_4 \qquad N_1 = \overline{M}_1 \cdot \overline{Y}_2 \qquad N_2 = \overline{M}_2 \cdot \overline{Y}_6$$
$$W_3 = Y_7 \cdot Y_8 \cdot Y_9 \qquad R_1 = \overline{M}_1 \cdot W_1 \cdot W_7 \qquad R_2 = \overline{M}_2 \cdot W_3 \cdot W_8$$
$$W_4 = W_3 \cdot Y_8 \qquad T_1 = \overline{M}_1 \cdot W_1 \cdot \overline{Y}_4 \cdot \overline{R}_1 \qquad T_2 = \overline{M}_2 \cdot W_3 \cdot \overline{R}_2$$
$$W_5 = Y_4 \cdot Y_2 \cdot \overline{Y}_6 \qquad S_1 = \overline{M}_1 \cdot Y_2 \cdot \overline{Y_1 \cdot Y_3} \qquad S_2 = \overline{M}_2 \cdot Y_8 \cdot \overline{Y_7 \cdot Y_9}$$
$$W_6 = Y_6 \cdot Y_2 \cdot \overline{Y}_4 + W_4 \cdot (\overline{Y}_2 \cdot \overline{Y}_4 + W_2)$$
$$W_7 = Y_4 \cdot Y_6 + W_4 + \overline{Y}_6 \cdot Y_8 \cdot \overline{Y_7 \cdot Y_9} \qquad H = \overline{M}_1 \cdot W_2 \cdot \overline{Y}_8$$
$$W_8 = Y_6 + W_2 + \overline{Y}_4 \cdot Y_2 \cdot \overline{Y_1 \cdot Y_3}$$

13. Decodiervorrichtung zur Wandlung von 9-Bit-Eingangscodeworten $Y = (Y_1, Y_2, \ldots, Y_9)$, die von einer Codiervorrichtung nach Anspruch 10 erzeugt wurden, in 8-Bit Ausgangsdatenblöcke $X = (X_1, X_2, \ldots, X_8)$, wobei die Decodiervorrichtung ein logisches Gate-Array umfaßt, in welchem die folgenden logischen Funktionen implementiert sind:

$$X_1 = T_3 \cdot Y_1 + T_4 + Q_5 \cdot Y_1 \cdot W_{11} + Q_7 \cdot \overline{Y}_1 + Q_9$$
$$X_2 = (T_3 + Q_5) \cdot Y_2 + T_4$$
$$X_3 = T_3 \cdot Y_3 + T_4 + Q_5 \cdot W_{11} + Q_7 \cdot \overline{Y}_2 + Q_9 \cdot \overline{W}_5 + Q_{10} \cdot Y_1$$
$$X_4 = (T_3 + Q_{12}) \cdot Y_4 + Q_4 \cdot Y_1 + Q_5 \cdot \overline{Y}_4 + Q_9 \cdot W_9 + Q_{10} \cdot Y_2 + Q_{11} \cdot Y_2 \cdot Y_4 + Q_{13} \cdot Y_6$$

$$X_5 = T_5 \cdot Y_6 + T_2 \cdot Y_9 + T_7 \cdot Y_8 + Q_6 \cdot Y_4 + T_6 \cdot Y_1 + Q_{13}$$
$$X_6 = T_5 \cdot Y_7 + T_8 + Q_6 \cdot Y_3 + Q_7 \cdot \overline{Y}_6 + Q_{10} \cdot \overline{Y}_9 + Q_{13}$$
$$X_7 = T_9 \cdot Y_8 + T_8 + Q_{10} \cdot \overline{Y}_9$$
$$X_8 = T_9 \cdot Y_9 + T_8 + Q_6 \cdot Y_1 + Q_7 \cdot \overline{Y}_9$$

mit

$$W_1 = \overline{Y_1 \cdot Y_2 \cdot Y_3}$$
$$W_2 = \overline{Y}_3 \cdot W_1$$
$$W_3 = Y_6 \cdot Y_7$$
$$W_4 = Y_2 \cdot Y_4$$
$$W_5 = \overline{Y}_2 \cdot Y_3$$
$$W_6 = Y_4 \cdot W_3$$
$$W_7 = Y_7 \cdot \overline{Y}_6$$
$$W_8 = Y_8 \cdot W_7$$
$$W_9 = Y_2 \cdot \overline{Y}_3$$
$$W_{10} = \overline{Y}_1 \cdot Y_2 \cdot W_4$$
$$W_{11} = Y_2 + Y_3$$
$$W_{12} = W_{11} + Y_1$$
$$W_{13} = Y_6 \cdot Y_2$$
$$W_{14} = Y_4 + Y_6$$
$$W_{15} = Y_3 + Y_7$$
$$W_{16} = W_{14} + Y_8$$
$$W_{17} = W_1 + \overline{Y}_4$$

$$Q_1 = Y_3 \cdot W_1$$
$$Q_2 = W_{10} \cdot Y_8 + W_9 \cdot (Y_8 \cdot W_{12} + W_5 \cdot \overline{Y}_7)$$
$$Q_3 = W_2 \cdot \overline{Y}_6 \cdot W_3 \cdot \overline{W}_4 \cdot W_{12}$$
$$Q_4 = \overline{Y}_6 \cdot \overline{Y}_2 \cdot W_4 \cdot \overline{W}_3$$
$$Q_5 = W_2 \cdot \overline{Y}_7 \cdot \overline{W}_4 \cdot W_{12}$$
$$Q_6 = \overline{Y}_3 \cdot W_3 \cdot (\overline{Y}_4 \cdot \overline{Y_2 \cdot Y_3} + \overline{Y}_9 \cdot W_2)$$
$$Q_7 = W_2 \cdot \overline{Y}_4 \cdot \overline{Y}_7 \cdot \overline{W}_{12}$$
$$Q_8 = W_2 \cdot W_{12} \cdot \overline{Y}_3 \cdot W_7 \cdot \overline{W}_3$$
$$Q_9 = W_2 \cdot \overline{Y}_4 \cdot Y_8 \cdot W_3$$
$$Q_{10} = \overline{Y}_3 \cdot \overline{Y}_6 \cdot \overline{Y}_7 \cdot (\overline{Y}_2 \cdot Y_4 + W_{10})$$
$$Q_{11} = \overline{Y}_3 \cdot W_8 \cdot (\overline{Y}_1 \cdot Y_2 \cdot Y_3 + Y_1 \cdot \overline{Y}_3 \cdot Y_4)$$
$$Q_{12} = \overline{Y}_3 \cdot \overline{W}_1$$
$$Q_{13} = \overline{Y}_3 \cdot \overline{W}_{12} + Y_6 \cdot \overline{W}_1$$

$$T_1 = Q_1 + Q_2$$
$$T_2 = Q_3 + Q_8$$
$$T_3 = T_1 + T_2$$
$$T_4 = Q_4 + Q_{11} + Q_{11}$$
$$T_5 = T_1 + Q_4 + Q_{12}$$
$$T_6 = Q_2 + Q_{11}$$
$$T_7 = Q_3 + Q_{10}$$
$$T_8 = Q_7 + Q_5$$
$$T_9 = T_5 + Q_{13}$$

Revendications

1. Procédé de codage de données binaires partagées en blocs ayant chacun un premier nombre donné k de bits, en mots de codes ayant chacun un second nombre donné de n = k + 1 bits, ledit procédé comprenant les étapes suivantes:
   - réception des données binaires;
   - production de séquences de mots de codes de longueur fixe,
     - lesdites séquences n'ayant pas plus d'un premier nombre présélectionné (G0) de zéros consécutifs à l'intérieur de celles-ci;
     - lesdites séquences comprenant deux séquences, l'une consistant uniquement en positions de bits impaires, et l'autre consistant uniquement en positions de bits paires, chacune desdites séquences n'ayant pas plus d'un second nombre présélectionné (I0) de zéros consécutifs à l'intérieur de celle-ci;

   caractérisé en ce que

* C L M S *

- des mots de codes sont sélectionnés de telle sorte que le nombre maximum de ceux qui sont consécutifs qui peut se produire dans toute séquence de mots de codes n'excède pas un troisième nombre présélectionné (G1) qui est un minimum possible en connexion avec la combinaison donnée du premier nombre présélectionné (G0) et du second nombre présélectionné (I0).

2. Procédé selon la revendication 1, pour coder des données à partir desquelles par partage des blocs de données à 8 bits ont été formés, en mots de codes à 9 bits, comprenant les étapes suivantes:
sélection à partir de la série de tous les mots possibles à 9 bits Y = (Y1, Y2, ..., Y9) d'une sous-série complète (L1) de 263 mots de codes potentiels satisfaisant aux conditions logiques

$$(Y1 + Y2 + Y3).(Y2 + Y3 + Y4 + Y5 + Y6).(Y3 + Y4 + Y5 + Y6 + Y7).(Y4 + Y5 + Y6 + Y7 + Y8).(Y7 + Y8 + Y9) = 1$$

$$(Y1 + Y3 + Y5 + Y7).(Y3 + Y5 + Y7 + Y9).(Y2 + Y4 + Y6 + Y8) = 1$$

et

$$Y1.Y2.Y3 + Y2.Y3.Y4.Y5.Y6.Y7 + Y3.Y4.Y5.Y6.Y7.Y8 + Y6.Y7.Y8.Y9 = 0;$$

- sélection à partir de la sous-série complète de mots de codes potentiels, d'une sous-série partielle de 256 mots de codes réels;
- assignation à chacun des mots de codes réels de la sous-série partielle, de l'un des 256 blocs de données à 8 bits possibles; et
- pour le codage, remplacement de chaque bloc de données par le mot de code assigné; de telle sorte que pour la séquence résultante de mots de codes, ledit premier nombre présélectionné (G0) est de quatre, ledit second nombre présélectionné (I0) est de six, et ledit troisième nombre présélectionné (G1) est de cinq.

3. Procédé selon la revendication 1, pour coder des données à partir desquelles par partage des blocs de données à 8 bits ont été formés, en mots de codes à 9 bits, comprenant les étapes suivantes:
- sélection à partir de la série de tous les mots à 9 bits possibles Y=(Y1, Y2, ..., Y9) d'une sous-série complète (L2) de 263 mots de codes potentiels satisfaisant aux conditions logiques

$$(Y1 + Y2 + Y3).(Y2 + Y3 + Y4 + Y5 + Y6).(Y3 + Y4 + Y5 + Y6 + Y7).(Y4 + Y5 + Y6 + Y7 + Y8).(Y7 + Y8 + Y9) = 1$$

$$(Y1 + Y3 + Y5 + Y7).(Y3 + Y5 + Y7 + Y9).(Y2 + Y4 + Y6 + Y8) = 1,$$

et

$$Y1.Y2.Y3.Y4 + Y2.Y3.Y4.Y5.Y6.Y7 + Y3.Y4.Y5.Y6.Y7.Y8 + Y7.Y8.Y9 = 0;$$

- sélection à partir de la sous-série de mots de codes potentiels, d'une sous-série partielle de 256 mots de codes réels;
- assignation à chacun des mots de codes réels de la sous-série partielle, de l'un des 256 blocs de données à 8 bits possibles; et
- pour le codage, remplacement de chaque bloc de données par le mot de code assigné; de telle sorte que pour la séquence résultante de mots de codes, ledit premier nombre présélectionné (G0) est de quatre, ledit second nombre présélectionné (I0) est de six, et ledit troisième nombre présélectionné (G1) est de cinq.

4. Procédé selon la revendication 1, pour coder des données à partir desquelles par partage des blocs de données à 8 bits ont été formés, en mots de codes à 9 bits, comprenant les étapes suivantes:
- sélection à partir de la série de tous les mots à 9 bits possibles Y=(Y1, Y2, ..., Y9) d'une sous-série complète (L1) de 259 mots de codes potentiels satisfaisant aux conditions logiques

$$(Y1 + Y2 + Y3).(Y2 + Y3 + Y4 + Y5 + Y6).(Y3 + Y4 + Y5 + Y6 + Y7).(Y4 + Y5 + Y6 + Y7 + Y8).(Y7 + Y8 + Y9) = 1$$

$$(Y1 + Y3 + Y5).(Y5 + Y7 + Y9).(Y2 + Y4 + Y6) = 1,$$

et

$$Y1.Y2.Y3.Y4.Y5.Y6 + Y5.Y6.Y7.Y8 + Y7.Y8.Y9 = 0;$$

- sélection à partir de la sous-série de mots de codes potentiels, d'une sous-série partielle de 256 mots de codes réels;
- assignation à chacun des mots de codes réels de la sous-série partielle, de l'un des 256 blocs de données à 8 bits possibles; et
- pour le codage, remplacement de chaque bloc de données par le mot de code assigné; de telle sorte que pour la séquence résultante de mots de codes, ledit premier nombre présélectionné (G0) est de quatre, ledit second nombre présélectionné (I0) est de quatre, et ledit troisième nombre présélectionné (G1) est de neuf.

* C L M S *

5. Procédé selon la revendication 1, pour coder des données à partir desquelles par partage des blocs de données à 8 bits ont été formés, en mots de codes à 9 bits, comprenant les étapes suivantes:
   - sélection à partir de la série de tous les mots à 9 bits possibles Y=(Y1, Y2, ..., Y9) d'une sous-série complète (L2) de 259 mots de codes potentiels satisfaisant aux conditions logiques

$$(Y1 + Y2 + Y3).(Y2 + Y3 + Y4 + Y5 + Y6).(Y3 + Y4 + Y5 + Y6 + Y7).(Y4 + Y5 + Y6 + Y7 + Y8).(Y7 + Y8 + Y9) = 1$$

$$(Y1 + Y3 + Y5).(Y5 + Y7 + Y9).(Y2 + Y4 + Y6).(Y4 + Y6 + Y8) = 1,$$

   et

$$Y1.Y2.Y3.Y4.Y5 + .Y4.Y5.Y6.Y7.Y8 + Y7.Y8.Y9 = 0;$$

   - sélection à partir de la sous-série de mots de codes potentiels, d'une sous-série partielle de 256 mots de codes réels;
   - assignation à chacun des mots de codes réels de la sous-série partielle, de l'un des 256 blocs de données à 8 bits possibles; et
   - pour le codage, remplacement de chaque bloc de données par le mot de code assigné; de telle sorte que pour la séquence résultante de mots de codes, ledit premier nombre présélectionné (G0) est de quatre, ledit second nombre présélectionné (I0) est de quatre, et ledit troisième nombre présélectionné (G1) est de neuf.

6. Procédé selon la revendication 1, pour coder des données à partir desquelles par partage des blocs de données à 8 bits ont été formés, en mots de codes à 9 bits, comprenant les étapes suivantes:
   - sélection à partir de la série de tous les mots à 9 bits possibles Y=(Y1, Y2, ..., Y9) d'une sous-série complète (L) de 230 mots de codes potentiels satisfaisant aux conditions logiques

$$(Y1 + Y2 + Y3).(Y2 + Y3 + Y4 + Y5 + Y6).(Y3 + Y4 + Y5 + Y6 + Y7).(Y4 + Y5 + Y6 + Y7 + Y8).(Y7 + Y8 + Y9) = 1$$

$$(Y1 + Y3 + Y5 + Y7).(Y3 + Y5 + Y7 + Y9).(Y2 + Y4 + Y6 + Y8) = 1,$$

   et

$$Y1.Y2.Y3 + Y2.Y3.Y4.Y5.Y6 + Y3.Y4.Y5.Y6.Y7 + Y4.Y5.Y6.Y7.Y8 + Y7.Y8.Y9 = 0;$$

   - une seconde sous-série (1) de 26 mots de codes conditionnels avec Y1=Y2=Y3=0, satisfaisant les conditions logiques

$$(Y4 + Y5).(Y4 + Y5 + Y6).(Y4 + Y5 + Y6 + Y7).(Y4 + Y5 + Y6 + Y7 + Y8).(Y7 + Y8 + Y9) = 1,$$

$$(Y5 + Y7).(Y5 + Y7 + Y9).(Y4 + Y6 + Y8) = 1,$$

   et

$$Y4.Y5.Y6.Y7.Y8 + Y7.Y8.Y9 = 0;$$

   - une troisième sous-série (0) de 35 seconds mots de codes conditionnels potentiels avec Y1=Y2=Y3=1, satisfaisant les conditions logiques

$$(Y4 + Y5 + Y6 + Y7 + Y8).(Y7 + Y8 + Y9) = 1,$$

   et

$$Y4.Y5 + Y4.Y5.Y6 + Y4.Y5.Y6.Y7 + Y4.Y5.Y6.Y7.Y8 + Y7.Y8.Y9 = 0;$$

   - sélection à partir de la troisième sous-série de secondsmots de codes conditionnels potentiels d'une sous-série partielle de 26 mots de codes conditionnels réels;
   - assignation à chacun des mots de codes réels des sous-séries, de l'un des 256 blocs de données à 8 bits possibles, avec 26 desdits blocs de données à 8 bits possibles étant assignés chacun à un premier mot de code conditionnel et à un second mot de code conditionnel; et
   - pour le codage, remplacement de chaque bloc de données par le mot de code assigné, la sélection des mots de codes conditionnels dépendant de la valeur binaire du dernier bit du mot de code généré au préalable;
     de telle sorte que pour la séquence résultante de mots de codes, ledit premier nombre présélectionné (G0) est de quatre, ledit second nombre présélectionné (I0) est de six, et ledit troisième nombre présélectionné (G1) est de quatre.

7. Appareil pour coder des données binaires en mots de codes, ledit appareil comprenant:
   - des moyens récepteurs pour recevoir les données binaires et pour les partager en blocs comprenant chacun un premier nombre donné de k bits;
   - des moyens codeurs, couplés aux moyens récepteurs, pour produire des séquences de mots de codes de longueur fixe comprenant chacune un second nombre donné de n = k+1 bits;
   - lesdites séquences n'ayant pas plus d'un premier nombre présélectionné (G0) de zéros consécutifs dans celles-ci;
   - lesdites séquences comprenant deux séquences, l'une consistant uniquement en positions de bits

impaires, et l'autre consistant uniquement en positions de bits paires, chacune desdites séquences n'ayant pas plus d'un second nombre présélectionné (I0) de zéros consécutifs à l'intérieur de celles-ci; caractérisé en ce que

- lesdites séquences n'ayant en outre pas plus d'un troisième nombre présélectionné (G1) de uns consécutifs à l'intérieur de celles-ci;
- ledit troisième nombre présélectionné (G1) ayant la valeur minimum possible en connexion avec la combinaison donnée du premier nombre présélectionné (G0) et le second nombre présélectionné (I0).

8. Appareil selon la revendication 7, comprenant un codeur pour convertir des blocs de données d'entrée à 8 bits X = (X1, X2, ..., X8) en mots de codes de sortie à 9 bits Y = (Y1, Y2, ..., Y9), ledit codeur étant un réseau de portes logiques mettant en oeuvre les relations logiques

$$Y1 = P3.X1 + P4X3 + P8 + S1 + S6.X1$$
$$Y2 = P3.X3 + P4.X5.X6 + P8 + S6.X2 + S3.X4$$
$$Y3 = P1.X3 + P7 + S7 + S8$$
$$Y4 = P1.X4 + P3.X4 + S2 + S8$$
$$Y5 = P\&$$
$$Y6 = P3 + P4 + P6 + P8.X6 + S3.X5$$
$$Y7 = P2.X6 + P3.(X5 + X6.X8) + P4.X4 + S3.X6 + S5.X5$$
$$Y8 = P4.(X1.X3 + X2) + P6.X7.X8 + P3.X4 + S4.X7 + S7.X5$$
$$Y9 = P2.5 + P3.X3 + P4.X1 + P5.\overline{X5X7} + P3 + S4.X8$$

avec

$V1=X1+X2+X3$	$P1=V1.V2.V3.V4.V5.V6$	$S1=P5+P6=V1$
$V2=\overline{X1.X2.X3}$	$P2=P1.X3.X4.(X1+X2)$	$S2=P5+P7$
$V3=X5+X7+X8$	$P3=P1.X2.X4.(X1+X3)$	$S3=P1+S2$
$V4=\overline{X5.X6.X7.X8}$	$P4=P1.P2.P3.V1.V2$	$S4=P2+P3+S3$
$V5=X2+X4+X5+X7$	$P5=V1.V7$	$S5=P6+P2+P3$
$V6=\overline{X3.X4.X5.X6.(X2+X7)}$	$P6=V1.V7$	$S6=P1+P2$
$V7=(X6+X8).V4$	$P7=V2.V3.X5$	$S7=P2+P3$
	$P8=V2.(\overline{P7+P8})$	$S8=P7+S1$

de telle sorte qu'en réponse à une séquence arbitraire de blocs d'entrée de données, pour la séquence résultante de mots de codes de sortie ledit premier nombre présélectionné (G0) est de quatre, ledit second nombre présélectionné (I0) est de six, et ledit troisième nombre présélectionné (G1) est de cinq.

9. Appareil selon la revendication 7, comprenant un codeur pour convertir des blocs de données d'entrée à 8 bits X = (X1, X2, ..., X8) en mots de codes de sortie à 9 bits Y = (Y1, Y2, ..., Y9), ledit codeur étant un réseau de portes logiques mettant en oeuvre les relations logiques

$$Y1 = M1.X1 + N1.X1 + R1 + S1 + X4 + T1 + H$$
$$Y2 = M1 + X2 + R1 + S1 + T1 + H$$
$$Y3 = M1.X3 + N1.X3 + R1 + S1.\overline{X4} + T1 + H$$
$$Y4 = M1.X4 + N1.\overline{X2} + R1.X4 + S1.(\overline{X5} + S2) + H$$
$$Y5 = M$$
$$Y6 = M2.X5 + N2.\overline{X7} + R2.X5 + S2.(\overline{X4} + S1)$$
$$Y7 = M2.X5 + N2.X5 + R2 + S2.\overline{X5} + T2 + H.X3$$
$$Y8 = M2.X7 + R2 + S2 + T2$$
$$Y9 = M2.X8 + N2.X8 + R2 + S2.X5 + T2 + H.X1$$

avec

$$V1=X1.X2.X3.X4.X5$$

$$M=(X2+V5).(X7+V6).(X4+X5+X2.X7).\overline{V1}.\overline{V3}.\overline{V4}$$

$$V2=X5.X6.X7.X8 \qquad H=\overline{X2}.V5.V4$$

$$V3=X4.V2 \qquad T1=V1.V6 \qquad T2=V3.V5$$

$$V4=\overline{X4}.V2 \qquad M1=\overline{M}+V5.X4.\overline{T1} \qquad M2=M+V6.X5.\overline{T2}.\overline{H}$$

$$V5=X1+X3 \qquad N1=\overline{M}.\overline{V5}.X4.\overline{H} \qquad N2=\overline{M}.V6.\overline{X5}$$

$$V6=X5+X6 \qquad R1=\overline{M}.\overline{V5}.X2 \qquad R2=\overline{M}.\overline{V6}.X7$$

$$S1=\overline{M}.\overline{V5}.X2 \qquad S2=\overline{M}.\overline{V5}.\overline{X7}$$

de telle sorte qu'en réponse à une séquence arbitraire de blocs d'entrée de données, pour la séquence résultante de mots de codes de sortie ledit premier nombre présélectionné (G0) est de quatre, ledit second nombre présélectionné (I0) est de quatre, et ledit troisième nombre présélectionné (G1) est de neuf

10. Appareil selon la revendication 7, comprenant un codeur pour convertir une séquence de blocs de données d'entrée à 8 bits X = (X1, X2, ..., X8) en mots de codes de sortie à 9 bits Y = (Y1, Y2, ..., Y9);

ledit codeur étant un codeur séquentiel (figure 6A) ayant neuf sorties pour fournir les neuf bits de chaque mot de code de sortie; et ayant neuf entrées, huit d'entre elles étant connectées pour recevoir les huit bits d'un bloc de données d'entrée, et la neuvième entrée (Z) étant connectée à la neuvième sortie par l'intermédiaire d'un élément à retard (21);

ledit codeur comprenant un réseau de portes logiques mettant en oeuvre les relations logiques

$$Y1 = S2.X1 + P4.X4 + P5.(X1 + \overline{X3}) + P6.X6 + P7.\overline{X1} + S5.X5 + P10.X3 + S6.\overline{Z}$$

$$Y2 = S2.X2 + P5.X2 + P6.\overline{X6} + P7.\overline{X3} + S5.V21 + P10.X4 + S6.\overline{Z}$$

$$Y3 = S2.X3 + P4 + P5.V12 + P6.X6 + P7 + P9.\overline{V22} + P10.\overline{X3} + P11.\overline{X5} + S6.\overline{Z}$$

$$Y4 = S2.X4 + P4 + P5.\overline{X4} + P6.X5 + P10 + P11.V22 + P12.X4 + P13.Z$$

$$Y5 = P1 + P12.Z + P13.\overline{Z}$$

$$Y6 = S3.X5 + P3 + P5 + P7.\overline{X6} + P3 + P13.X4$$

$$Y7 = S3.X6 + S8 + P6 + S5 + P13$$

$$Y8 = S4.X7 + P5.X5 + S5 + S7 + P10.X5$$

$$Y9 = S4.X8 + S8.X5 + P5 + P7.X8 + P10.\overline{X6}$$

avec

$V1=X1+X2$ $\quad V9=\overline{X5.X6}$ $\quad V17=V2+X4$

$V2=V1+X3$ $\quad V10=\overline{X7}+V9$ $\quad V18=\overline{V2}.X4$

$V3=X6+X7+X8$ $\quad V11=(V8+V9).(\overline{X2}+V8+\overline{X5}).(V10+\overline{X4})$

$V19=\overline{V3}.V1$

$V4=\overline{X1.X2.X3}$ $\quad V12=V2.V4$ $\quad V20=\overline{X2}.(X3+\overline{X4})$

$V5=\overline{X6.X7.X8}$ $\quad V13=V3.V5$ $\quad V21=X4+X3.\overline{X5}$

$V6=X5+X7$ $\quad V14=V12.V13$ $\quad V22=X4+X3.X5$

$V7=V6+X2+X4$ $\quad V15=V8.V12$

$V8=\overline{X3.X4}$ $\quad V16=\overline{V4}.V13$

et

$P1=V7.V11.V14$ $\quad P8=V19.V4.\overline{V20}$ $\quad S1=P1+P2$

$P2=V14.\overline{V11}$ $\quad P9=V19.V20$ $\quad S2=S1+P3+P8$

$P3=\overline{V5}.V15$ $\quad P10=\overline{V3}.\overline{V1}.V17+V18.\overline{V10}.\overline{X8}$

$S3=S1+P4+P12$

$P4=V15.V3$ $\quad P11=\overline{V4}.\overline{V3}$ $\quad S4=S3+P13$

$P5=\overline{V5}.\overline{V15}$ $\quad P12=V13.(\overline{V17}.V6+V18.V10)$

$S5=P9+P11$

$P6=\overline{V17}.\overline{V6.V3}$ $\quad P13=V16.\overline{V9}$ $\quad S6=P12+P13$

$P7=\overline{V7}.V14$ $\quad S7=P6+P7$

$S8=P3+P8$

de telle sorte qu'en réponse une séquence arbitraire de blocs de données d'entrée, pour la séquence résultante de mots de code de sortie ledit premier nombre présélectionné (G0) est de quatre, ledit second nombre présélectionné (I0) est de six, et ledit troisième nombre présélectionné (G1) est de quatre.

11. Appareil décodeur pour convertir des mots de code d'entrée à 9 bits Y = (Y1, Y2, ..., Y9) générés par un appareil codeur selon la revendication 8, en blocs de données de sortie à 8 bits X = (X1, X2, ..., X8), ledit appareil décodeur comprenant un réseau de portes logiques mettant en oeuvre les relations logiques

$$X1 = Q3.\overline{Y1} + Q4Y9 + T1.Y1 + T2$$
$$X2 = Q4.\overline{Y8.Y9} + T1.Y2 + T2$$
$$X3 = Q1.Y3 + Q2 + Q3.\overline{Y2} + Q4.W7 + T2$$
$$X4 = Q1.Y4 + Q2 + Q4 + \overline{Y7} + Q8.Y4 + Q9.Y8 + T6.Y2$$
$$X5 = Q2.(Y7 + \overline{Y9}) + Q3.(W8 + W9) + Q4 + (Y1 + Y2) + T3.Y6 + T5.Y7$$
$$X6 = (Q2 + T3).Y7 + Q3.\overline{Y8} + Q4.Y1 + Q6.W7 + Q8.Y6 + Q9.Y7$$
$$X7 = Q2.W8 + Q3.W9 + Q4.Y1.Q6.(Y8 + W7) + Q9.Y7 + T4.Y8$$
$$X8 = Q2.W3 + (Q3 + T4).Y9 + Q4.Y1 + Q6.W7 + Q9.Y7$$

avec

$W1=\overline{Y4}.\overline{Y5}$ $Q1=Y5$ $T1=Q1+Q2$

$W2=Y4.\overline{Y5}$ $Q2=W1.W5$ $T2=Q7+Q8+Q9$

$W3=Y3.Y6$ $Q3=W1.W3$ $T3=Q1+Q5+Q7$

$W4=Y1.\overline{Y3}$ $Q4=W2.W3$ $T4=Q8+T3$

$W5=Y1.\overline{Y5}$ $Q5=W2.W4$ $T5=Q6+Q8+Q9$

$W6=\overline{Y1}.\overline{Y5}$ $Q6=W1.W4$ $T6=Q5+Q6+Q7$

$W7=\overline{Y8+Y9}$ $Q7=W6.W5.Y4$

$W8=Y7.Y8$ $Q8=W5.\overline{Y3}$

$W9=Y7.Y9$ $Q9=W2+W5+Y1$

12. Appareil décodeur pour convertir des mots de codes d'entrée à 9 bits Y = (Y1, Y2, ..., Y3) générés par un appareil codeur selon la revendication 9, en blocs de données de sortie à 8 bits X = (X1, X2, ..., X8), ledit appareil décodeur comprenant un réseau de portes logiques mettant en oeuvre les relations logiques

$$X1 = M1.Y1 + N1.Y1 + T1 + H.Y9$$
$$X2 = M1.Y2 + N1.\overline{Y4} + R1 + T1$$
$$X3 = M1.Y3 + N1.Y3 + T1 + H.Y7$$
$$X4 = M1.Y4 + R1.\overline{Y4} + T1 + S1.\overline{Y3}$$
$$X5 = M2.Y6 + R2.\overline{Y6} + T2 + S2.\overline{Y7} + H$$
$$X6 = M2.Y7 + N2.Y7 + T2 + H$$
$$X7 = M2.Y8 + N2.\overline{Y6} + R2 + T2 + H$$
$$X8 = M2.Y9 + N2.Y9 + T2 + H$$

avec

$W1=Y1.Y2.Y3$ $M1=Y5+W5$ $M2=Y5+W6$

$W2=W1.Y4$ $N1=\overline{M1}.\overline{Y2}$ $N2=\overline{M2}.Y8$

$W3=Y7.Y8.Y9$ $R1=\overline{M1}.W1.W7$ $R2=\overline{M2}.W3.W8$

$W4=W3.Y6$ $T1=\overline{M1}.W1.W7$ $T2=\overline{M2}.W3.R2$

$W5=Y4.Y8.\overline{Y9}$ $S1=\overline{M1}.Y2.\overline{Y1.Y3}$ $S2=\overline{M2}.Y8.\overline{Y7.Y9}$

$W6=Y6.Y2.\overline{Y4}+W4.(\overline{Y2}.\overline{Y4}+W2)$ $H=\overline{M1}.W2.\overline{Y8}$

$W7=Y4.Y6+W4+\overline{Y6}.Y8.\overline{Y7.Y9}$

$W8=Y6+W2+\overline{Y4}.Y2.\overline{Y1.Y3}$

13. Appareil décodeur pour convertir des mots de codes d'entrée à 9 bits Y = (Y1, Y2, ..., Y9) générés par un appareil codeur selon la revendication 10, en blocs de données de sortie à 8 bits X = (X1, X2, ..., X8), ledit appareil décodeur comprenant un réseau de portes logiques mettant en oeuvre les relations logiques

$$X1 = T3.Y1 + T4 + Q5.Y1.W11 + Q7.\overline{Y1} + Q9$$
$$X2 = (T3 + Q5).Y2 + T4$$
$$X3 = T3.Y3 + T4 + Q5.W11 + Q7.\overline{Y2} + Q9.\overline{W5} + Q10.Y1$$
$$X4 = (T3 + Q12).Y4 + Q4.Y1 + Q5.\overline{Y4} + Q9.W9 + Q10.Y2 + Q11.Y2.Y4 + Q13.Y6$$
$$X5 = T5.Y6 + T2.Y9 + T7.Y8 + Q6.Y4 + T6.Y1 + Q13$$
$$X6 = T5.Y7 + T8 + Q6.Y3 + Q7.\overline{Y6} + Q10.\overline{Y9} + Q13$$
$$X7 = T9.Y8 + T8 + Q10.\overline{Y9}$$
$$X8 = T9.Y9 + T8 + Q6.Y1 + Q7.\overline{Y9}$$

avec

$W1=\overline{Y1.Y2.Y3}$ $Q1=Y5.W1$ $T1=Q1+Q2$

$W2=\overline{Y5}.W1$ $Q2=W10.Y6+W6.(Y8.W12+W5.\overline{Y8})$

$T2=Q3+Q8$

$W3=Y6.Y7$ $Q3=W2.\overline{Y8}.W3.\overline{W4}.W12$

$T3=T1+T2$

$W4=Y3.Y4$ $Q4=\overline{Y5}.\overline{Y2}.W4.\overline{W3}$

$T4=Q4+Q11+Q13$

$W5=\overline{Y2}.Y3$ $Q5=W2.\overline{Y7}.\overline{W4}.W13$

$T5=T1+Q4+Q12$

$W6=Y4.W3$ $Q6=Y5.W8.(\overline{Y4}.\overline{Y2.Y3}+\overline{Y1}.W9)$

$T6=Q9+Q11$

$W7=Y7.\overline{Y6}$ $Q7=W2.\overline{Y4}\overline{Y7}.\overline{W13}$

$T7=Q5+Q10$

$W8=Y8.W7$ $Q8=W2.W12.\overline{Y8}.W7.\overline{W5}$

$T8=Q3+Q5$

$W9=Y2.\overline{Y3}$ $Q9=W2.\overline{Y4}.Y8.W3$

$T9=T5+Q13$

$W10=\overline{Y1}.Y2.W4$ $Q10=\overline{Y5}.\overline{Y6}.\overline{Y7}.(\overline{Y3}.Y4+W10)$

$W11=Y2+Y3$ $Q11=\overline{Y5}.W8.(\overline{Y1}.Y2.Y3+Y1.\overline{Y3}.Y4)$

$W12=W11+Y1$ $Q12=\overline{Y5}.\overline{W1}$

$W13=Y6.Y9$ $Q13=\overline{Y5}.\overline{W12}+Y5.\overline{W1}$

$W14=Y4+Y6$

$W15=Y3+Y7$

$W16=W14+Y8$

$W17=W1+\overline{Y4}$