OPIC
Office de la propriété intellectuelle du Canada

CIPO
Canadian Intellectual Property Office

(12) (19) (CA) **Brevet-Patent**

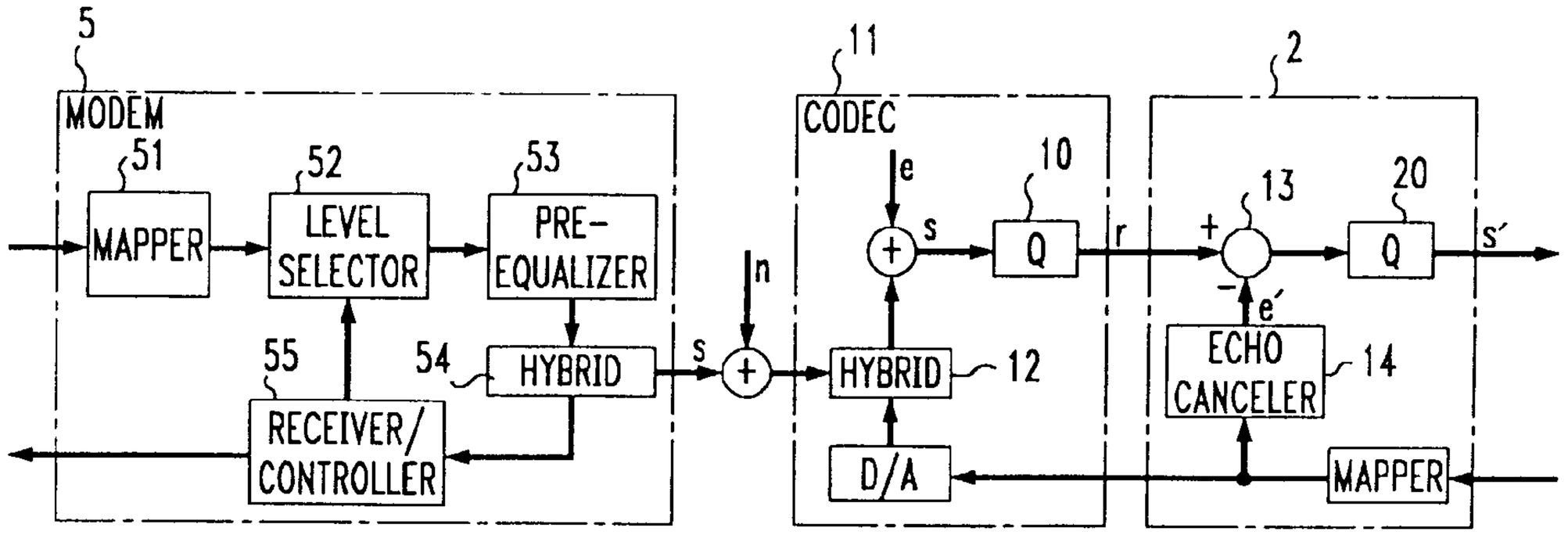
(11) (21) (C) **2,167,745**
(22) 1996/01/22
(43) 1996/08/02
(45) 2000/10/31

(72) Herzberg, Hanan, US
(72) Saltzberg, Burton Reuben, US
(73) AT&T CORP., US

(51) Int.Cl.[6] H04L 27/00, H04L 1/00
(30) 1995/02/01 (384,498) US

(54) **DISPOSITIF DE COMMUNICATION A SUPPRESSION AMELIOREE DES ECHOS ET DES BRUITS DANS UN CANAL PORTANT DES DONNEES QUANTIFIEES**

(54) **COMMUNICATION ARRANGEMENT WITH IMPROVED ECHO AND NOISE SUPPRESSION IN A CHANNEL CONTAINING QUANTIZATION**



(57) A modem that matches itself to a central office codec to which it is connected and incorporates encoding of the data. An echo canceller following the central office codec subtracts an estimate of the echo introduced in the central offices. More particularly, digital data applied to the modem is multi-level encoded and mapped onto a subset of the levels recognized by the central office codec. In an illustrative embodiment, the multi-level encoding encodes the two least significant bits in the binary representation of the levels delivered by the modem. The least significant bit is encoded with a code that is more robust than the code of the next-to-least significant bit, in recognition of the fact that errors in the least significant bit are much more likely. The corresponding echo canceller in the network first estimates the echo and subtracts it from the signal applied to the digital network by the central office. Thereafter, a decoder decodes the received signal in stages. In the first stage, the least significant bit of symbols is identified based on a first error correcting code. In the second stage the remaining bits of the symbols are determined, with the aid of a second error correcting code and the results of the first stage. Block or convolutional encoding can be used.

A Communication Arrangement With Improved Echo and Noise Suppression in a Channel Containing Quantization

Abstract

A modem that matches itself to a central office codec to which it is connected and incorporates encoding of the data. An echo canceller following the central office codec subtracts an estimate of the echo introduced in the central offices. More particularly, digital data applied to the modem is multi-level encoded and mapped onto a subset of the levels recognized by the central office codec. In an illustrative embodiment, the multi-level encoding encodes the two least significant bits in the binary representation of the levels delivered by the modem. The least significant bit is encoded with a code that is more robust than the code of the next-to-least significant bit, in recognition of the fact that errors in the least significant bit are much more likely. The corresponding echo canceller in the network first estimates the echo and subtracts it from the signal applied to the digital network by the central office. Thereafter, a decoder decodes the received signal in stages. In the first stage, the least significant bit of symbols is identified based on a first error correcting code. In the second stage the remaining bits of the symbols are determined, with the aid of a second error correcting code and the results of the first stage. Block or convolutional encoding can be used.

A Communication Arrangement With Improved Echo and Noise Suppression in a Channel Containing Quantization

Background Of the Invention

1. Field of the invention

This invention relates to codecs.

2. Description of Prior Art

Modems communicate digital data over an analog communication medium by mapping the data onto analog signals. Some of today's sophisticated modems communicate digital information by mapping the data onto analog signals and, thereafter, modulating an analog carrier with those signals. Typically, a collection of bits are combined to form a two-dimensional symbol, the symbol components are converted to analog form, and the analog-form components respectively modulate a carrier and a quadrature replica of the carrier. The two modulation products are added and filtered, and the result is applied to the transmission medium. A remote modem receives the signal, samples it, identifies the magnitudes and phases of the analog samples, converts the samples to symbols, and finally recovers the original bits of data. What these modems do, in effect, is encode the digital signals onto a two dimensional symbol constellation that is modulated onto a carrier.

When the telecommunication network is completely analog the primary sources of error are signal echoes from imperfect hybrids and other discontinuities in the transmission lines, as well as noise from a myriad of unknown sources. When the telecommunication network includes digital links, where the analog data is sampled and quantized, an additional noise source is introduced that is rooted in the sampling and quantization that occurs at the network.

One might believe that the additional noise caused by the quantization in the network would lower the maximum data rate that may be achieved, but a United States Patent 5,394,437 entitled, "High Speed Modem Synchronized to a Remote CODEC" issued February 28, 1995, discloses a modem that achieves actually higher data rates than those that can be achieved by conventional modems. The higher rate is realized by synchronizing the sampling in the new modem to the sampling which takes place in the μ-law codec that is in the network, and by arranging for the signal levels of the modem to coincide with the quantization levels of the μ-law codec. Basically, the modem of the '437 patent insures that no error signal is created by the sampling process in the network.

While the improved modem of the '437 patent solves the quantization problem created by the digital network's codec, a difficulty still arises from echoes and

noise that are unavoidably introduced into the signal just prior to the network's quantization.

Summary

The difficulty that arises from the noise and echo that interact with codecs in central offices is overcome with a new modem and a corresponding echo canceller embedded in the digital network. The new modem matches itself to a central office codec to which it is connected and incorporates encoding of the data. The echo canceller subtracts as estimate of the echo introduced in the central offices.

In accordance with one aspect of the present invention there is provided communication apparatus comprising: an encoder responsive to an applied digital signal for developing digitized symbols, where said digitized symbols comport with a preselected error correcting code; a level selector for developing analog levels in response to said digitized symbols; and means responsive to said analog levels for communicating signals to a remote quantizer, where levels of the signals communicated to the remote quantizer are set to correspond to quantization levels of the remote quantizer.

In accordance with another aspect of the present invention there is provided decoding apparatus for developing a stream of output symbols comprising: means for receiving a stream of received symbol levels that is related to a stream of transmitted symbols; means for subtracting interference estimate levels from said stream of received symbol levels to develop substractor output symbol levels, and a decoder responsive to said means for substracting, having a first stage that identifies the most likely sequence of transmitted symbols corresponding to said stream of received symbols based on a first error correcting code associated with a first set of bits of a label related to the subtractor output symbol levels, and a second stage responsive to the first stage and employing a second error correcting code associated with second set of bits of a label related to the subtractor output symbol levels, for developing said stream of output symbols that corresponds to the most likely sequence of transmitted symbols.

In a disclosed illustrative embodiment, digital data applied to the modem is encoded and mapped onto a subset of the levels recognized by the central office codec, with the two least significant bits in the binary representation of the levels delivered by the modem having been encoded with a multi-level code. The least significant bit is

B

encoded with a code that is more robust than the code of the next-to-least significant bit, in recognition of the fact that errors in the least significant bit are much more likely. The corresponding echo canceller in the network first estimates the echo and subtracts it from the signal applied to the digital network by the central office.

Thereafter, a decoder decodes the received signal in stages. In the first stage, the least significant bit of symbols is identified based on a first error correcting code. In the second stage the remaining bits of the symbols are determined, with the aid of a second error correcting code and the results of the first stage. Block or convolutional encoding can be used.

Brief Description of the Drawings

FIG. 1 shows the prior art arrangement;

FIG. 2 presents a possible approach for solving the echo problem which does not fully overcome echo-induced errors;

FIGS. 3-5 illustrate the possible echo-induced errors that are possible with the arrangement of FIG. 2;

FIGS. 6-8 depict a number of encoding arrangements;

FIG. 9 illustrates an arrangement that eliminates echo-induced errors;

FIG. 10 illustrates one convolutional encoding arrangement for encoder 31 of FIG. 8;

FIG. 11 depicts still another encoding arrangement;

FIGS. 12A-12B present a flow chart of decoding a signal encoded by encoder 31;

FIG. 13 is a block diagram of a decoder that combines the decoding associated with encoders 31 and 32 and develops an estimate of the sent signal; and

FIG. 14 presents a flow chart of the decoding performed in block 62 of FIG. 13.

Detailed Description

FIG. 1 depicts an arrangement where a modem 5 of the type disclosed in the aforementioned '437 patent is connected to a central office codec 11. Local equipment which is coupled to modem 5 generates digital data, that data is mapped into symbols in mapper 51, the symbols are converted to levels of quantizer 10 in synchronism with the central office's μ-law codec 11 in level selector 52, and equalized in pre-equalizer 53. The output of equalizer 53 is applied to a two-wire facility by means of hybrid 54 which sends the signal to the central office. The matching of level selector 52 to quantizer 10 (both in level and in sampling times) is achieved via feedback from the central office derived through hybrid 54 and receiver/controller 55. Of course, it is possible to adjust the levels *a priori* in the factory, or dynamically. The dynamic adjustment may be continuous or take place only at the beginning of transmission. The sampling time is adjusted continually. The central office receives the signals through hybrid 12 and forwards them to quantizer 10, where the received levels are quantized and converted to digital form to form digital stream, r. Though not shown in FIG. 1, it is expected that the digital stream developed by quantizer 10 is sent to a remote central office and, therefrom, to a digital subscriber device (perhaps without even going through a conversion to analog levels via a complementary modem).

Experience tells us that the signal applied to quantizer 10 within the central office is adulterated by an additive noise, n, and an additive echo signal, e, which originates primarily from the behavior of hybrid 12 (though for sake of simplicity, FIG. 1 shows the echo being added by means of an adder). Thus, the input to quantizer 10 within the central office is s+e+n, and the output of the central office's codec is r, which is the quantized level of s+e+n. Even though the signal s is adjusted by modem 5 to coincide with the output levels of quantizer 10, the output r may be different from s because of the additive signals.

The level of the noise is unknown, but the echo can be reasonably estimated. It makes sense, therefore, for the central office to subtract the echo estimate from the signal before it is transmitted. Alas, the central office equipment is already in place, and it is not practical to modify it. An alternative that must be fashioned, therefore, is to attempt to remove the echo component from the digital signal that flows in the network which *follows* the central office. Such equipment is shown in FIG. 2 in the form of block 2, where subtractor 13 subtracts the echo estimate, e', from the values of r, and the result is applied to quantizer, 20, which develops the digital signal, s'. The echo

A8

estimate is derived in accordance with well known techniques by means of echo canceller 14 which is responsive to the signal that is coupled to hybrid 12 through a D/A converter. For echo canceling, see Gitlin et al, "Data Communications Principles", Plenum Press, 1992.

One might expect that the signal s' would be essentially equal to signal s. That is, the only remaining error should be the additive noise signal, n, and a signal that corresponds to the difference between the echo, e, and its estimate, e'. Unfortunately, the FIG. 2 arrangement fails to remove all effects of the echo. FIGS. 3-5 demonstrate the problem.

In FIG. 3, the signal s is sent at level 11, the echo brings the signal up to level 12, which is close to threshold 23 between levels 13 and 14. When the noise happens to have a small negative value, quantizer 10 within the central office chooses level 13, the signal e' is subtracted to yield level 15, and quantizer 20 develops signal s', which is at level 11 of signal s. In other words, no error is introduced. In FIG. 4, on the other hand, the small noise level happens to be positive and it moves the signal at the input of quantizer 10 to above threshold 23. The result is that quantizer 10 outputs level 14, the echo estimate is subtracted to reach level 16; but now quantizer 20 yields level 17 rather than level 11. That, of course, is a one-level error.

In FIG. 5 the levels are not uniform (which indeed is the case in the μ-law codec), and that leads to the possibility of a two-level error. Starting at level 11, the echo signal moves the input signal of quantizer 10 to level 18. The noise pushes the signal to above the threshold between level 19 and 21, so quantizer 10 develops level 21. Subtracting the estimate signal e' yields level 22, and quantizer 20 develops the signal of level 24. That is two levels above level 11. It can be shown that the probability of an error of the type shown in FIG. 4 is substantially greater than an error of the type shown in FIG. 5.

As indicated above, the quantized levels of signal s correspond a set of symbols that are created by selector 52. In terms that are familiar to artisans who deal with modems, the symbols of s and r can be viewed as a constellation of symbols along the x axis. The most prevalent error is an error that shifts a symbol to one of its neighbors in the constellation. The next most likely error is one that skips over a symbol in the constellation. If the set of quantized levels, s, is denoted in order (i.e., the first symbol is 00000, the next is 00001, the following one is 00010, etc.,) then the most prevalent error is an error in the least significant bit of the symbol. The next most prevalent is in the next to the least significant bit of the symbol. It follows that whatever error

correcting codes are incorporated into signal s, at least these two types of error should be correctable.

Aside from the fact that the echo and noise can combine to cause errors even when the echo is removed and that the μ-law codec action of quantizer 10 can compound the problem, it is recognized that the lower levels of the μ-law codec are too close together to give a good performance level, and that the very high levels of the μ-law codec work fine but "cost" too much power. It appears advantageous, therefore, to not use all 255 levels of the network's μ-law codec, which means that the bit rate that can be sent through the network's codec is less than 8 bits per symbol. On the other hand, the number of useful levels may be greater than 128 (corresponding to 7 bits per symbol), which means that a fractional number of bits per symbol can be employed. The number of useful levels is dependent on the level of noise immunity that is desired. As is well known in the art, fractional bit rate transmission results when a number of symbols A is developed from a number of input bits B, and the ratio of B/A is a mixed fraction. In the context of this disclosure, the mapping of bits into symbols also insures (through careful design of the mapping) that those symbols which correspond to codec levels which we do not wish to use do not appear at the output of the mapper.

Fractional bit rate mapping can map the entire set of incoming bits, but it doesn't have to. For a fractional bit rate, m, the number of symbols that the codec must support is at least 2^m, rounded up to the next integer. It can be shown that a slightly less stringent arrangement can be obtained with only a slightly heavier burden on the codec by mapping less than all of the bits. For example, where m is 7.5 bits per symbol and the codec must support 182 symbols, a mapping can be employed where two bits are not mapped at all (that is, they are kept as is) and the remaining 5.5 bits per symbol are mapped, requiring the codec to support 184 symbols. Thus, mapping 5.5 bits per symbol requires the codec to support two more symbols but concomitantly requires less hardware than mapping 7.5 bits per symbol. An example of such a mapping approach is disclosed in U.S. Patent 4,941,154, issued July 10, 1990, in connection with trellis encoding of two dimensional constellations.

Combining the need to protect the two least significant bits of the symbols applied to codec 11 with the desire to use a fractional bit rate to maximize utilization of codec 11, we realized that fractional bit rate mapping of the type where not all of the bits are mapped can be combined to an advantage with coding of the bits that do not participate in the fractional bit rate mapping to achieve superior performance. FIG. 6 shows one example of such an arrangement where every one of the bits that is not

mapped is applied to a coder, FIG. 7 shows an example where only some of the bits that do not participate in the fractional bit rate mapping are encoded (and the other ones are not), and FIG. 8 presents an example where the encoding is multi-level. The FIG. 8 arrangement is particularly well suited to the problem at hand, because the encoding that is desired for the least significant bit of the symbols going to quantizer 10 aims to more robustly protect the bit next to the least significant bit. Of course, a more robust encoding requires more redundancy and is more expensive in terms of transmission capacity.

FIG. 9 presents an arrangement according to the principles disclosed herein where it may be noted that elements 30, 31, 32, and 33 replace mapper 51 of FIG. 1. Equalizer 53 can follow the principles taught in U.S. Patent No. 5,528,625 filed for Ayangolu et al, entitled "High Speed Quantization-Level-Sampling Modem with Equalization Arrangement", issued on June 18, 1996.

As indicated above, it is advantageous to not use the full 255 levels of quantizer 10 and, therefore, a rate that is lower than 8 bits per symbol is the result. We have concluded that a rate of 6.73333 is good for certain applications, and that rate corresponds to a transmission of 202 information bits with every 30 symbols. That is with every 202 information bits that flow through buffer 30, there are 30 symbols that are created by blocks 30-33. With every two symbol periods, buffer 30 delivers 11 bits to fractional bit rate mapper 33, in response to which mapper 33 delivers 6 bits with every symbol period. Effectively, mapper 33 creates a bit every two symbols. It may be noted that the output of this mapping is capable of supporting an alphabet of 2^{12}, but since the input is restricted to 2^{11} combinations, there are some members of the alphabet that are never applied to level selector 52. The 6 bits of mapper 33 form the most significant bits of the symbols applied to the level selector. Getting back to buffer 30, with every 30 symbol periods buffer 30 delivers 27 bits to encoder 32. Encoder 32, correspondingly, delivers a single bit to level selector 52 with every symbol period. Effectively, encoder 32 creates a bit every 10 symbol periods. The output of encoder 32 forms the next to least significant bit of the symbols applied to the level selector. Lastly, with every 30 symbol periods, buffer 30 delivers 10 bits to encoder 31, and encoder 31 delivers a single bit to level selector 52 with every symbol period. Effectively, encoder 31 creates 2 bits every 3 symbol periods. The output of encoder 31 forms the least significant bit of the symbols applied to the level selector. Specifically which bits of the 202 bits are employed by encoders 31, 32 or 33 is not important, so

any convenient routing algorithm will do, as long as that algorithm is known, so that the received symbols can be decomposed eventually into the proper sequence of bits.

The design approach specifics of elements 31, 32 and 33 are well known in the art, in the sense that artisans know how to create these elements given the particular input/output function that is desired. For example, mapper 33 may be a ROM with the 11 bit input being merely an address that is applied to the ROM. The ROM has 12 outputs which are divided into two groups of 6 bits each, and the groups are alternately directed to the output of mapper 33 with each symbol period. Mapper 32, for example, may be a simple parity code generator which for every 9 bits adds a parity bit. Mappers 31 and/or 32 may be block encoders or convolutional encoders. An illustrative convolutional encoder for mapper 31 is shown in FIG. 10. The coder of FIG. 10 includes 4 delay elements, modulo 2 adders 35, 36, and 37 and commutating switch 38. With every set of three symbol intervals, a bit is inserted into the first delay element 34 of the FIG. 10 encoder (with data shifting through the succeeding delay elements). Switch 38 accesses a different adder with each symbol interval and commutates among the three adders.

It may be noted that different encoders may be used, and that a multi-level encoder arrangement that is modeled after FIG. 7 is also possible. Such an arrangement is depicted in FIG. 11, where the multi-level encoder is the same as in FIG. 8, and the only change is in the number of bits that are produced by Fractional Bit Rate (FBR) mapper 39. It may be noted that a ROM implementation of mapper 33 requires 12×2^{11} bits, whereas a ROM implementation of mapper 39 requires 10×2^{9} bits; a reduction by a factor of 4.8.

The fact that modem 5 of FIG. 9 includes a means for incorporating error protection for the two least significant bits of the symbols applied to level selector 52 and, thence, to quantizer 10 is not the end of the journey, of course. Means must be provided in element 200 to make use of the error correcting codes and to thereby correct the errors that might otherwise appear at the output. That task is assigned to decoder 60. The conventional approach for correcting such errors is through "soft decoding", where the actual level that is obtained after subtracting the echo estimate, e', is evaluated vis-à-vis the two possible quantized values that the signal might assume. Describing the process very generally, a "cost/benefit" metric is assigned to the two choices and the "metric" is accumulated over the entire sequence of the 30 symbols. The encoding action of elements 31, 32, and 33 results in a finite set of impermissible sequences in

signal s; or sequences with a high associated costs. Based on the accumulated "metric" value, the right sequence can be identified and, hence, errors can be corrected.

FIG. 12 presents a flow diagram of the decoding algorithm performed in block 200 of FIG. 9, as it relates to the least significant bit. It describes the decoding of a block code where the block length is N. For a convolutional code, a Viterbi decoder can be used (see the aforementioned Gitlin reference), with the same "cost/benefit" metric as described below. The flow chart of FIG. 12 incorporates the action of elements 13 and 20, as well as the decoding of block 60. In other words, the FIG. 12 flow chart corresponds to an embodiment of block 200 via a stored program controlled processor that carries out the processing of FIG. 12.

Starting at element 101 of the flowchart, an index k, which designates a member in a block of signals, is set to 1. Control passes to element 102, where signal t (the output of subtractor 13 in FIG. 9) is evaluated. Element 103 in FIG. 12 quantizes signal t in accordance with the reduced alphabet of the μ-law encoder to obtain an intermediate signal, $z_A^{(k)}$, which corresponds to the output of block 20 in FIG. 9. In element 104, the signal $z_A^{(k)}$ is recognized to be associated with a label of the signal $x^{(k)}$, which is the signal applied to block 60.

The quantization performed by the Z operator of element 103 chooses the nearest of the two neighbors on either side of $t^{(k)}$. The two neighbors are identified here by $z_A^{(k)}$, and $z_B^{(k)}$, where the neighbor that is closest to $t^{(k)}$ is chosen by the quantization process to be labeled $z_A^{(k)}$. Decision element 105 determines whether the quantization of signal $t^{(k)}$ chose a lower level neighbor or a higher level neighbor. When $z_A^{(k)}$ is higher than $t^{(k)}$, decision element 106 determines whether the corresponding label, $x_A^{(k)}$, is equal to M-1, where M is the number of labels in the block 60 modem (M-1 is the last label in the set of labels and corresponds to the biggest binary number). When that is true, element 107 sets the other neighbor, $x_B^{(k)}$, to M-2, and control passes to element 108. When that is false, element 109 sets $x_B^{(k)}$ to $x_A^{(k)}$+1 and, again, control passes to element 108. When $t^{(k)}$ is not greater than $z_A^{(k)}$ and when $x_A^{(k)}$ equal to zero (i.e., the lowest label), element 111 $x_B^{(k)}$ to 1 and control passes to element 108. Lastly, when $x_A^{(k)}$ is not equal to zero, element 112 sets $x_B^{(k)}$ to $x_A^{(k)}$-1 and, as before, control passes to element 108.

Having evaluated the second neighbor, $x_B^{(k)}$, element 108 determines the $z_B^{(k)}$ level that corresponds to $x_B^{(k)}$. Element 114 evaluates distances from thresholds, which relate to the "cost/benefit" metric, and control passes to element 115.

The "metric" over the entire block associated with selecting $x_A^{(k)}$ or $x_B^{(k)}$ is evaluated in the loop that follows element 115. Element 115 sets index i to zero and element 116 compares the value of i to the least significant bit of $x_A^{(k)}$, $x_{A0}^{(k)}$. When they are equal, the metric is determined in block 117. When the variance of the noise, ^{2}, is known, the "cost/benefit" metric is $\phi_i^{(k)} = \phi(\delta_1^{(k)}, \delta_2^{(k)})$, where $\phi(\delta_1, \delta_2) = \log \frac{Q(-\delta_1/\sigma)}{Q(-\delta_2/\sigma)}$.

In the equation, $\delta_1^{(k)} = \frac{\Delta^{(k)}}{2} - |x_A^{(k)} - t^{(k)}|$, $\delta_2^{(k)} = \frac{\Delta^{(k)}}{2} - |x_B^{(k)} - t^{(k)}|$, and

$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-y^2/2} dy$, where Δ is the distance between the thresholds that surround $r^{(k)}$. When the least significant bit of $x_A^{(k)}$ is not equal to i, then according to element 118, $\phi_i^{(k)} = 0$. Once the metric is determined for i=0, with the help of decision element 119 and action element 110 which increments i, a "cost/benefit" metric is determined for i=1. Control then passes to element 121, with $\phi_1^{(k)}$ and $\phi_0^{(k)}$determined and the index k incremented. As an aside, if the noise variance is not known, a sub-optimal decoding can take place, with the "cost/benefit" metric defined by $\phi(\delta_1^{(k)}, \delta_2^{(k)})$ being equal to δ_2^2 when $\delta_1^2 \geq 0$ and $\delta_2^2 < 0$; being equal to 0 when $\delta_1 \geq 0$ and $\delta_2 \geq 0$; and being equal to $\delta_2^2 - \delta_1^2$ when $\delta_1 < 0$ and $\delta_2 \leq 0$.

Returning to FIG. 12, element 121 increments index k and passes control to decision element 122. When k is not greater than the size of the block, N, element 122 passes control back to element 102. When k is greater than N, control passes to element 123 where index i is set to zero and the process of determining the best suited codeword commences. Element 124 evaluates the accumulated metrics for the ith codeword, E_i, element 125 increments index i, and element 126 determines whether a metric of a next codeword should be evaluated. When the index i is not greater than the number of codewords in E, control returns to element 124 (where E is the error correcting code used for the least significant bit). Otherwise, element 127 identifies the index i for which the maximum benefit is obtained, and the value of this index is memorialized by assigning it to index f. Control then passes to element 128, where index, k, is set to one.

Following the setting of index k in element 128, decision element 113 compares the least significant bit of the kth component in the code corresponding to index f, $E_f^{(k)}$ to $x_{A0}^{(k)}$ and, if they are equal, then the final estimate of the sent label, $\hat{x}^{(k)}$, is set equal to $x_A^{(k)}$, and the final estimate of the sent symbol, $\hat{s}^{(k)}$, is set to $z_A^{(k)}$. This is accomplished in element 130. When decision element 113 obtains a false answer, then pursuant to element 131, the final estimate of the sent label, $\hat{x}^{(k)}$, is set equal to $x_B^{(k)}$, and the final estimate of the sent symbol, $\hat{s}^{(k)}$, is set to $z_B^{(k)}$. Thereafter, element 131 increments k by

one and, if k is not greater than N, then according to decision element 132, control returns to decision element 113. Otherwise, element 133 sets k equal to one and control returns to element 102 to start decoding another block.

The result of the FIG. 12 process is a selection of a symbol which satisfies the criteria imposed by encoder 31 and which probably is the symbol that was sent. More specifically, it is a symbol that is part of a valid codeword and it has the correct least significant bit. What remains to be determined is whether the other bits are also the correct bits, and in particular, whether the next-to-least-significant bit is correct.

The determination of the correct symbol follows the FIG. 13 structure. In FIG.13, decoder 61 corresponds to the flow chart of FIG. 12 and it determines the least significant bit of the symbol while it determines a tentative symbol, $\hat{s}$. That symbol is applied to decoder 62 which identifies a valid level that comports with the requirements of encoder 32 and, by doing so, identifies the next-to-least-significant bit of the final symbol, s', as well as the remaining bits. Since the next-to-least-significant bit of the symbol was derived through simple parity check encoding, the process of correcting that bit is somewhat simpler than the process shown in FIG. 12. That process is presented in FIG. 14. The output of decoders 61 and 62 corresponds to the least significant bits that were applied to the error correction encoding of elements 31 and 32, and to the bits at the output of mapper 33. To recover the bits at the input of mapper 33, a simple table look-up translation may be employed, and that is accomplished via translator 63.

Addressing the decoding process of FIG. 14, element 201 sets index k to 1, and element 202 determines the level $t^{(k)}$, which is $r^{(k)} - e^{(k)}$. When $t^{(k)}$ is greater than $\hat{s}^{(k)}$ (the selection made by decoder 61), elements 206, 207 and 209 look at $\hat{x}^{(k)}$, which is the label associated with $\hat{s}^{(k)}$, and select a neighboring label, $x_2^{(k)}$. Control then passes to element 208. Similarly, when $t^{(k)}$ is not greater than $\hat{s}^{(k)}$, elements 210, 211 and 213 also select label, $x_2^{(k)}$ and pass control to element 208. Element 208 associates symbol $s_2^{(k)}$ with level $x_2^{(k)}$. Element 214 evaluates a "cost/benefit" metric for the selection of that second symbol, and it corresponds to $\gamma^{(k)} = |s_2^{(k)} + e^{(k)} - r^{(k)}| - \frac{\Delta^{(k)}}{2}$, where Δ is the distance between the thresholds that surround $r^{(k)}$. Thereafter, element 215 determines whether k is less than N-1, where N is the length of the block (the length of blocks in encoding the least significant bit and the next-to-least significant bit do not need to be the same), and in such a circumstance, control returns to element 201 to determine the metric associated with another member of the block. Otherwise, control passes to element 216 where a determination is made regarding the parity of the signal; i.e., the

parity of the second bit in the label of the symbol. When it is determined that parity is even (which is the parity imposed by encoder 32), it is concluded that the initial estimation made by decoder 61 is valid, and the final estimate, $s'^{(k)}$, for all values of k is set equal to the entered estimates $\hat{s}^{(k)}$. Control then passes to element 201 to start a new record. Otherwise, element 217 selects an index η that corresponds to the maximum γ, and modifies that label by setting $s'^{(\eta)}$ to $s_2^{(\eta)}$ (leaving the remaining $s'^{(k)}$ set to $\hat{s}^{(k)}$). Lastly, control again returns to element 201 to begin processing a new record.

The above discloses a number of principles, and some examples embodying the disclosed principles but, of course, various enhancements are possible that are within the spirit and scope of the disclosed invention. For example, while the above speaks of echo as the source of error that is compensated for, it should be apparent that other sources of error that can be estimated within element 200 can also be handled.

Claims:

1. Communication apparatus comprising:

an encoder responsive to an applied digital signal for developing digitized symbols, where said digitized symbols comport with a preselected error correcting code;

a level selector for developing analog levels in response to said digitized symbols; and

means responsive to said analog levels for communicating signals to a remote quantizer, where levels of the signals communicated to the remote quantizer are set to correspond to quantization levels of the remote quantizer.

2. The apparatus of claim **1** further comprising means, responsive to signals derived from said quantizer for adjusting signals levels developed by said level selector.

3. The apparatus of claim **1** wherein said error correcting code is a block code.

4. The apparatus of claim **3** where a sequence of symbols is developed from a contiguous block of bits of said applied digital signal.

5. The apparatus of claim **1** wherein said error correcting code is a convolutional code.

6. The apparatus of claim **1** wherein said encoder comprises a mapper which, in response to every A bits of said applied signal that are applied to the mapper develops B sets of bits, where each set of bits contains C bits which comprise the most significant bits of the symbols, where A, B, and C are integers.

7. The apparatus of claim **6** where the mapper is a fractional bit rate mapper.

8. The apparatus of claim **6** where A/B is a mixed fraction.

9. The apparatus of claim **6** further comprising coding means responsive to bits of said applied signal for developing bits which comprise the least significant bits of said symbols and which contain redundancy adapted to enable error correction.

10. The apparatus of claim **6** further comprising:

first coding means responsive to bits of said applied signal for developing the least significant bits of said symbols; and

second coding means responsive to bits of said applied signal for developing the next to least significant bits of said symbols.

11. The apparatus of claim **6** further comprising:

first coding means responsive to bits of said applied signal for developing the least significant bits of said symbols, where said least significant bits of said symbols comport with a first preselected error correcting code; and

second coding means responsive to bits of said applied signal for developing the next to least significant bit of said symbols, where said next to least significant bits of said symbols comport with a second preselected error correcting code.

12. The apparatus of claim **11** where said first preselected error correcting code is different from said second preselected error correcting code.

13. The apparatus of claim **11** where said first preselected error correcting code is more capable at correcting errors than said second preselected error correcting code.

14. The apparatus of claim **6** further comprising:

first coding means responsive to bits of said applied signal for developing groups of bits comprising the least significant bits of said symbols, where said least significant bits of said symbols comport with a first preselected error correcting code; and

second coding means responsive to bits of said applied signal for developing middle groups of bits comprising the bits of said symbols that are between said least significant bits of said symbols and said most significant bits of said symbols, where said middle groups of bits of said symbols comport with a second preselected error correcting code.

15. The apparatus of claim **6** further comprising:

first coding means responsive to bits of said applied signal for developing groups D of bits, where said groups D comport with a first preselected error correcting code; and

second coding means responsive to bits of said applied signal for developing groups E of bits, where said middle groups of bits of said symbols comport with a second preselected error correcting code;

where each of said symbols is a concatenation of the C bits from the mapper and the bits from groups E, and D, in order, where the C bits of the mapper comprise the most significant bits of each symbol.

16. The apparatus of claim **6** further comprising:

first coding means responsive to bits of said applied signal for developing groups D of bits, where said groups D comport with a first preselected error correcting code; and

second coding means responsive to bits of said applied signal for developing groups E of bits, where said middle groups of bits of said symbols comport with a second preselected error correcting code;

means for developing groups F from preselected bits of said applied signal; and

means for forming symbols from said groups D, E, and F, and the C bits from the mapper, where each of said symbols is a concatenation of the C bits from the mapper and the bits from groups F, E, and D, in order, where the C bits of the mapper comprise the most significant bits of each symbol.

17. The apparatus of claim **1** where said encoder comprises a fractional bit rate mapper and coding means and where, according to a preselected routing schema, some bits of said applied digital signal are applied to said fractional bit rate mapper to develop sets A of bits, and some bits of said applied digital signal are applied to said coding means to develop sets B of bits, where sets A of bits and sets B of bits combine to drive the level selector.

18. The apparatus of claim **17** where said coding means performs multi-level coding.

19. The apparatus of claim **17** where

said coding means performs multi-level coding to create sets B of bits,

each set B comprises two subsets, and

one of the subsets comports with a first preselected error correction code, and the other of the subsets is unencoded.

20. The apparatus of claim **17** where

said coding means performs multi-level coding to create sets B of bits,

each set B comprises two subsets, and

one of the subsets comports with a first preselected error correction code, and the other of the subsets comports with a second preselected error correction code.

21. The apparatus of claim **20** where said first preselected error correction code is a block code, and said second preselected error correction code is a parity code.

22. The apparatus of claim **20** where said first preselected error correction code is a convolution code, and said second preselected error correction code is a parity code.

23. The apparatus of claim **20** where said first preselected error correction code is a convolution code, and said second preselected error correction code is a block code.

24. The apparatus of claim **17** where

said coding means performs multi-level coding to create sets B of bits,

each set B comprises three subsets, and

one of the subsets comports with a first preselected error correction code, another of the subsets comports with a second preselected error correction code, and a third of the subsets is uncoded.

25. Decoding apparatus for developing a stream of output symbols comprising:

means for receiving a stream of received symbol levels that is related to a stream of transmitted symbols;

means for subtracting interference estimate levels from said stream of received symbol levels to develop subtractor output symbol levels, and

a decoder responsive to said means for subtracting, having

a first stage that identifies the most likely sequence of transmitted symbols corresponding to said stream of received symbols based on a first error correcting code associated with a first set of bits of a label related to the subtractor output symbol levels, and

a second stage responsive to the first stage and employing a second error correcting code associated with second set of bits of a label related to the subtractor output symbol levels, for developing said stream of output symbols that corresponds to the most likely sequence of transmitted symbols.

26 . The decoding apparatus of claim **25** where the first stage identifies the most likely sequence of transmitted symbols with the aid of metrics that are related to the distance between said subtractor output symbol levels and preselected thresholds.

27. The decoding apparatus of claim **25** where the first error correcting code is a block code.

28. The decoding apparatus of claim **25** where the first error correcting code is a convolutional code.

29. The decoding apparatus of claim **25** where the first error correcting code is more robust than the second error correcting code.

30. The decoding apparatus of claim **25** where the second error correcting code is a parity code.

31. The apparatus of claim **25** where the first set of decoded bits is the least significant bit.

32. The apparatus of claim **25** where the first set of decoded bits is the least significant bit and the second set of bits the next to least significant bit.

33. The apparatus of claim **25** where the first stage determines the first set of bits of each symbol in said stream of output symbols, and the second stages determines remaining bits of each symbol in said stream of output symbols.

FIG. 1
(PRIOR ART)

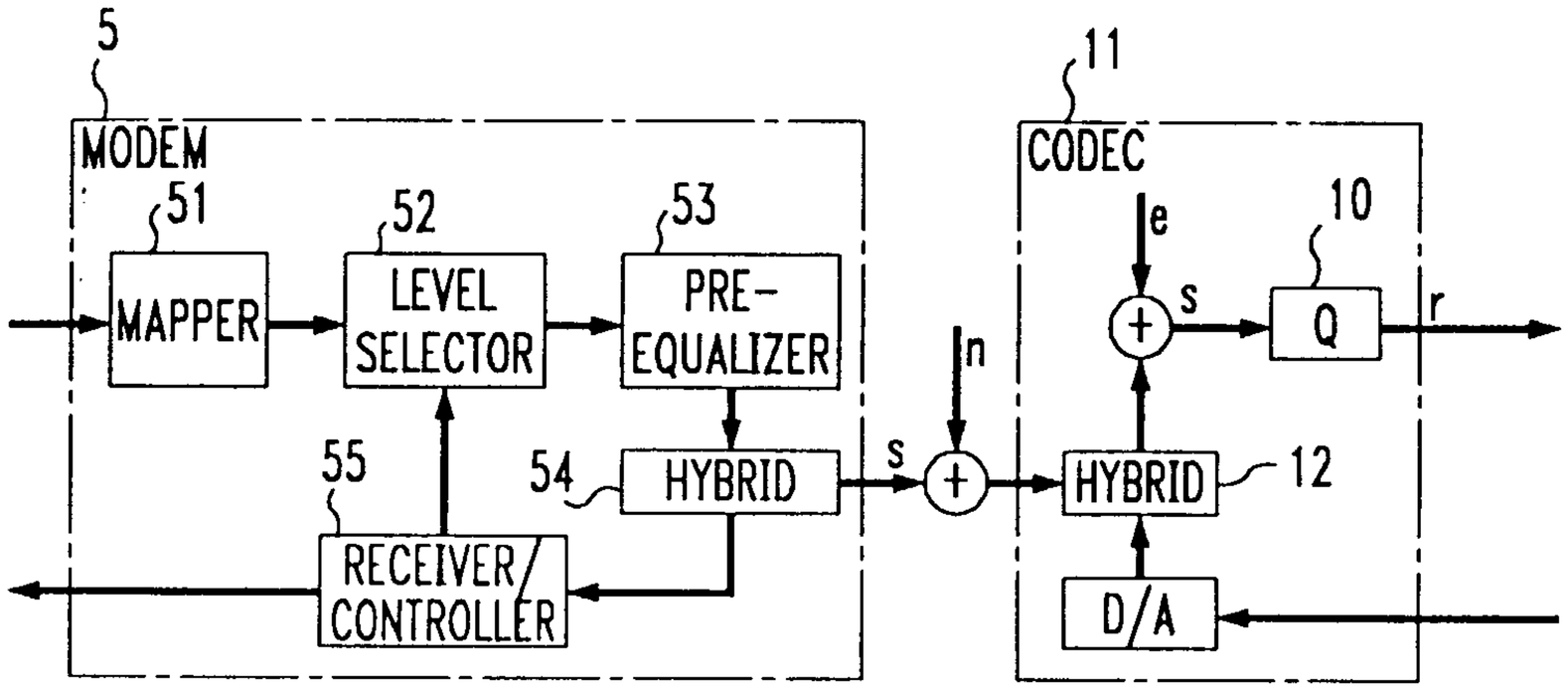
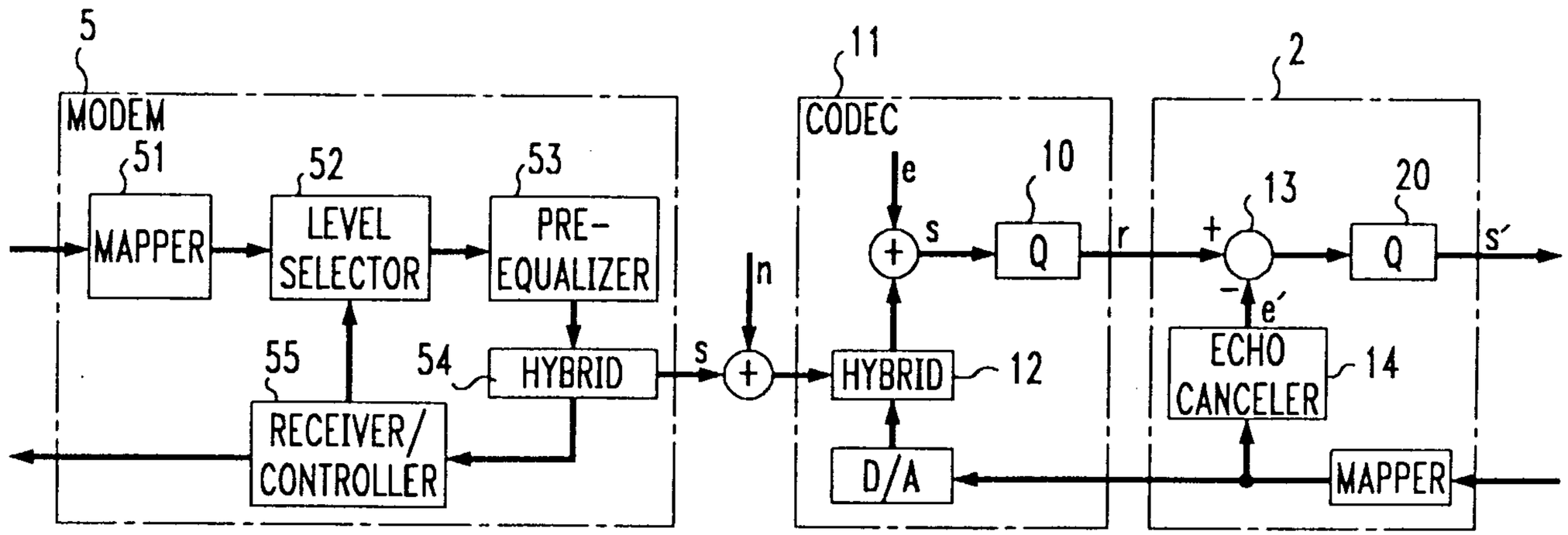


FIG. 2



2167745

FIG. 3

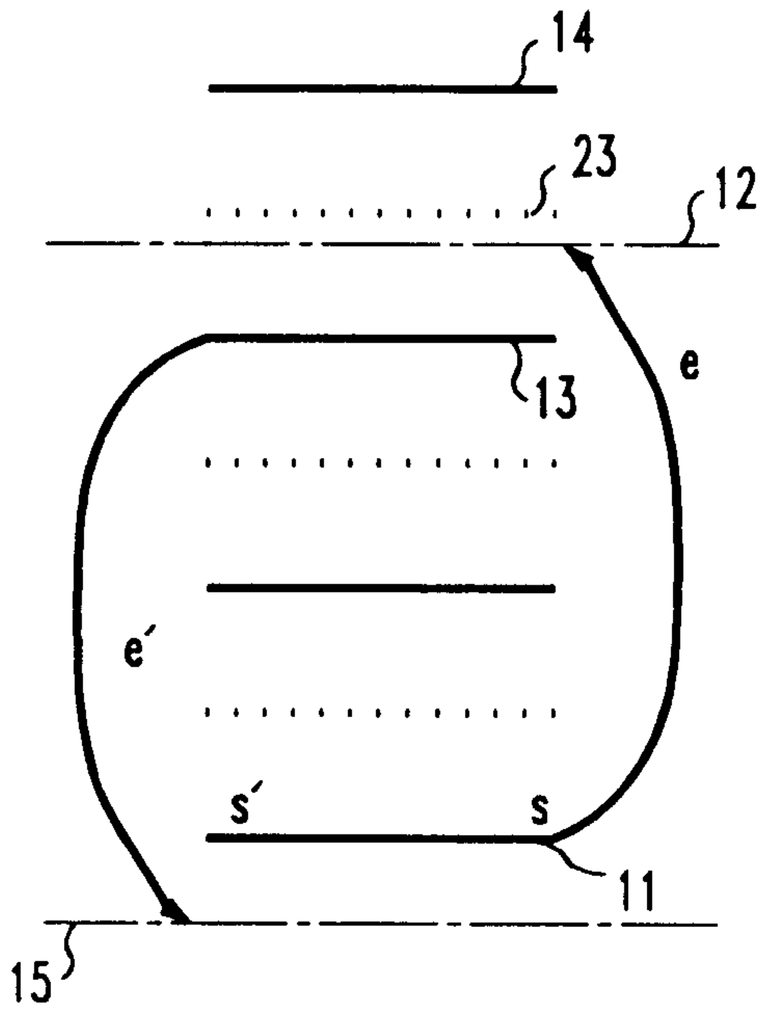


FIG. 4

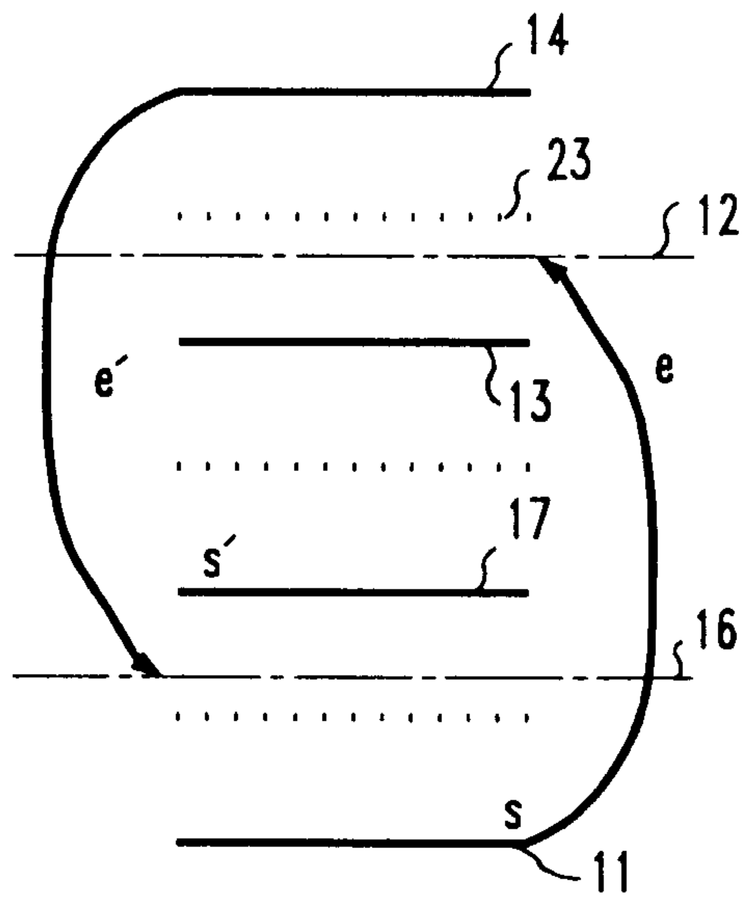
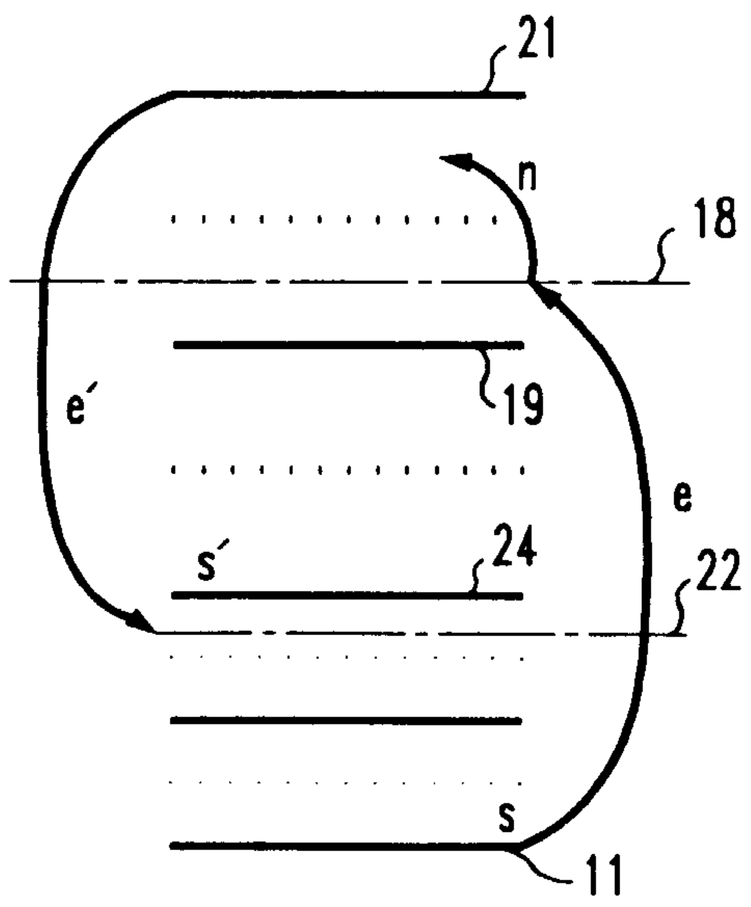


FIG. 5



2/8

2167745

Kirby, Eades, Gale, Baker

2167745

FIG. 6

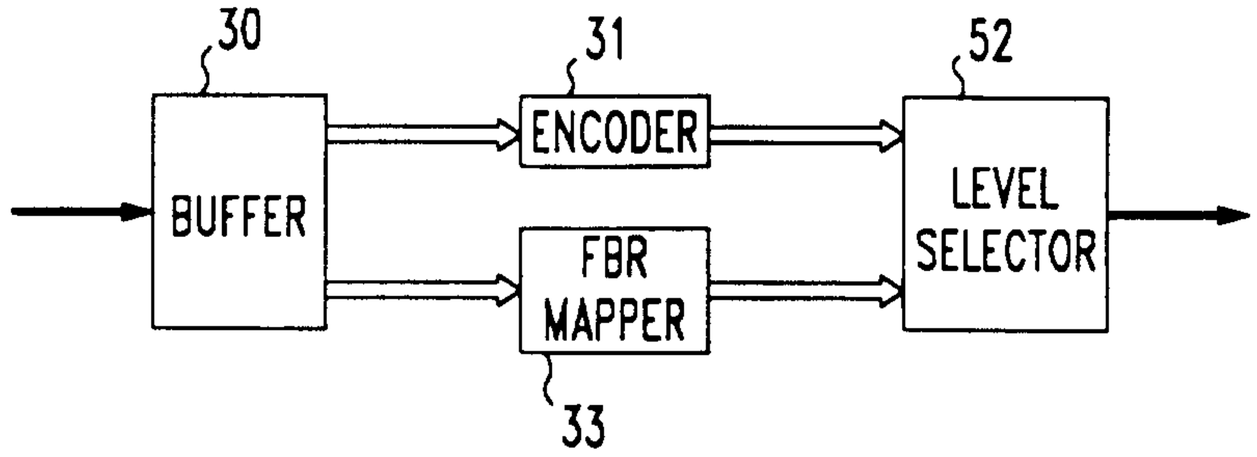


FIG. 7

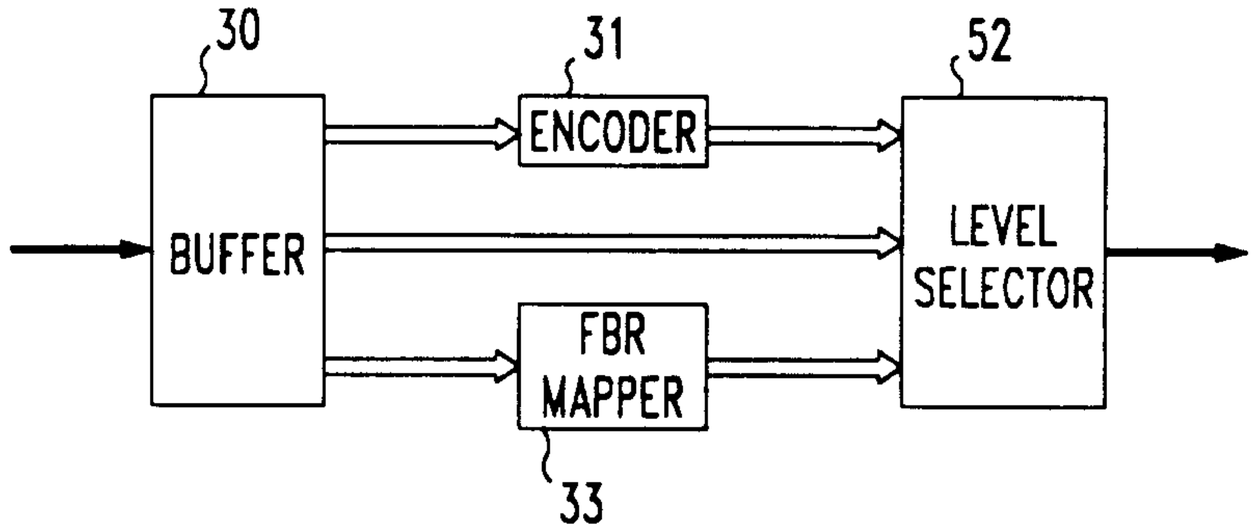
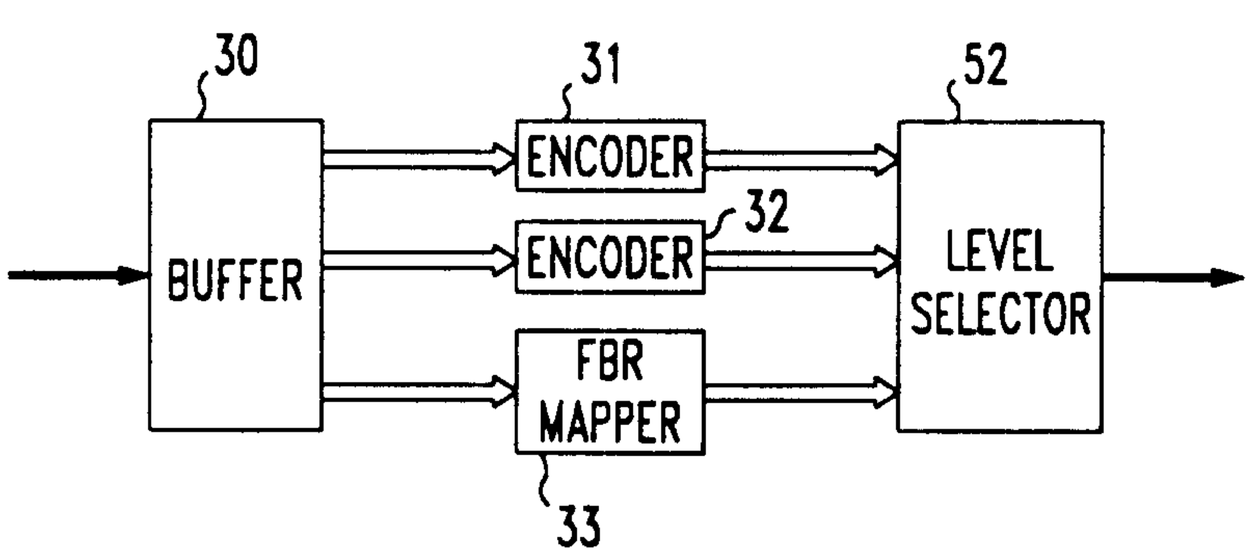


FIG. 8



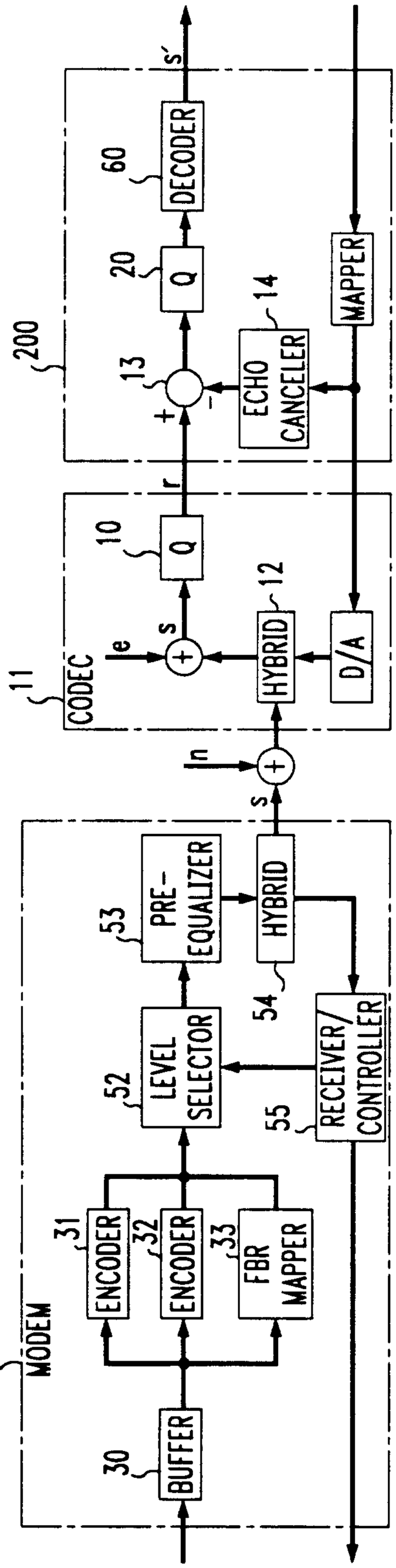

MODEM
5
CODEC
11
200
ENCODER
31
ENCODER
32
FBR MAPPER
33
BUFFER
30
LEVEL SELECTOR
52
PRE-EQUALIZER
53
RECEIVER/ CONTROLLER
55
HYBRID
54
HYBRID
12
Q
10
Q
20
ECHO CANCELER
14
MAPPER
D/A
e
s
n
r
s'
DECODER
60

FIG. 9

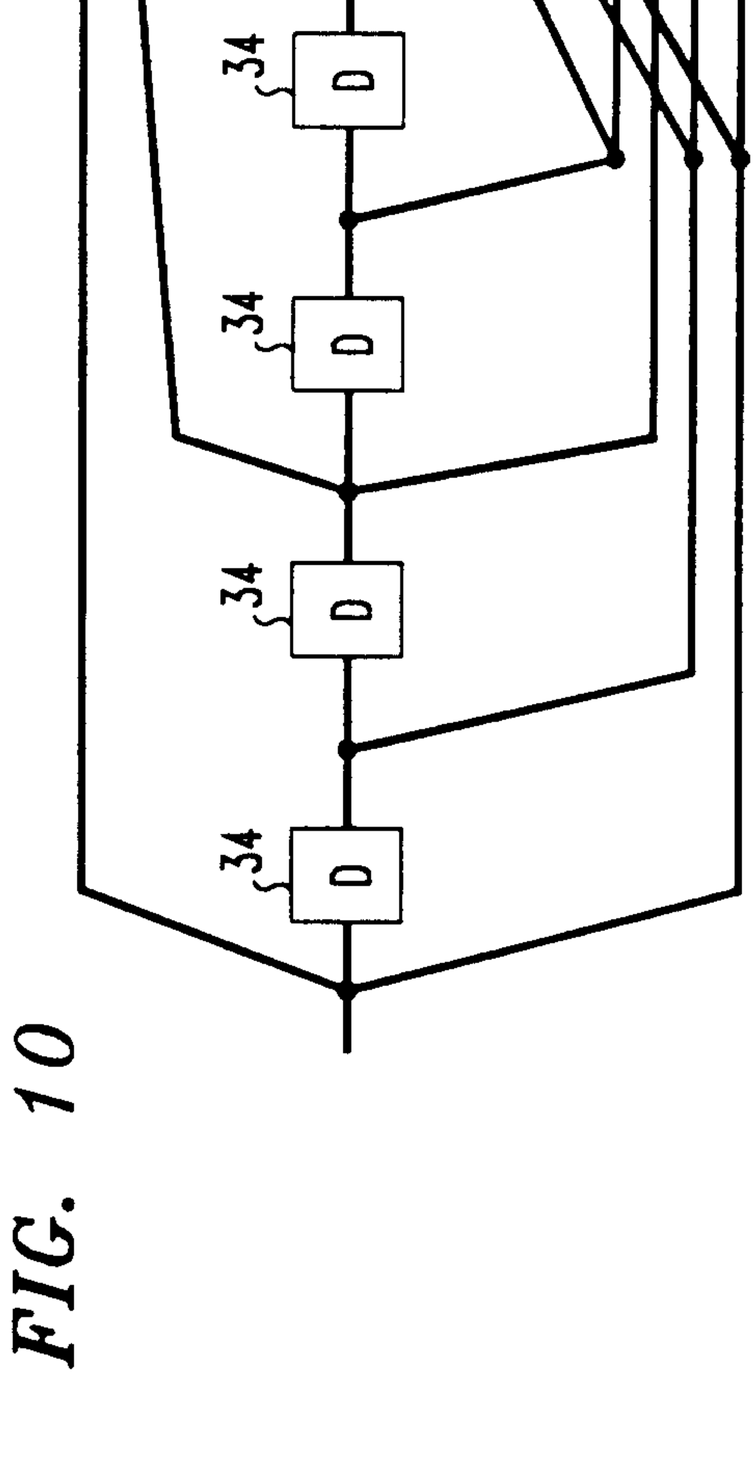
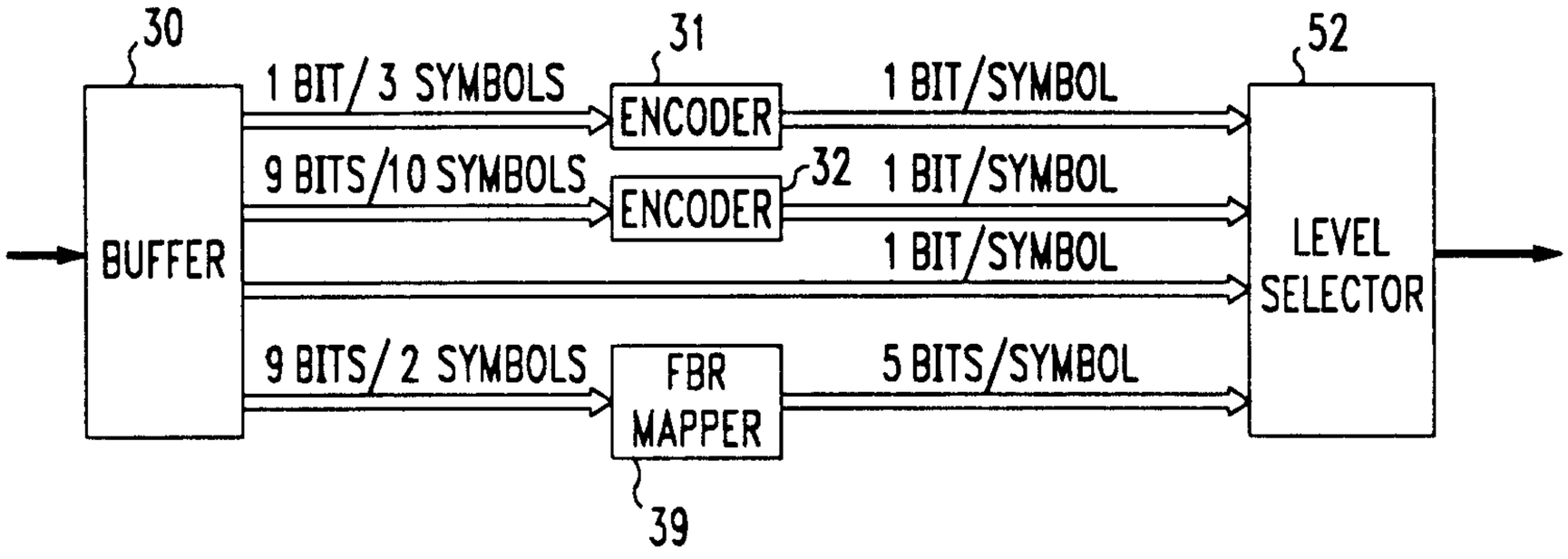

35
36
37
38
D
34

FIG. 10

2167745

FIG. 11



2167745

FIG. 12A

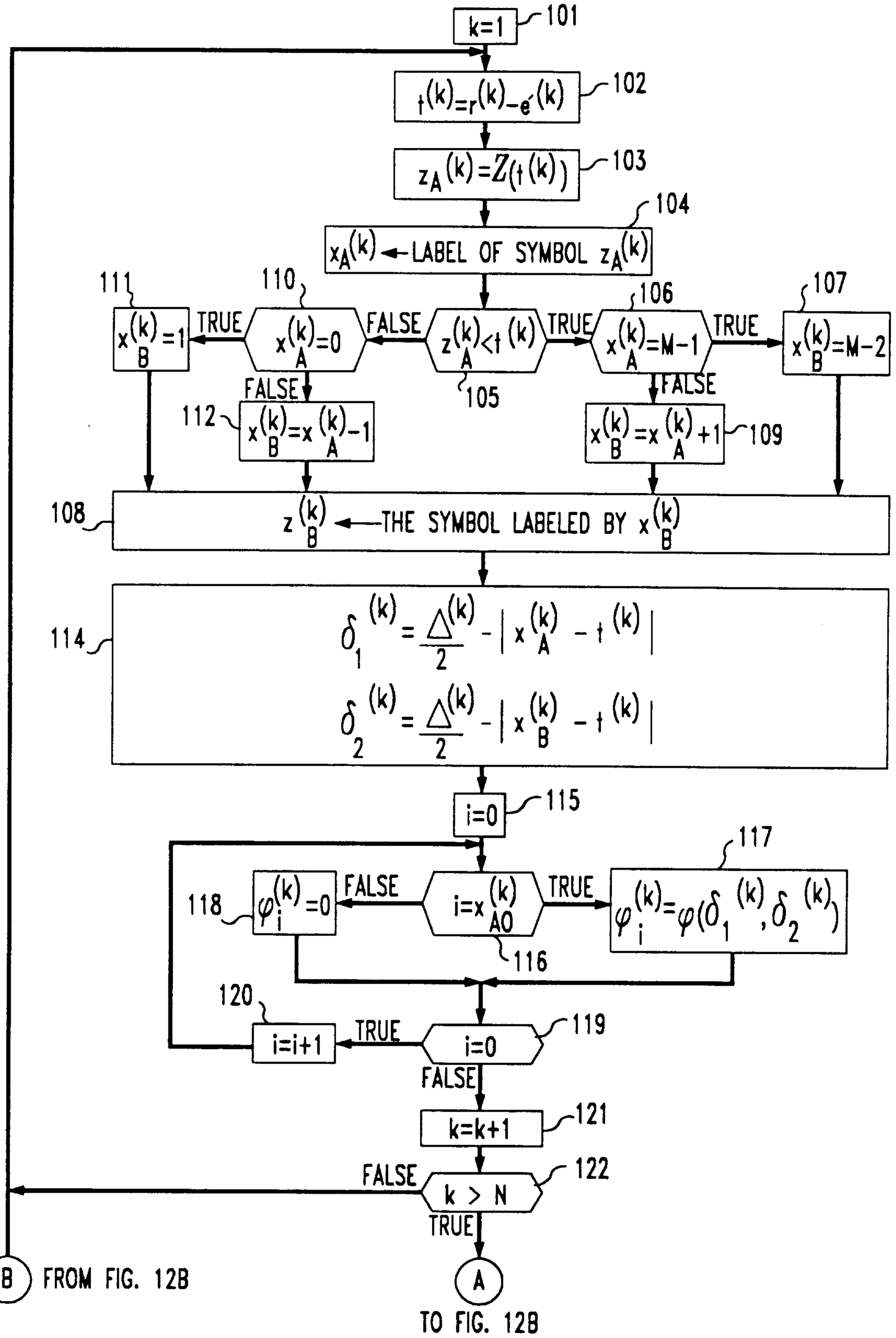


FIG. 12B

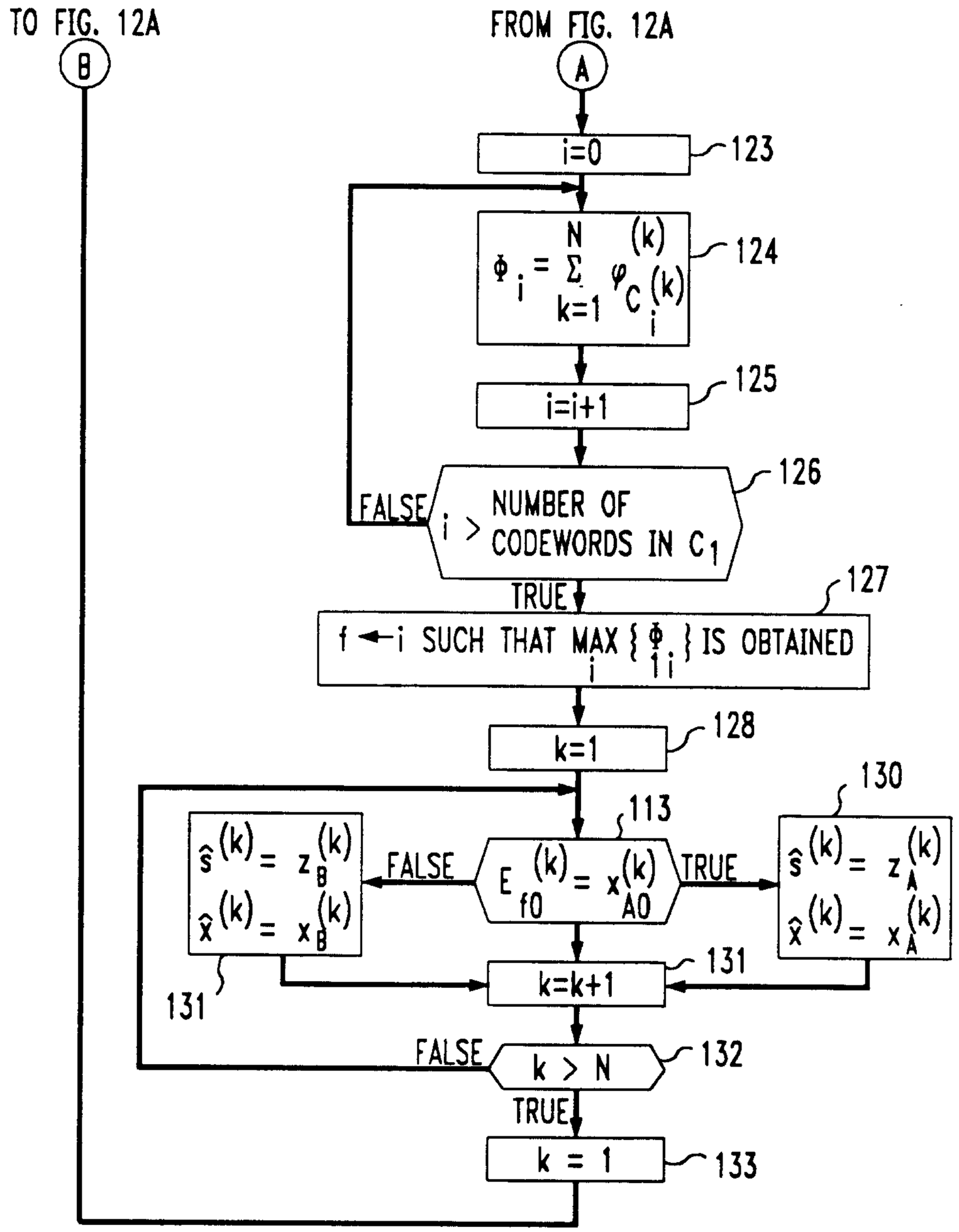

TO FIG. 12A
B
FROM FIG. 12A
A
i=0
123
$\Phi_i = \sum_{k=1}^{N} \varphi_{C_i}(k)$
124
i=i+1
125
126
FALSE
i > NUMBER OF CODEWORDS IN C_1
TRUE
127
f ← i SUCH THAT MAX { Φ_i } IS OBTAINED
i
k=1
128
130
113
$\hat{s}(k) = z_B(k)$
$\hat{x}(k) = x_B(k)$
FALSE
$E_{f0}(k) = x_{A0}(k)$
TRUE
$\hat{s}(k) = z_A(k)$
$\hat{x}(k) = x_A(k)$
131
131
k=k+1
FALSE
k > N
132
TRUE
k = 1
133

FIG. 13

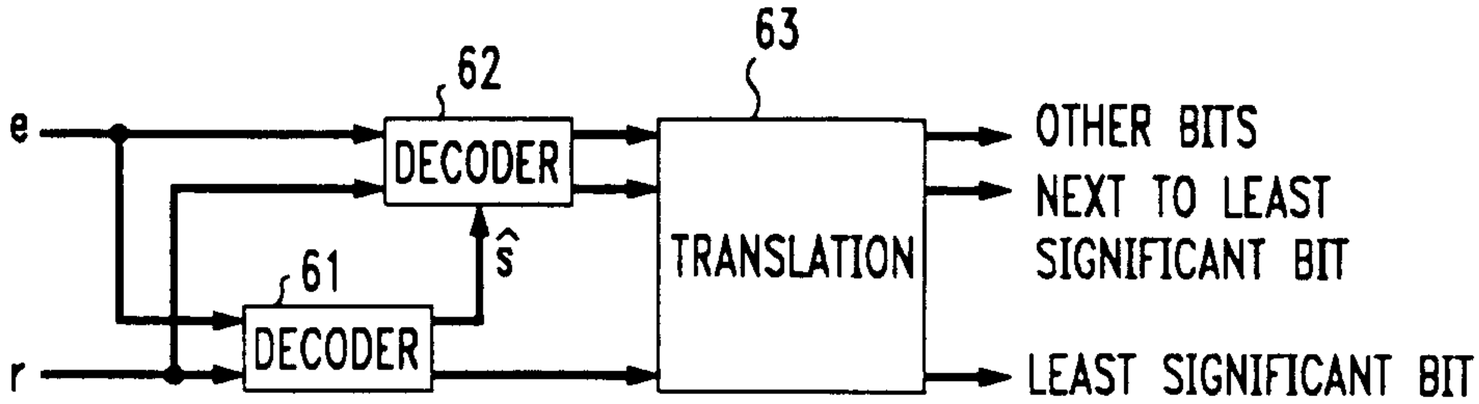
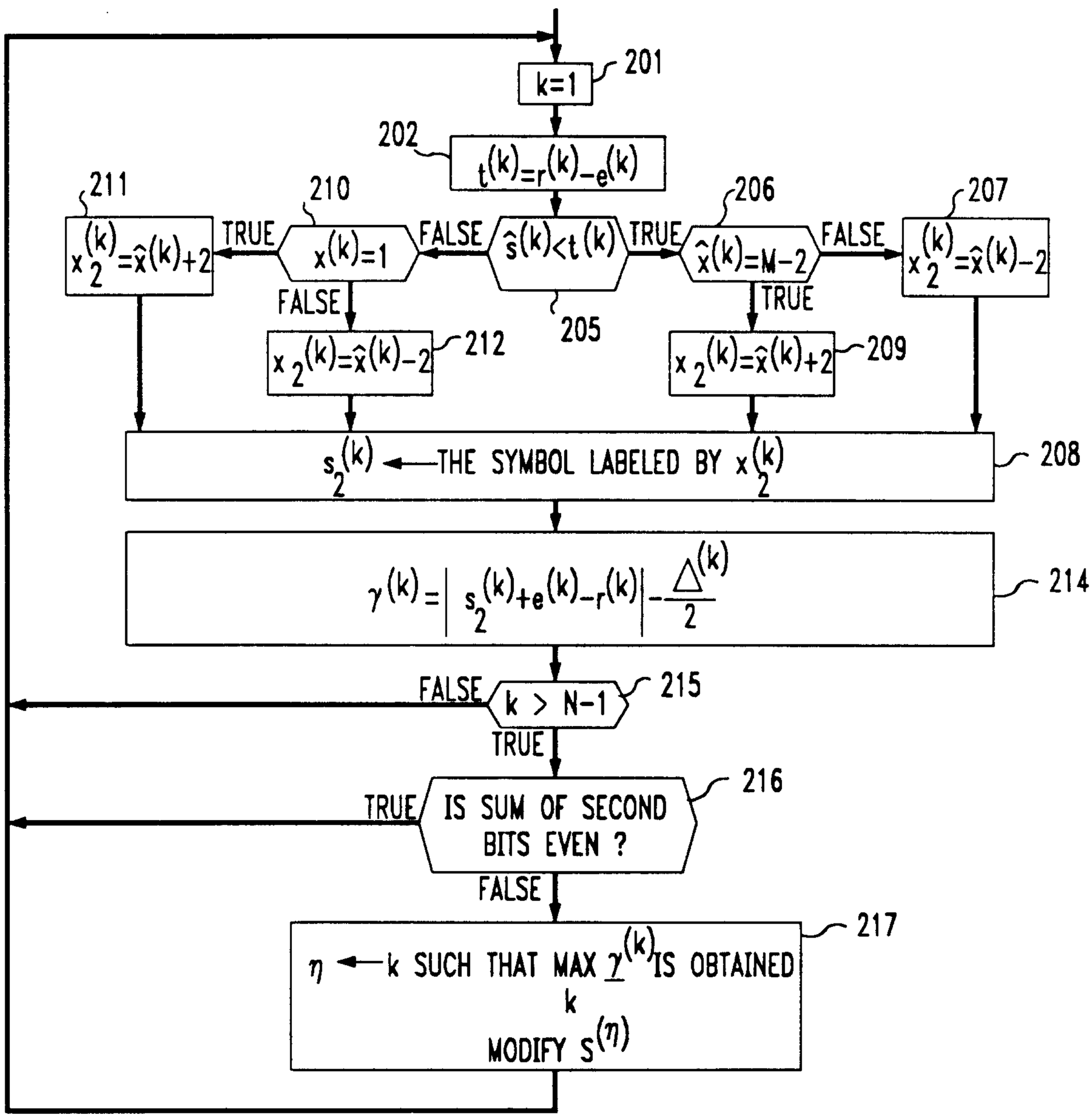
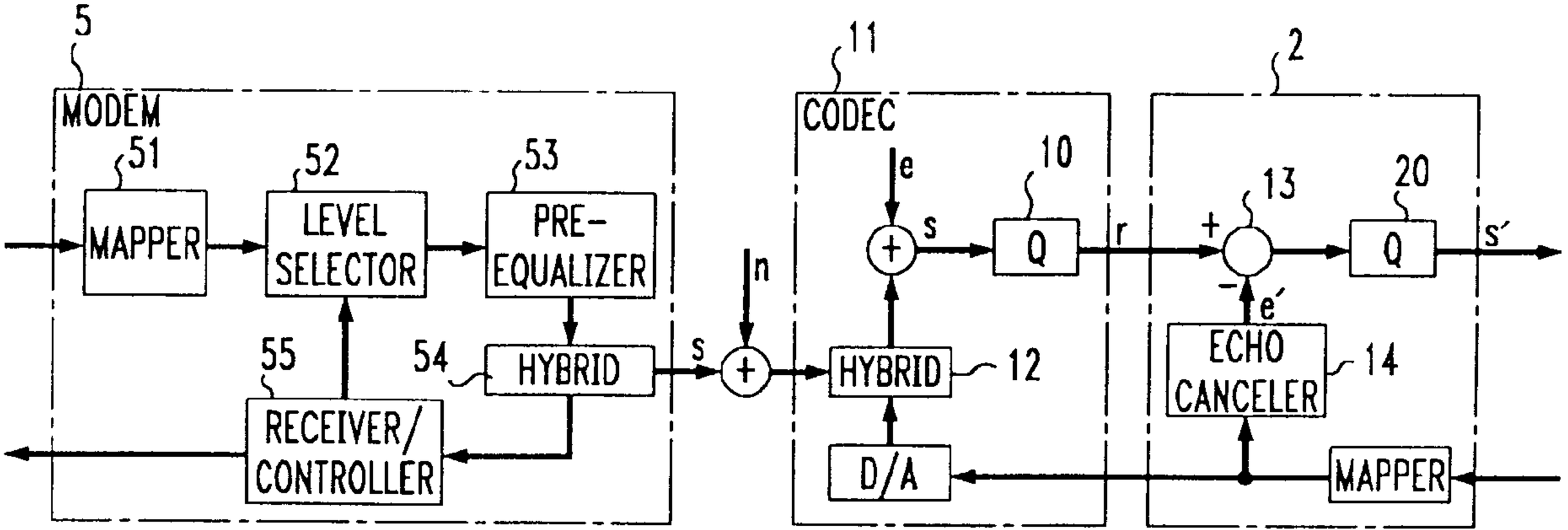

e
r
62
DECODER
61
DECODER
ŝ
63
TRANSLATION
OTHER BITS
NEXT TO LEAST SIGNIFICANT BIT
LEAST SIGNIFICANT BIT

FIG. 14


k=1
201
$t^{(k)}=r^{(k)}-e^{(k)}$
202
$\hat{s}^{(k)}<t^{(k)}$
205
TRUE
FALSE
$\hat{x}^{(k)}=M-2$
206
TRUE
FALSE
$x^{(k)}=1$
210
TRUE
FALSE
$x_2^{(k)}=\hat{x}^{(k)}+2$
211
$x_2^{(k)}=\hat{x}^{(k)}-2$
207
$x_2(k)=\hat{x}(k)-2$
212
$x_2(k)=\hat{x}(k)+2$
209
$s_2^{(k)}$ ← THE SYMBOL LABELED BY $x_2^{(k)}$
208
$\gamma^{(k)}=\left| s_2^{(k)}+e^{(k)}-r^{(k)} \right|-\frac{\Delta^{(k)}}{2}$
214
FALSE
$k > N-1$
215
TRUE
IS SUM OF SECOND BITS EVEN ?
216
TRUE
FALSE
η ← k SUCH THAT MAX $\gamma^{(k)}$ IS OBTAINED
k
MODIFY $s^{(\eta)}$
217


5
MODEM
51
MAPPER
52
LEVEL SELECTOR
53
PRE-EQUALIZER
55
54
HYBRID
RECEIVER/ CONTROLLER
s
n
11
CODEC
e
10
s
Q
r
HYBRID
12
D/A
2
13
20
+
−
Q
s´
e´
ECHO CANCELER
14
MAPPER