

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
29 June 2006 (29.06.2006)

PCT

(10) International Publication Number
WO 2006/067628 A1

(51) International Patent Classification:
H03M 13/13 (2006.01)

(74) Agent: SANTARELLI; 14, Avenue de la Grande Armée,
BP 237, F-75822 Paris Cedex 17 (FR).

(21) International Application Number:
PCT/IB2005/004034

(81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, LY, MA, MD, MG, MK, MN, MW, MX, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SM, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.

(22) International Filing Date:
22 December 2005 (22.12.2005)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
0413840 23 December 2004 (23.12.2004) FR

(71) Applicant (for all designated States except US): CANON KABUSHIKI KAISHA [JP/JP]; 3-30-2, Shimomaruko, 3-chome, Ohta-ku, Tokyo (JP).

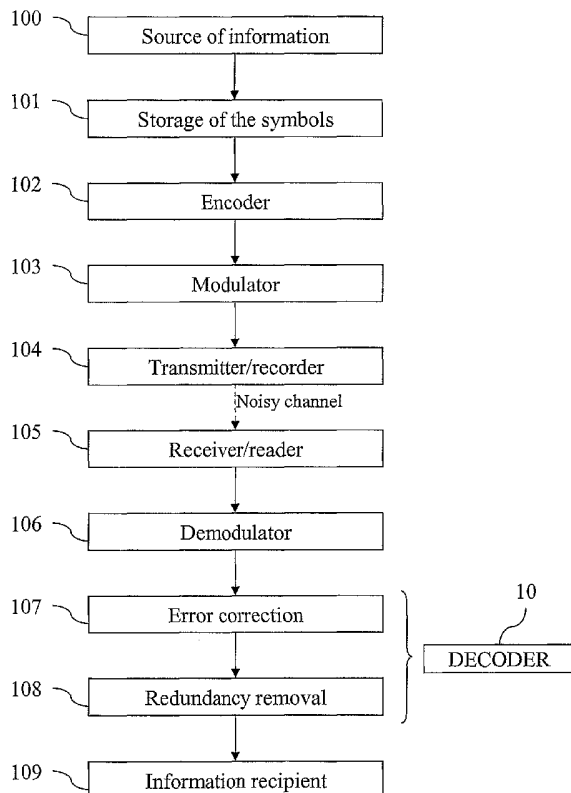
(84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, LV, MC, NL, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

(72) Inventors; and

(75) Inventors/Applicants (for US only): PIRET, Philippe [BE/FR]; 4, boulevard des Métairies, F-35510 Cesson-Sevigne (FR). LE BARS, Philippe [FR/FR]; 12, rue du Manoir, F-35235 Thorigne-Fouillard (FR).

[Continued on next page]

(54) Title: A DECODING METHOD FOR ALGEBRAIC GEOMETRIC CODES AND ASSOCIATED DEVICE



(57) Abstract: The present invention relates to a method of decoding a one-point algebraic geometric code defined on an algebraic curve of type $C(a,b)$ represented by an equation $F(X,Y) = 0$ of degree b in X and of degree a in Y over F_q , comprising the following steps: - calculating extended error syndromes $(\sigma_j(i))$ associated with a received word (r) ; - determining the values of errors in each component $(r(x, y_p(x)))$ of the received word r , on the basis of the extended error syndromes calculated. Since the error value is determined for each component, it is not necessary to have recourse to an error locating step. The invention also relates to devices and apparatuses associated with the method.



Declaration under Rule 4.17:

- *of inventorship (Rule 4.17(iv))*

Published:

- *with international search report*
- *before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

**"A decoding method for algebraic geometric codes
and associated device"**

5 The present invention concerns systems for communication or recording of data in which the data are subjected to a channel encoding in order to improve the fidelity of the transmission or storage. It concerns more particularly a decoding method, as well as the devices and apparatus adapted to implement this method.

10 It will be recalled that channel "block" encoding consists of transmitting "codewords" to a receiver (or recording them on a data carrier) that are formed by introducing a certain level of redundancy into the data to transmit. Considering in more detail, by means of each codeword, the information is transmitted that is initially contained in a predetermined number k of symbols termed "information symbols" taken from an "alphabet" of finite
15 size q ; on the basis of these k information symbols, calculation is made of a number $n > k$ of symbols belonging to that alphabet, which constitute the components of the codewords: $\underline{v} = (v_1, v_2, \dots, v_n)$. The set of codewords obtained when each information symbol takes some value in the alphabet constitutes a sort of dictionary referred to as a "code" of "dimension" k and "length" n .

20 In particular, when the size q of the alphabet is taken equal to a power of a prime number, the alphabet can be given a field structure known as a "Galois field" denoted $\mathbf{F}_q$. For example, for $q = 2$, $\mathbf{F}_q$ is a binary alphabet, and for $q = 2^8 = 256$, $\mathbf{F}_q$ is an alphabet of bytes.

25 The "Hamming distance" between two words of the same length is the number of indices for which the component of the first word is different from the component of the second word. For a given code, the smallest Hamming distance between any two different words belonging to the code is termed the "minimum distance" d ; this is an important parameter of the code.

30 Certain codes, termed "linear codes" are such that any linear combination of codewords (with the coefficients taken from the alphabet) is still a codeword. These codes can, conveniently, be associated with a matrix H of

dimension $(n-k) \times n$, referred to as a "parity check matrix": a given word $\underline{v}$ of length n is a codeword if, and only if, it satisfies the equation: $H \cdot \underline{v}^T = 0$ (where the exponent T indicates the transposition); the code is then said to be "orthogonal" to the matrix H .

5 There is a transmission error if the difference $\underline{e}$ between a received word $\underline{r}$ and the corresponding codeword $\underline{v}$ sent by the transmitter is non-zero. It is said that the transmission errors are caused by the "channel noise". To detect possible transmission errors and if possible correct them, a decoding method is implemented at the receiver which judiciously exploits the
10 redundancy mentioned above.

More particularly, the decoding is carried out in two main steps.

The first step consists of associating an "associated codeword" $\hat{\underline{v}}$ with the received word. To do this, the decoder first of all calculates the "error syndromes vector" $\underline{s} = H \cdot \underline{r}^T = H \cdot \underline{e}^T$ of length $(n-k)$ (in the context of the
15 present invention, no difference is made between the term "word" and the term "vector"). If the syndromes are all zero, it is assumed that no transmission error has occurred, and the "associated codeword" will then simply be taken to be equal to the received word. If that is not the case, it is thereby deduced that the received word is erroneous, and calculations are then performed that are
20 adapted to estimate the value of the error $\underline{e}$; in other words, these calculations provide an estimated value $\hat{\underline{e}}$ of the error such that $(\underline{r} - \hat{\underline{e}})$ is a codeword, which will then constitute the "associated codeword" $\hat{\underline{v}}$. Usually, this first step of the decoding is divided into two distinct sub-steps: a first so-called "error locating" sub-step, during which the components of the received word are determined of
25 which the value is erroneous, and a second so-called "error correction" sub-step, during which an estimation is calculated of the transmission error affecting those components.

The second step simply consists in reversing the encoding method. If the received word was correct, or if correction was achieved of all the
30 transmission errors therein during the first step, this second step of course

makes it possible to retrieve the k initial information symbols (before encoding) corresponding to that received word.

It will be noted that in the context of the present invention, reference will often be made to "decoding" for brevity, to designate solely the first of those steps, it being understood that the person skilled in the art is capable without difficulty of implementing the second step.

The objective usually assigned to decoding is to associate with the received word the codeword situated at the shortest Hamming distance from that received word. Reasonably, it will be attempted to identify the position of the possible errors in a received word, and to provide the correct replacement symbol (i.e. identical to the one sent by the sender) for each of those positions, each time the number of erroneous positions is at most equal to $\text{INT}[(d-1)/2]$ (where "INT" designates the integer part) for a code of minimum distance d . For certain error configurations, it is sometimes possible to do better. However, it is not always simple to perfect a decoding algorithm achieving that performance. It should also be noted that, when the chosen algorithm manages to propose a correction for the received word, that correction is all the more reliable (at least, for most transmission channels) the smaller the number of positions it concerns.

Among known codes, "Reed-Solomon" codes may be cited, which are reputed for their efficiency. They are linear codes, of which the minimum distance d is equal to $(n-k+1)$. The parity-check matrix H of the Reed-Solomon code of dimension k and length n (where n is less than or equal to $(q-1)$) is a matrix with $(n-k)$ lines and n columns, which has the structure of a Vandermonde matrix. This parity matrix H , which may be defined for example by taking $H_{ij} = \alpha^{i(j-1)}$ ($1 \leq i \leq n-k$, $1 \leq j \leq n$), where α is an n^{th} root of unity in $\mathbf{F}_q$; it is then possible to label the component v_j , where $1 \leq j \leq n$, of any codeword $\underline{v} = (v_1, v_2, \dots, v_n)$ by means of the element $\alpha^{(j-1)}$ of $\mathbf{F}_q$; it is for this reason that a set such as $(1, \alpha, \alpha^2, \dots, \alpha^{n-1})$ is termed "locating set" of the Reed-Solomon code.

Like all codes, Reed-Solomon codes may be "modified" and/or "shortened". It is said that a given code C_{mod} is a "modified" version of the code C if there is a square non-singular diagonal matrix A such that each word of C_{mod} is equal to $\underline{y} \cdot A$ with $\underline{y}$ being in C . It is said that a given code is a

5 "shortened" version of the code C if it comprises solely the words of C of which, for a number R of predetermined positions, the components are all zero: as these positions are known to the receiver, their transmission can be obviated, such that the length of the shortened code is $(n - R)$.

As mentioned above, the step of a method of decoding during which

10 a "codeword associated with the received word" is calculated is usually divided into two sub-steps: the first sub-step referred to as an "error locating" sub-step, consists of identifying in the received word any components whose value is erroneous; and the second sub-step consists then of calculating the corrected value of those erroneous components.

15 For the decoding of Reed-Solomon codes, as regards *error locating*, use is usually made of the algorithm known as the "Berlekamp-Massey" algorithm, which will now be briefly described: firstly a matrix S is constructed, termed "syndromes matrix", of which each element is a certain component of the error syndromes vector $\underline{s} = H \cdot \underline{r}^T = H \cdot \underline{e}^T$; next a vector $\underline{\Lambda}$ is

20 sought such that $\underline{\Lambda} \cdot S = 0$, then an "error locating polynomial" $\Lambda(Z)$ is formed of which the coefficients are components of the vector $\underline{\Lambda}$; the inverses of the roots of that polynomial $\Lambda(Z)$ are then, among the elements ω_i (where $i = 1, \dots, n$) of the locating set, those which label the erroneous components of the received word $\underline{r}$.

25 As regards the *error correction*, use is usually made of the algorithm known as the "Forney" algorithm which will now be briefly described. The error calculating polynomial $\Omega(Z) = \Lambda(Z)S(Z)$ modulo Z^{n-k} is constructed, where

$$S(Z) = \sum_{i=0}^{n-k-1} s_i Z^i$$

and the s_i are the components of the error syndromes vector $\underline{s}$; the errors are then given, for $i = 1, \dots, n$, by:

$$e_i = \begin{cases} 0 & \text{si } \Lambda(\omega_i^{-1}) \neq 0 \\ -\frac{\Omega(\omega_i^{-1})}{p_i \Lambda'(\omega_i^{-1})} & \text{si } \Lambda(\omega_i^{-1}) = 0 \end{cases},$$

where $\Lambda'(Z)$ designates the derivative of $\Lambda(Z)$, and p_i is equal to 1 for a "standard" Reed-Solomon code and to the diagonal element in position (i, i) of the matrix P for a modified code (see above).

For more details on Reed-Solomon codes, and in particular the algorithms of Berlekamp-Massey and of Forney, reference may for example be made to the work by R.E. Blahut entitled "*Theory and practice of error-control codes*", Addison-Wesley, Reading, Mass., 1983.

For modern information carriers, for example on computer hard disks, CDs ("*compact discs*") and DVDs ("*digital video discs*"), it is sought to increase the density of information. When such a carrier is affected by a physical defect such as a scratch, a high number of information symbols may be rendered unreadable. This problem may nevertheless be remedied by using a very long code. However, as indicated above, the length n of the words in Reed-Solomon codes is less than the size q of the alphabet of the symbols. Consequently, if a Reed-Solomon code is desired having codewords of great length, high values of q must be envisaged, which leads to costly implementations in terms of calculation and storage in memory. Moreover, high values of q are sometimes ill-adapted to the technical application envisaged. For this reason, it has been sought to build codes which naturally provide words of greater length than Reed-Solomon codes without however requiring a longer alphabet.

In particular so-called "algebraic geometric codes" or "Goppa geometric codes" have been proposed (see for example "*Algebraic Geometric Codes*" by par J.H. van Lint, in "*Coding Theory and Design Theory*" IMA Volumes Math. Appl., vol. 21, pages 137 to 162, Springer-Verlag, Berlin, 1990). These codes are constructed from a set of n distinct pairs (x, y) of symbols

belonging to a chosen Galois field $\mathbf{F}_q$; this set of pairs constitutes the locating set of the algebraic geometric code. In general terms, there is an algebraic equation with two unknowns X and Y such that the pairs (x, y) of that locating set are all solutions of that algebraic equation. The values of x and y of these
 5 pairs may be considered as coordinates of "points" P_β (where $\beta = 1, \dots, n$) forming an "algebraic curve".

An important parameter of such a curve is its "genus" g . In the particular case where the curve is a simple straight line (the genus g is then zero), the algebraic geometric code reduces to a Reed-Solomon code. For
 10 given q and g , certain algebraic curves, termed "maximum", make it possible to achieve a length equal to $(q + 2g\sqrt{q})$, which may be very high; for example, with an alphabet size of 256 and a genus equal to 120, codewords are obtained of length 4096.

Among all the algebraic geometric codes, those usually considered are
 15 the ones which are defined on an algebraic curve represented by an equation $F(X, Y) = 0$, with:

$$F(X, Y) = X^b + cY^a + \sum c_{ij} Y^j X^i,$$

where $c \neq 0$ and the c_{ij} are elements of $\mathbf{F}_q$, a and b are strictly positive mutually prime integers, and where the sum only applies to the integers i and
 20 j which satisfy $ai + bj < ab$. An equation of this form will be referred to as "of type $C(a, b)$ ", and the codes defined on a curve of type $C(a, b)$ will be referred to as "codes of type $C(a, b)$ ". The genus g of a curve of type $C(a, b)$ is equal to $(a-1)(b-1)/2$. Moreover, in the following portion of the present text, and without losing generality (given that the names " X " and " Y " of the unknowns
 25 may be exchanged), it will be assumed that $a < b$.

With every monomial of the form $Y^j X^i$, where i and j are positive integers or zero, a "weight" is conventionally associated, which is equal by definition to $(ai + bj)$. It can be shown that the monomials $h_\alpha = Y^j X^i$ of weight $\rho_\alpha \leq m$, where $\alpha = 1, \dots, n-k$, and j is an integer between 0 and $(a-1)$,

constitute a basis of the vector space $L(mP_\infty)$ of the polynomials in X and Y with coefficients in $\mathbb{F}_q$ of which solely the poles are situated at infinity and are of order less than or equal to m , where $0 < m < n$. It can also be shown that this vector space is of dimension greater than or equal to $(m - g + 1)$ (equal if
 5 $m > 2g - 2$).

For any code of type $C(a, b)$, a parity-check matrix H is conventionally defined in the following manner: the element in line α and column β of the parity-check matrix is equal to the monomial h_α evaluated at the point P_β (where, it may be recalled, $\beta = 1, \dots, n$) of the algebraic curve. Each point
 10 P_β then serves to identify the β^{th} component of any codeword. A code having such a parity-check matrix is termed a "one-point" code since its parity-check matrix is obtained by evaluating (at the n points P_β) functions (the monomials h_α) which have poles only at a single point, i.e. the point at infinity.

The codewords $\underline{c}$ are such that $H \cdot \underline{c}^T = 0$ and it is possible on account of this fact in particular to designate a component of a codeword $\underline{c}$ (and
 15 also of a received word) by referring to the corresponding point P_β in the locating set (in general itself expressed by its coordinates x, y). It can thus be said that the components of the word are "labeled" by means of the points of the locating set.

20 Algebraic geometric codes are advantageous as to their minimum distance d , which is at least equal to $(n - k + 1 - g)$, and, as has been said, as to the length of the codewords, but they have the drawback of requiring decoding algorithms that are rather complex, and thus rather expensive in terms of equipment (software and/or hardware) and processing time. This
 25 complexity is in fact greater or lesser according to the algorithm considered, a greater complexity being in principle the price to pay for increasing the error correction capability of the decoder (see for example the article by Tom Høholdt and Ruud Pellikaan entitled "On the Decoding of Algebraic-Geometric Codes", *IEEE Trans. Inform. Theory*, vol. 41 no. 6, pages 1589 to 1614, November
 30 1995). Generally, the higher the genus g of the algebraic curve used, the

greater the length of the codewords, but also the greater the complexity of the decoding.

It is sometimes useful to shorten an algebraic geometric code. In particular, it is quite common to delete a point from the locating set, or several
5 points, of which the x coordinate is zero.

Various *error locating* algorithms are known for algebraic geometric codes (defined on a curve of non-zero genus).

Such an algorithm, termed "basic" algorithm, has been proposed by A.N. Skorobogatov and S.G. Vlăduț in the article entitled "*On the Decoding of*
10 *Algebraic-Geometric Codes*", *IEEE Trans. Inform. Theory*, vol. 36 no. 5, pages 1051 to 1060, November 1990). This algorithm uses a "syndromes matrix" S , of dimension $(n-k) \times (n-k)$, of which each coefficient S_{ij} , where j is less than or equal to a "boundary" value $w(i)$, is equal to a judiciously chosen linear combination of the elements s_ν ($\nu = 1, 2, \dots, n-k$) of the syndrome $\underline{s}$, the
15 coefficients S_{ij} beyond the boundary remaining indeterminate. The basic algorithm makes it possible to construct, on the basis of the syndromes matrix S , an "error locating polynomial" $\Lambda(x, y)$, of which the zeros comprise all the pairs (x, y) labeling the positions of the received word for which the component in that position has suffered a transmission error.

20 Skorobogatov and Vlăduț have also proposed, in the same article cited above, a "modified" version of the "basic" algorithm, which generally enables a higher number of errors to be corrected than the "basic" algorithm.

Algorithms are also known which operate using an iterative principle: each new iteration of such an algorithm invokes a supplementary component of
25 the error syndromes vector $\underline{s} = H \cdot \underline{r}^T$.

An example of such an iterative decoding algorithm is disclosed in the article by M. Sakata *et al.* entitled "*Generalized Berlekamp-Massey Decoding of Algebraic-Geometric Codes up to Half the Feng-Rao Bound*" (*IEEE Trans. Inform. Theory*, vol 41, pages 1762 to 1768, November 1995). This
30 algorithm can be viewed as a generalization of the Berlekamp-Massey algorithm to algebraic geometric codes defined on a curve of non-zero genus.

Another example of an iterative decoding algorithm for algebraic geometric codes has been disclosed in the article by M.E. O'Sullivan entitled "*Decoding of Codes Defined by a Single Point on a Curve*" (IEEE Trans. Inform. Theory, vol. 41, pages 1709 to 1719).

5 For any given received word $\underline{r}$, it is possible to determine error locating polynomials of which the zeros comprise all the pairs (x, y) labeling the erroneous components of that received word. The set of the error locating polynomials has the structure of an ideal termed a "Gröbner ideal" (associated with the transmission errors affecting that word). It is possible to generate the

10 Gröbner ideal by means of a finite set of f polynomials, where $f \leq a$, which constitutes a "Gröbner basis" of the ideal. As the pairs (x, y) labeling the erroneous components of a received word of course all satisfy the equation $F(X, Y) = 0$ of the algebraic curve, the polynomial $F(X, Y)$ forms part of the Gröbner ideal, and it is consequently possible to reduce modulo $F(X, Y)$ the f

15 polynomials $G_\varphi(X, Y)$ of any given Gröbner basis; the result of this is that there is always a Gröbner basis of which all the elements have a degree in Y less than a .

It is possible to obtain a Gröbner basis $\mathcal{G} = \{G_\varphi(X, Y) \mid \varphi = 1, \dots, f\}$ from a matrix S^* , of size $n \times n$, obtained by "extending" the matrix S (in other

20 words, the elements of S and those of S^* are identical for $j \leq w(i)$ with $i \leq n - k$). This extension is possible each time the number of errors in the received word is less than or equal to $(n - k - g)/2$.

Thus, when the number of errors in the received word is less than or equal to $(n - k - g)/2$, it is in general necessary, in order to be able to locate

25 those errors, to know further elements of the syndromes matrix than the elements which we will qualify as "known" due to the fact that they are equal to components of the error syndromes vector $\underline{s} = H \cdot \underline{r}^T$ or to simple linear combinations of those components (see the numerical example described below). It is fortunately possible to calculate these elements of "unknown" value

30 by a method comprising a certain number of "majority decisions", for example

the algorithm known as the "Feng-Rao" algorithm (see article by G.-L. Feng and T.R.N. Rao entitled "*Decoding Algebraic-Geometric Codes up to the Designed Minimum Distance* », IEEE Trans. Inform. Theory, vol. 39, no. 1, January 1993). The object of this algorithm is essentially to extend the matrix S by means of

5 calculation steps having the role of successive iterations. A number of iterations is necessary equal to a certain number g' , where g' is at most equal to $2g$, in order to reach the state where, as explained above, it becomes possible to calculate a Gröbner basis from the "extended" syndromes matrix S^* so obtained. At this stage, it is also possible to calculate additional "unknown"

10 elements of the matrix S^* from elements obtained previously, either by means of new iterations of a "majority decisions" algorithm, or more conveniently by means of a certain number of relationships, known as "recursion" relationships, using "feedback polynomials" chosen from the Gröbner basis. In relation to this, reference can be made to the article by Sakata *et al.* cited above.

15 In the context of the present invention, it will be said that the elements of the syndromes matrix S^* ("known" or "unknown") are "extended error syndromes".

Moreover, various algorithms are known for algebraic geometric codes making it possible to *calculate the corrected value* of the erroneous

20 components of the received word; in other words, these algorithms are adapted to provide an estimated value $\hat{e}$ of the transmission error e suffered by the transmitted codeword.

The calculation of errors for algebraic geometric codes is *prima facie* more complicated than for Reed-Solomon codes. This is because:

- 25
- the error locating sub-step not only produces *one* error locating polynomial (such as $\Lambda(Z)$ above), but *several* polynomials, which form a Gröbner basis of the ideal of the error locating polynomials;
 - these error locating polynomials are polynomials with *two* variables instead of one; and
- 30
- these error locating polynomials have partial derivatives with respect to those two variables, such that the conventional correction algorithms (such

as the algorithm of Forney mentioned above), which involve a single derivative, are no longer applicable.

Various error value calculating algorithms are known for algebraic geometric codes.

5 The article "*Algebraic Geometry Codes*", by Tom Høholdt, Jacobus Van Lint and Ruud Pelikaan (Chapter 10 of the "*Handbook of Coding Theory*", North Holland, 1998) constructs the product of certain powers of the polynomials of the Gröbner basis. It then performs a linear combination of those products, allocated with appropriate coefficients. Finally it shows that the value
10 of the polynomial so obtained, taken at the point (x, y) of the locating set, is, with the sign being the only possible difference, the value of the error for the component of the received word labeled by that point (x, y) .

 The article "*A Generalized Forney Formula for Algebraic Geometric Codes*" by Douglas A. Leonard (*IEEE Trans. Inform. Theory*, vol. 42,
15 no. 4, pages 1263 to 1268, July 1996) calculates the values of the errors by evaluating a polynomial with two variables at the points of which the coordinates are the common zeros of the error locating polynomials. The article "*A Key Equation and the Computation of Error Values for Codes from Order Domains*" by John B. Little (published on the Internet on April 7, 2003)
20 calculates the values of the errors by evaluating two polynomials with a single variable at the same points as earlier.

These three algorithms are complex to implement.

European patent application EP-1 434 132 in the name of CANON describes a decoding method applicable in particular to the one point algebraic
25 geometric codes described above defined on an algebraic curve of type $C(a, b)$. This decoding method performs both the location and the correction of errors. It will now be described in some detail.

 This decoding method relies on the subdivision of the locating set of the code into subsets termed "aggregates". By definition, an "aggregate"
30 groups together the pairs (x, y) of the locating set sharing the same value of x , when $a < b$ is taken (still on the assumption that $a < b$, the aggregates could be defined as grouping together the pairs (x, y) of the locating set sharing the

same value of y , but the first definition will be held to in what follows). When it is desired to emphasize this aggregate structure, the pairs of the locating set will be denoted $(x, y_p(x))$, where $p = 1, \dots, \lambda(x)$ and $\lambda(x)$ is the cardinal of the aggregate considered, and the components of any word $\underline{c}$ of length n will be denoted $c(x, y_p(x))$; it will be said that the components of $\underline{c}$ which, labeled in this manner, possess the same value of x form an "aggregate of components" of the word $\underline{c}$; in particular, when it is a received word, it will be said that an aggregate associated with a value x of X is "erroneous" when there exists at least one point (x, y) of the locating set of the code such that the component of said received word labeled by that point is erroneous.

Let m be the maximum weight of the monomials defining the lines of the parity-check matrix (see above). According to application EP-1 434 132, these monomials are classified in sets of monomials

$$M_j = \{Y^j X^i \mid 0 \leq i \leq (m - bj) / a\}$$

for $0 \leq j \leq j_{\max}$, where $j_{\max} < a$. The cardinal of this set M_j is thus:

$$t(j) = 1 + \text{INT} [(m - bj) / a] .$$

Let $x_1, \dots, x_\mu$ denote the different values of x in the locating set, and

$$\underline{v} = [v(x_1, y_1(x_1)), \dots, v(x_1, y_{\lambda(x_1)}(x_1)), \dots, v(x_\mu, y_{\lambda(x_\mu)}(x_\mu))] ,$$

denote any particular codeword. For each aggregate attached to one of the values $x_1, x_2, \dots, x_\mu$ of x , there are constructed $(j_{\max} + 1)$ "aggregate symbols"

$$v_j(x) = \sum_{p=1}^{\lambda(x)} [y_p(x)]^j v(x, y_p(x))$$

for $j = 0, \dots, j_{\max}$. These aggregate symbols serve to form $(j_{\max} + 1)$ "aggregate words"

$$\underline{v}_j = [v_j(x_1), \dots, v_j(x_\mu)] ,$$

of length μ .

It is easily verified that the condition of belonging to the algebraic geometric code (i.e. $H \cdot \underline{v}^T = 0$) is equivalent to the set of $(j_{\max} + 1)$ equations:

$$H_{t(j)} \cdot \underline{v}_j^T = 0,$$

where the function $t(j)$ is given above and is, by definition,

$$H_t = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_\mu \\ \vdots & \vdots & \vdots & \vdots \\ x_1^{t-1} & x_2^{t-1} & \cdots & x_\mu^{t-1} \end{bmatrix}.$$

However, this matrix H_t is a Vandermonde matrix defined over $\mathbf{F}_q$;

5 consequently, if, for each value of j , it is considered that $H_{t(j)}$ is a parity-check matrix defining a set of codewords $\underline{v}_j$, that set constitutes a Reed-Solomon code. It is then said that the algebraic geometric code considered has been "broken down" into a certain number of "component" Reed-Solomon codes.

10 The advantage of this formulation is that it gives decoding algorithms for Reed-Solomon codes, that are very simple, and that have a very high level of performance, at least for certain types of channels. For example, if a word $\underline{r}$ has been received, calculation is first made, for $j = 0, \dots, j_{\max}$, of the "aggregate received words"

$$15 \quad \underline{r}_j = [r_j(x_1), \dots, r_j(x_\mu)],$$

in which, for $x = x_1, \dots, x_\mu$, the "aggregate received symbols" $r_j(x)$ are given by

$$r_j(x) = \sum_{p=1}^{\lambda(x)} [y_p(x)]^j r(x, y_p(x)). \quad (1)$$

20 Next, the Berlekamp-Massey algorithm is used for locating the erroneous symbols $r_j(x)$ of each word $\underline{r}_j$. Except for accidental compensation (the case arising) in equation (1) for a certain value of j , the aggregate received symbols $r_j(x)$ associated with an "erroneous aggregate" (see definition above) will clearly themselves also be erroneous.

25 Next the Forney algorithm is implemented for the correction of those erroneous symbols, according to the error syndromes vector $\underline{s}_j = H_{t(j)} \cdot \underline{r}_j^T$.

Finally, calculation is made of the symbols $\hat{v}(x, y_p(x))$ of the associated codeword on the basis of the corrected symbols $\hat{r}_j(x)$ using the system of equations

$$\hat{r}_j(x) = \sum_{p=1}^{\lambda(x)} [y_p(x)]^j \hat{v}(x, y_p(x)) \quad (2)$$

5 where j takes a number of different values (the number of equations) at least equal to $\lambda(x)$ (the number of unknowns). This decoding method thus requires that $(j_{\max} + 1)$ be at least equal to $\lambda_{\max}$, where $\lambda_{\max}$ is the greatest among the aggregate cardinals $\lambda(x)$. In what follows it will be accepted that the same condition is satisfied.

10 With respect to the known error correction algorithms generally applicable to the algebraic geometric codes considered, the saving in terms of complexity resulting from the implementation of the method according to application EP-1 434 132 is significant, despite the necessity to implement an error correction algorithm for Reed-Solomon code (for example the Forney
15 algorithm) $\lambda_{\max}$ times, and to solve for each erroneous aggregate labeled by a value x of X a system of equations (2); it will be noted in this connection that the number of equations in each of these systems is at most equal to a (the exponent of Y in the equation representing the algebraic curve), since the size of any aggregate (that is to say the number of solutions, for fixed x , of the
20 equation representing the algebraic curve) is at most equal to a .

It will furthermore be noted that the system of equations (2) is a non-singular Vandermonde system: it thus always possesses one, and only one, solution; moreover, as is well known to the person skilled in the art, the solution of this type of system of linear equations is, advantageously, particularly simple.

25 It may furthermore be noted that the known methods for calculating the values of the errors use the syndromes $\sigma_j(i) = (\underline{Y}^j \underline{X}^i, \underline{e})$, $0 \leq i, 0 \leq j \leq a - 1$ where $\underline{e}$ is the error in the received word, $\underline{Y}^j \underline{X}^i$ is the word obtained by evaluation of the monomial $Y^j X^i$ at the points of the curve used to define the code and $(\underline{u}, \underline{v})$ represents the scalar product of the words $\underline{u}$ and $\underline{v}$. For $0 \leq i \leq t$
30 $(j) - 1$, these syndromes are directly obtained as components of $H \underline{r}^T$, where H

is the parity-check matrix of the algebraic geometric code and $\underline{r}$ is the received word. For $t(j) \leq i$, these syndromes remain properly calculable as soon as the weight of the error is $\leq (n - k - g) / 2$. In this connection reference may be made to the article by Sakata *et al.* cited above. Below the vector of the $\sigma_j(i)$ placed in order of increasing i will be designated by $\underline{\sigma}_j$.

As mentioned above, the method described in the document EP-1 434 132 is designed to be implemented once the erroneous aggregates of a received word have been located. However, recourse to a preliminary locating step has the following two drawbacks:

- the implementation of the method is complicated by the fact that it must include parameters that are adjustable during the individual decoding of each received word, depending on the position of the (possible) erroneous aggregates detected during location, and
- the locating step in itself implies an operational cost.

The invention provides a method of decoding a one-point algebraic geometric code defined on an algebraic curve of type $C(a,b)$ represented by an equation $F(X,Y)=0$ of degree b in X and of degree a in Y over $\mathbf{F}_q$, characterized in that it comprises the following steps:

- calculating extended error syndromes $\sigma_j(i)$ associated with a received word $\underline{r}$;
- determining the values of errors in each component $r(x, y_p(x))$ of the received word $\underline{r}$, on the basis of the extended error syndromes calculated.

Since the error value is determined for each component, it is not necessary to have recourse to the error locating step.

For example, when the algebraic curve is partitioned into aggregates, the step of determining the values of errors in each component on the basis of the extended error syndromes comprises the following steps for each aggregate:

- calculating, on the basis of the extended error syndromes, compound error values each representing a linear combination of the errors in the components of the aggregate;

- determining that the errors in the components of the aggregate are zero if all said compound error values are zero;

- calculating the errors in the components of the aggregate if at least one of said compound error values is not zero.

5 Thus the authors of the present invention realized that it is possible to consider that the estimated errors $\hat{e}(x, y_p(x))$ in each component represent a "zero error" when the corresponding component $r(x, y_p(x))$ is correct; hence, by using a number of equations at least equal to the size of the aggregate, it is possible to obtain all the values of the error estimations $\hat{e}(x, y_p(x))$, whether
10 they be zero or not. The inventors thus wondered what, in practice, would be the cost associated with the decision to process *all* the aggregates *in advance* as being potentially erroneous, which decision makes it possible not to have to distinguish between the aggregates containing errors and those which do not. In this case, as usual, the Gröbner basis of the locator polynomials would be
15 calculated, but the trouble would not be taken to explicitly seek which are the points of the curve that are zeros common to those locators.

They then realized that the cost was in fact rather minimal, and all the more so in that, in certain conditions, the utility of this locating step is particularly mediocre: these are conditions where *all*, or *almost all*, the
20 aggregates are erroneous, for example when the channel considered is sufficiently noisy with respect to the size of the aggregate, since the probability that a given aggregate, labeled by a value x of X , comprises at least one erroneous component is all the greater that the cardinal $\lambda(x)$ of that aggregate is great.

25 The calculation of the extended error syndromes $\sigma_j(i)$ is in practice performed for $j = 0, \dots, j_0 - 1$, where j_0 is at least equal to the maximum value $\lambda_{\max}$ of the cardinals $\lambda(x)$ of the aggregates of the code. Sufficient equations are thus obtained to enable the possible calculation of the errors in each aggregate.

30 From a practical point of view, it is worthwhile deleting the pairs (x, y) for which x is zero from the locating set of the algebraic geometric code.

With this restriction, determination of the values of errors for example comprises the following steps:

- calculating $\underline{\tau}_j = \underline{\sigma}_j \cdot V^{-1}$ for $j = 0, \dots, a - 1$, where V^{-1} is the matrix of dimensions $(q - 1) \times (q - 1)$ of which the element in position (i, j) , $1 \leq i, j \leq q - 1$, is $\alpha^{-(i-1)(j-1)}$, α being a primitive element of $\mathbf{F}_q$;

- for all h such that there is j such that the $(h+1)^{\text{th}}$ component $\tau_j(h)$ of $\underline{\tau}_j$ is non-zero, calculating the error $\hat{e}(x, y_p(x))$ of the components of the aggregate defined by $x = \alpha^h$ in the received word $\underline{r}$ by:

$$\begin{bmatrix} \hat{e}(\alpha^h, y_1(\alpha^h)) \\ \hat{e}(\alpha^h, y_2(\alpha^h)) \\ \vdots \\ \hat{e}(\alpha^h, y_{\lambda(\alpha^h)}(\alpha^h)) \end{bmatrix} = \begin{bmatrix} 1 & \dots & \dots & 1 \\ y_1(\alpha^h) & \dots & \dots & y_{\lambda(\alpha^h)}(\alpha^h) \\ \vdots & \vdots & \vdots & \vdots \\ y_1^{\lambda(\alpha^h)-1}(\alpha^h) & \dots & \dots & y_{\lambda(\alpha^h)}^{\lambda(\alpha^h)-1}(\alpha^h) \end{bmatrix}^{-1} \begin{bmatrix} \tau_0(h) \\ \tau_1(h) \\ \vdots \\ \tau_{\lambda(\alpha^h)-1}(h) \end{bmatrix}$$

- 10 In a possible implementation, for each j , the calculation of the elements of the vector $\underline{\tau}_j$ is performed by a circuit comprising a shift register and receiving as input the elements of the vector $\underline{\sigma}_j$.

According to a another embodiment, determining the values of errors may comprise the following steps:

- 15 - implementing, for $j = 0, \dots, \lambda_{\max} - 1$, by means of the error syndromes polynomial

$$S_j(Z) = \sum_{i=0}^{q-2} \sigma_j(i) Z^i,$$

- an error correction algorithm adapted to Reed-Solomon codes, so as to calculate the error $E_j(x)$ in each component labeled by the element x of $\mathbf{F}_q$ of a word of Reed-Solomon code defined over $\mathbf{F}_q$, and

- calculating the estimations $\hat{e}(x, y_p(x))$ of the respective errors in the components $r(x, y_p(x))$ of $\underline{r}$ by solving the system of equations of type

$$E_j(x) = \sum_{p=1}^{\lambda(x)} [y_p(x)]^j \hat{e}(x, y_p(x)), \quad (3)$$

where equations are chosen that are associated with consecutive values of j at least equal in number to the number of components $\lambda(x)$ of the aggregate considered.

5 The calculation of the estimations $\hat{e}(x, y_p(x))$ may then be performed only for the values of x such that there is at least one value of j for which $E_j(x)$ is non-zero. Calculation of the estimations is thus limited to certain aggregates and it is considered that the error values in the components of the other aggregates are zero.

10 According to a possible feature of implementation, if there is at least one value of $x = \alpha^h$ associated with an erroneous aggregate of cardinal $\lambda(x) < a$, comparison is then made between the members of at least one equation

$$E_j(x) = \sum_{p=1}^{\lambda(x)} [y_p(x)]^j \hat{e}(x, y_p(x)) , \text{ where } \lambda(x) \leq j \leq a-1 ,$$

15 associated with said value of x , if need be after having calculated $E_j(x)$ by means of an error correction algorithm adapted to Reed-Solomon codes.

By virtue of these provisions, it will be possible to detect a possible erroneous correction should said at least one equation prove not to be satisfied by the values of $e(x, y_p(x))$ obtained previously.

20 According to a second aspect the invention concerns a device for correcting errors for a one-point algebraic geometric code defined on an algebraic curve of type $C(a, b)$ represented by an equation $F(X, Y) = 0$ of degree b in X and of degree a in Y over $\mathbf{F}_q$, characterized in that it comprises:

- means for calculating extended error syndromes $\sigma_j(i)$ associated with a
25 received word $\underline{r}$;

- means for determining the values of errors in each component $r(x, y_p(x))$ of the received word $\underline{r}$, on the basis of the extended error syndromes calculated.

When the algebraic curve is partitioned into aggregates, the means for determining the values of errors in each component on the basis of the extended error syndromes may comprise:

- means for calculating, for each aggregate and on the basis of the
5 extended error syndromes, compound error values $\tau_j(h)$, $E_j(x)$ each representing a linear combination of the errors in the components of the aggregate;
- means for determining that the errors in the components of an aggregate are zero if all said compound error values relative to that aggregate
10 are zero;
- means for calculating the errors in the components of an aggregate if at least one of said compound error values relative to that aggregate is not zero.

Such a device may also have features corresponding to those
15 already presented for the decoding method and the corresponding advantages arising therefrom.

The invention also relates to:

- a decoder comprising at least one error correction device as described succinctly above, and at least one redundancy removal device,
- 20 - an apparatus for receiving encoded digital signals comprising a decoder as succinctly described above, and comprising means for demodulating said encoded digital signals,
- a computer system comprising a decoder as succinctly described above, and further comprising at least one hard disk as well as at least one means for
25 reading that hard disk,
- a non-removable data storage means comprising computer program code instructions for the execution of the steps of any one of the decoding methods succinctly described above,
- a partially or wholly removable data storage means comprising computer
30 program code instructions for the execution of the steps of any one of the decoding methods succinctly described above, and
- a computer program containing instructions such that, when said

program controls a programmable data processing device, said instructions cause said data processing device to implement one of the decoding methods succinctly described above.

5 The advantages provided by this decoder, this reception apparatus, this computer system, these data storage means and this computer program are essentially the same as those provided by the error correction methods according to the invention.

10 Other aspects and advantages of the invention will emerge from a reading of the following detailed description of particular embodiments, given by way of non-limiting example. The description refers to the accompanying drawings, in which:

- Figure 1 is a block diagram of a system for transmitting information implementing a method according to the invention,
- Figure 2 represents an apparatus for receiving digital signals
15 incorporating a decoder according to the invention, and
- Figure 3 represents a circuit element that can be used in an embodiment according to the invention.

Figure 1 is a block diagram of a system for transmitting information implementing a method according to the invention.

20 The function of this system is to transmit information of any nature from a source 100 to a recipient or user 109. First of all, the source 100 transforms this information into a series of symbols belonging to a certain Galois field F_q (for example bytes of bits for $q = 2^8$), and transmits these symbols to a storage unit 101, which accumulates the symbols so as to form
25 sets each containing k symbols. Next, each of these sets is transmitted by the storage unit 101 to an encoder 102 which incorporates redundancy therein, so as to construct a word of length n belonging to the chosen code.

The code words so formed are next transmitted to a modulator 103, which associates a modulation symbol with each symbol of the codeword. Next,
30 these modulation symbols are transmitted to a recorder (or a transmitter) 104, which inserts the symbols in a transmission channel. This channel may for example be storage on a suitable carrier such as a DVD or a magnetic disc or a

magnetic tape. It may also correspond to be a wired transmission or wireless transmission as is the case with a radio link.

The message transmitted arrives at a reader (or a receiver) 105, after having been affected by a "transmission noise" whose effect is to modify
5 or erase some of the modulation symbols.

The reader (or receiver) 105 then transmits these symbols to the demodulator 106, which transforms them into symbols of $\mathbf{F}_q$. The n symbols resulting from the transmission of the same codeword are next grouped together into a "received word" in an error correction unit 107, which
10 implements a decoding method according to the invention, so as to provide an "associated codeword". Next, this associated codeword is transmitted to a redundancy removal unit 108, which extracts from it k information symbols by implementing a decoding algorithm that is the reverse of that implemented by the encoder 102. Finally, these information symbols are supplied to their
15 recipient 109.

Units 107 and 108 can be considered to form conjointly a "decoder"
10.

The encoding method according to the invention will now be illustrated, with the aid of a numerical example. It should be noted that this
20 example does not necessarily constitute a preferred choice of parameters for the encoding or decoding. It is only provided here to enable the person skilled in the art to understand more easily the operation of the method according to the invention.

The designation $\mathcal{Q}$ will be given to the "one-point" algebraic geometric
25 code of dimension 22 and length 60, defined in the following manner.

The alphabet of the symbols is constituted by the Galois field $\mathbf{F}_{16}$. As the cardinal of this field is a power of 2 ($16 = 2^4$), the sign "+" is equivalent to the sign "-" before any coefficient of a polynomial with coefficients belonging to that field.

The following "algebraic curve" is considered of genus $g = 6$ constituted
30 by the set of the solutions $(X = x, Y = y)$ of the equation with two unknowns

$$Y^4 + Y + X^5 = 0 \quad (4)$$

over $\mathbf{F}_{16}$. It is found that by giving to X some particular value x in $\mathbf{F}_{16}$, each time there are 4 values $y_p(x)$ ($p=1,2,3,4$) in $\mathbf{F}_{16}$ such that the pair $(x, y_p(x))$ is a solution of equation (4); these solutions of equation (4) are the coordinates of the "finite points of the curve" (the curve also contains a point at infinity denoted P_∞). It is chosen to constitute the locating set by means of all these solutions except those where $x=0$; the locating set thus has a cardinal equal to 60, and it can be divided into 15 aggregates which each have a cardinal $\lambda(x)$ equal to 4. Generally, the designation $\lambda_{\max}$ is given to the maximum value among the cardinals $\lambda(x)$ of the aggregates, here $\lambda_{\max} = 4$. It is recalled that each point P_j of the locating set serves to identify the j -th element of any codeword; the number of such points being here equal to 60, the length n of the code is thus also equal to 60.

Next, the vector space $L(mP_\infty)$ is considered of polynomials in X and Y with coefficients in $\mathbf{F}_{16}$ of which solely the poles are situated in P_∞ and are of order less than or equal to m , where $0 \leq m \leq 65 = n + g - 1$. This vector space, which is of dimension greater than or equal to $(m - g + 1)$ (equal if $m > 2g - 2$), has a base constituted by the monomials $h_i = Y^t X^u$, where t is an integer between 0 and 3, u is a positive integer or zero, $4u + 5t \leq m$, and $i = 1, \dots, n - k$. This quantity $\rho(h_i) = 4u + 5t$ is usually called the "weight" of the monomial h_i .

Take for example: $m = 43$; a set of monomials h_i is then obtained where $i = 1, \dots, 38$, since $m - g + 1 = 43 - 6 + 1 = 38$.

The monomials h_i may be classified into ordered subsets of monomials

$$M_t = \{Y^t X^u \mid 0 \leq u \leq (43 - 5t) / 4\},$$

where: $0 \leq t \leq 3$. These ordered subsets of monomials are explicitly:

$$M_0 = \{1, X, X^2, \dots, X^{10}\},$$

$$M_1 = \{Y, YX, YX^2, \dots, YX^9\},$$

$$M_2 = \{Y^2, Y^2X, Y^2X^2, \dots, Y^2X^8\}, \text{ and}$$

$$M_3 = \{Y^3, Y^3X, Y^3X^2, \dots, Y^3X^7\}.$$

It is verified that the total number of monomials h_i is indeed equal to:
 $11+10+9+8 = 38$.

Finally, the parity-check matrix H of the code is defined in the following manner: the element in line i and column j of that matrix is equal to the value
 5 taken by the monomial h_i at the point P_j of the algebraic curve.

The redundancy $(n - k)$ of the code Q being equal to 38, its dimension is $k = 60 - 38 = 22$. The minimum distance d of this code is at least equal to $n - k + 1 - g = 33$.

Let the received word be the word $\underline{r}$ of weight 16, given by the
 10 following table where α is a primitive element of $\mathbf{F}_{16}$ satisfying $\alpha^4 + \alpha + 1 = 0$, x and y are the coordinates of the point concerned in the locating set, and $r(x, y)$ is the component associated with (i.e. labeled by) that point. At the points (x, y) of the curve not included in the table, the received symbol $r(x, y)$ has the value 0.

15	x	y	$r(x, y)$
	α	α^{13}	α^{12}
	α^2	α^3	α^{10}
	α^3	α	α^5
20	α^3	α^2	α^7
	α^4	α^{13}	1
	α^7	α^7	α^9
	α^7	α^9	α^9
	α^8	α^{11}	α^2
25	α^8	α^{14}	1
	α^8	α^3	α^2
	α^9	α	α^5
	α^9	α^8	α^{11}
	α^9	α^2	α^{11}
30	α^{11}	α^{12}	α^5
	α^{11}	α^{11}	α^5

$$\alpha^{14} \quad \alpha^{14} \quad \alpha^{10}$$

Use is then made of the "extended error syndromes" $\sigma_j(i) = \langle Y^j X^i | \underline{e} \rangle$, where $\underline{e}$ is the transmission error affecting the received word

5 $\underline{r}$, and $Y^j X^i$ represents the word of which the components are equal to the value taken by the monomial $Y^j X^i$ at the points of the locating set (the notation $\langle \underline{c}_1 | \underline{c}_2 \rangle$ represents the scalar product of any two words $\underline{c}_1$ and $\underline{c}_2$ of length n).

In passing, it will be noted that, due to the fact that $\alpha^{q-1} = 1$ for any non-zero element α of $\mathbf{F}_q$, the extended error syndromes $\sigma_j(i)$ are periodic in i , with a
10 period equal to $(q-1)$.

In the example set out here, given the received word $\underline{r}$, it is possible to calculate the 38 syndromes corresponding to the 38 lines of the parity-check matrix H . Let $\sigma_j(i)$ be the syndrome corresponding to the monomial $Y^j X^i$. These syndromes are directly obtained for the non-negative values of i and j
15 which satisfy $0 \leq i \leq (43 - 5j) / 4$, using the parity-check matrix H . For the other useful values of i and j , use is made of a method of calculating the syndromes that are referred to as "unknown". For the first unknown syndromes, this method implies majority decisions. For the other unknown syndromes, it may also be carried out by "majority" decisions which are then unanimous, or more
20 simply by using each time one of the polynomials of the Gröbner basis produced by the decoding algorithm used. It may be recalled here that this basis of polynomials generates the ideal of the locator polynomials. In this connection reference may be made to the article by Sakata *et al.* mentioned above.

25 For $0 \leq j \leq 3$, it is thus possible to calculate $\sigma_j(i)$ for $i=0, \dots, 14$. By giving the designation $\underline{\sigma}_j$ to the vector of the 15 components $\sigma_j(i)$ for $i=0, \dots, 14$, the four following vectors are obtained:

$$\begin{aligned} \underline{\sigma}_0 &= [\alpha^2 \alpha^8 \alpha^2 \alpha \alpha^{14} \alpha^{11} \alpha^{14} \alpha^2 \alpha^2 \alpha^{14} \quad 1 \alpha^{10} \alpha^5 \alpha^{14} \alpha^{14}], \\ 30 \quad \underline{\sigma}_1 &= [\alpha^{10} \alpha^5 \alpha^{12} \alpha \alpha^8 \alpha^{13} \alpha^8 \alpha^8 \alpha^{11} \alpha^{10} \alpha^{11} 1 \alpha^{10} \alpha^3 \alpha^{13}], \end{aligned}$$

25

$$\underline{\sigma}_2 = [\alpha \alpha^3 \alpha^{10} \alpha^{10} \alpha^5 \alpha^{12} \alpha^{13} \alpha^5 0 \alpha^{11} \alpha^{13} \alpha^{12} \alpha^5 \alpha^8 \alpha^{10}],$$

$$\underline{\sigma}_3 = [0 \alpha^{12} \alpha^{13} 1 \alpha^{14} \alpha^{12} \alpha^{12} \alpha^9 \alpha^{14} \alpha^{11} 0 \alpha^{12} 0 \alpha^6 \alpha^2].$$

On the basis of the syndromes so obtained, an error correction
 5 algorithm is implemented, for $j = 0, \dots, \lambda_{\max} - 1$ (i.e. for j going from 0 to 3 in the
 example described), that is adapted to Reed-Solomon codes, so as to calculate
 the errors $E_j(x)$ in the components of a word of Reed-Solomon code, defined
 over the same Galois field as said algebraic geometric code, the components
 being labeled by the $(q-1) = 15$ values of x each defining an aggregate. Such
 10 an error correction is for example performed by using the error syndromes
 polynomials:

$$S_j(Z) = \sum_{i=0}^{q-2} \sigma_j(i) Z^i.$$

It is noted that the error correction is performed on all the
 aggregates, without seeking to determine beforehand whether or not the
 15 aggregate concerned is an erroneous aggregate for example by means of an
 error locating algorithm.

In the above example, the matrix V is used, of dimensions
 15 x 15, of which the element in position (i, j) , $1 \leq i, j \leq 15$, is $\alpha^{(i-1)(j-1)}$
 where α is a root of $X^4 + X + 1$. This is because it is possible to define the 15-
 20 tuples $\underline{\tau}_j = \underline{\sigma}_j V^{-1}$ which are used profitably to calculate the estimated values $\hat{e}$
 $(x, y(x))$ of the errors. It may be noted that the inverse matrix V^{-1} of V is the
 matrix having $\alpha^{-(i-1)(j-1)}$ in position (i, j) . In the example treated the following is
 thus obtained:

$$\begin{aligned} \underline{\tau}_0 &= [0 \alpha^{12} \alpha^{10} \alpha^{13} 1 0 0 0 1 \alpha^5 0 0 0 0 \alpha^{10}], \\ \underline{\tau}_1 &= [0 \alpha^{10} \alpha^{13} \alpha^5 \alpha^{13} 0 0 \alpha^9 \alpha \alpha 0 \alpha^5 0 0 \alpha^9], \\ \underline{\tau}_2 &= [0 \alpha^8 \alpha \alpha^8 \alpha^{11} 0 0 \alpha^9 \alpha \alpha^8 0 \alpha^5 0 0 \alpha^8], \\ \underline{\tau}_3 &= [0 \alpha^6 \alpha^4 \alpha^3 \alpha^9 0 0 \alpha^{13} \alpha^{10} \alpha^{10} 0 \alpha^7 0 0 \alpha^7]. \end{aligned}$$

It will be noted that the calculation of $\underline{\tau}_j$ from $\underline{\sigma}_j$ may advantageously be performed using shift registers. Each column of V^{-1} is in fact a series of consecutive powers of the same element of $\mathbf{F}_q$. For all fixed h , the calculation of the component $\tau_j(h)$ of $\underline{\tau}_j$ is thus easily performed by the

5 circuit of **Figure 3**, where the element D represents a delay element. Advantageously, for all j , the $q - 1$ components $\tau_j(h)$ of $\underline{\tau}_j$ may then be calculated in parallel by circuits of the **Figure 3** type.

Thus let T be the 4×15 matrix of which the four lines are the four 15-tuples $\underline{\tau}_j$. The knowledge of this matrix T is equivalent to that of the 60

10 values of $E_j(\alpha^h)$ since giving the designation $\tau_j(h)$ to the element in position $(j, 1+h)$ of T gives:

$$\tau_j(h) = E_j(\alpha^h).$$

This leads to a simple calculation of the errors $\hat{e}(\alpha^h, y_p(\alpha^h))$ in the following manner. Let the designation t_h , $h = 0, \dots, 14$, be given to the $(h + 1)^{\text{th}}$

15 column of that matrix T , and for any such h , let there be constructed the Vandermonde matrix V_h of type 4×4 based on the four elements $y_p(\alpha^h)$, $p = 1, \dots, 4$. Here is thus the form of these matrices V_h :

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ y_1(\alpha^h) & y_2(\alpha^h) & y_3(\alpha^h) & y_4(\alpha^h) \\ [y_1(\alpha^h)]^2 & [y_2(\alpha^h)]^2 & [y_3(\alpha^h)]^2 & [y_4(\alpha^h)]^2 \\ [y_1(\alpha^h)]^3 & [y_2(\alpha^h)]^3 & [y_3(\alpha^h)]^3 & [y_4(\alpha^h)]^3 \end{bmatrix}.$$

20

The column vector

$$\hat{e}(\alpha^h) = [\hat{e}(\alpha^h, y_1(\alpha^h)), \hat{e}(\alpha^h, y_2(\alpha^h)), \hat{e}(\alpha^h, y_3(\alpha^h)), \hat{e}(\alpha^h, y_4(\alpha^h))]^T$$

(where the circumflex accent ^ indicates estimated errors and where T indicates the transposition) is then simply given by $\hat{e}(\alpha^h) = V_h^{-1} t_h$.

25

Preferably, it can be decided not to "materially" perform the calculation of the errors in the aggregate indexed by α^h when the "transformed" errors $E_j(\alpha^h)$ are all zero. This is because in this case it is certain in advance that the errors in the aggregate α^h will also be zero.

In summary, the four lines $\underline{\tau}_j$ of the matrix T are calculated by multiplying the four vectors of the syndromes $\underline{\sigma}_j$ (completed by the syndromes components referred to as "unknown") by V^{-1} and the errors are determined in the aggregate indexed by α^h by multiplying the corresponding column of T by the inverse of the Vandermonde matrix based on the four values $y_p(\alpha^h)$, $p = 1, \dots, 4$, of Y such that $(\alpha^h, y_p(\alpha^h))$ is a point on $\mathbf{F}_{16}$ of the curve $Y^4 + Y + X^5 = 0$.

It is noted that this determination of the errors $\hat{e}(\alpha^h, y_p(\alpha^h))$ is performed directly without having beforehand and explicitly located the aggregates containing at least one error.

The passage to a more general case does not give rise to difficulties. It is still possible to calculate a vectors $\underline{\sigma}_j$ of length $q - 1$ and the "transformed" components $E_j(x)$ of the errors are the components of the vector $\underline{\sigma}_j V^{-1}$ where V is the Vandermonde matrix based on the non-zero elements of $\mathbf{F}_q$. If for given x , all the $E_j(x)$ are zero, it is naturally considered that the components of the received word $\underline{r}$ in the aggregate of index x are correct. On the contrary, if at fixed x there is at least one component $E_j(x)$ which is non-zero, it is considered that the aggregate of index x is erroneous and it is sought to determine the errors affecting each component $r(x, y_p(x))$ of the received word in that aggregate.

For this purpose, calculation is made for each such value of x of the estimations $\hat{e}(x, y_p(x))$ of the respective errors in the components $r(x, y_p(x))$ of the received word by solving the equation system

$$E_j(x) = \sum_{p=1}^{\lambda(x)} [y_p(x)]^j \hat{e}(x, y_p(x))$$

in which equations have been chosen that are associated with consecutive values of j and are at least equal in number to the number $\lambda(x)$ of components of the aggregate indexed by x . This system contains a non-singular Vandermonde system of small size of which the vector solution formed by the components of the error in the aggregate indexed by $x = \alpha^h$ for α primitive in $\mathbf{F}_q$ is given by:

$$\begin{bmatrix} e(\alpha^h, y_1(\alpha^h)) \\ e(\alpha^h, y_2(\alpha^h)) \\ \vdots \\ e(\alpha^h, y_{\lambda(\alpha^h)}(\alpha^h)) \end{bmatrix} = \begin{bmatrix} 1 & \cdots & \cdots & 1 \\ y_1(\alpha^h) & \cdots & \cdots & y_{\lambda(\alpha^h)}(\alpha^h) \\ \vdots & \vdots & \vdots & \vdots \\ y_1^{\lambda(\alpha^h)-1}(\alpha^h) & \cdots & \cdots & y_{\lambda(\alpha^h)}^{\lambda(\alpha^h)-1}(\alpha^h) \end{bmatrix}^{-1} \begin{bmatrix} \tau_{0,h} \\ \tau_{1,h} \\ \vdots \\ \tau_{\lambda(\alpha^h)-1,h} \end{bmatrix} = \mathcal{G}^{-1} \begin{bmatrix} \tau_{0,h} \\ \tau_{1,h} \\ \vdots \\ \tau_{\lambda(\alpha^h)-1,h} \end{bmatrix}.$$

Returning to the example described here in detail, it may be noted that, for non-zero x in fixed $\mathbf{F}_{16}$, the four solutions of the equation $Y^4 + Y + x^5 = 0$ in Y are of four different types:

- 5 $[\alpha \ \alpha^2 \ \alpha^4 \ \alpha^8]$ for $x = 1, \alpha^3, \alpha^6, \alpha^9, \alpha^{12}$,
 $[\alpha^6 \ \alpha^7 \ \alpha^9 \ \alpha^{13}]$ for $x = \alpha, \alpha^4, \alpha^7, \alpha^{10}, \alpha^{13}$,
 $[\alpha^3 \ \alpha^{11} \ \alpha^{12} \ \alpha^{14}]$ for $x = \alpha^2, \alpha^5, \alpha^8, \alpha^{11}, \alpha^{14}$.

For example, the Vandermonde matrix $\mathcal{G}$ constructed on $[\alpha^6 \ \alpha^7 \ \alpha^9 \ \alpha^{13}]$ (usable for the aggregates corresponding to $x = \alpha, \alpha^4, \alpha^7, \alpha^{10}, \alpha^{13}$) is:

$$10 \quad \begin{bmatrix} 1 & 1 & 1 & 1 \\ \alpha^6 & \alpha^7 & \alpha^9 & \alpha^{13} \\ \alpha^{12} & \alpha^{14} & \alpha^3 & \alpha^{11} \\ \alpha^3 & \alpha^6 & \alpha^{12} & \alpha^9 \end{bmatrix}$$

and its inverse $\mathcal{G}^{-1}$ is:

$$\begin{bmatrix} \alpha^{14} & \alpha^{12} & \alpha^6 & 1 \\ \alpha^{13} & \alpha^{14} & \alpha^7 & 1 \\ \alpha^{11} & \alpha^3 & \alpha^9 & 1 \\ \alpha^7 & \alpha^{11} & \alpha^{13} & 1 \end{bmatrix}.$$

By multiplying this inverse matrix by the second column of the matrix $\mathcal{T}$ corresponding to $x = \alpha$, the vector $[0 \ 0 \ 0 \ \alpha^{12}]^T$ is obtained;

- 15 By multiplying this inverse matrix by the fifth column of the matrix $\mathcal{T}$ corresponding to $x = \alpha^4$, the vector $[0 \ 0 \ 0 \ 1]^T$ is obtained;

By multiplying this inverse matrix by the eighth column of the matrix $\mathcal{T}$ corresponding to $x = \alpha^7$, the vector $[0 \ \alpha^9 \ \alpha^9 \ 0]^T$ is obtained;

- By multiplying this inverse matrix by the eleventh column of the matrix $\mathcal{T}$ corresponding to $x = \alpha^{10}$, the vector $[0 \ 0 \ 0 \ 0]^T$ is obtained;

- 20 By multiplying this inverse matrix by the fourteenth column of the matrix $\mathcal{T}$ corresponding to $x = \alpha^{13}$, the vector $[0 \ 0 \ 0 \ 0]^T$ is obtained.

In the aggregate indexed by $x = \alpha$, there is thus a single non-zero error. It corresponds to $y = \alpha^8$ and the value of that error is α^{12} ;

in the aggregate indexed by $x = \alpha^4$, there is thus a single non-zero error. It corresponds to $y = \alpha^8$ and the value of that error is 1;

in the aggregate indexed by $x = \alpha^7$, there are thus two non-zero errors corresponding to $y = \alpha^2$ and to $y = \alpha^4$ and the value of both these errors is α^9 ;

in the aggregate indexed by $x = \alpha^{10}$, all the errors are zero;

in the aggregate indexed by $x = \alpha^{13}$, all the errors are zero.

When these estimated values of error are subtracted from the corresponding components of the received word, the zero symbol is obtained everywhere.

Similar calculations may be carried out for both the other classes of aggregates corresponding respectively to $x = 1, \alpha^3, \alpha^6, \alpha^9, \alpha^{12}$ and to $x = \alpha^2, \alpha^5, \alpha^8, \alpha^{11}, \alpha^{14}$. In all cases, by subtracting the estimated values of error from the corresponding components of the received word, the zero symbol is obtained everywhere. The decoded word is thus the zero word.

In the general case, the aggregates do not necessarily all have the same size. This is the case if the code, although constructed on a maximum curve of type $C(a, b)$, that is to say having aq points over $\mathbf{F}_q$, is shortened. This is also the case, more fundamentally, if for the curve of type $C(a, b)$ given by the equation $F(X, Y) = 0$, there are different values x_1 and x_2 in $\mathbf{F}_q$, for which the number of solutions at Y (in $\mathbf{F}_q$) of $F(x_1, Y) = 0$ and of $F(x_2, Y) = 0$ is different.

For example, with $a = 4$, if there are available the four polynomials

$$S_j(Z) = \sum_{i=0}^{q-2} \sigma_j(i) Z^i, \quad j = 0, \dots, 3, \text{ from which can be calculated the four}$$

vectors $\underline{\tau}_j = \underline{\sigma}_j V^{-1}$, where $\underline{\sigma}_j$ is the vector of the $\sigma_j(i)$. In equivalent manner, $\underline{\tau}_j$ is the vector having in position h , $0 \leq h \leq q-2$, the value of the polynomial $S_j(Z)$ evaluated in $Z = \alpha^{-h}$ with α primitive in $\mathbf{F}_q$.

If for example the case is taken in which the aggregate corresponding to $X = x$ only contains three coordinates denoted $(x, y_1(x))$, $(x, y_2(x))$ and $(x, y_3(x))$, there are four equations relating to the three unknowns $\hat{e}(x, y_s(x))$, $s = 1, 2, 3$:

$$\hat{e}(x, y_1(x)) + \hat{e}(x, y_2(x)) + \hat{e}(x, y_3(x)) = S_0(x^{-1}) = \tau_0(x),$$

$$y_1(x) \hat{e}(x, y_1(x)) + y_2(x) \hat{e}(x, y_2(x)) + y_3(x) \hat{e}(x, y_3(x)) = S_1(x^{-1}) = \tau_1(x),$$

$$(y_1(x))^2 \hat{e}(x, y_1(x)) + (y_2(x))^2 \hat{e}(x, y_2(x)) + (y_3(x))^2 \hat{e}(x, y_3(x)) = S_2(x^{-1}) = \tau_2(x),$$

$$(y_1(x))^3 \hat{e}(x, y_1(x)) + (y_2(x))^3 \hat{e}(x, y_2(x)) + (y_3(x))^3 \hat{e}(x, y_3(x)) = S_3(x^{-1}) = \tau_3(x),$$

5 that are written as

$$\begin{bmatrix} \tau_0 \\ \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ y_1 & y_2 & y_3 \\ y_1^2 & y_2^2 & y_3^2 \\ y_1^3 & y_2^3 & y_3^3 \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix}$$

where the arguments x and $y_i(x)$ have been omitted from the elements of the matrix and of the two vectors. In such a case, if no $y_i(x)$ is zero, it is possible to choose either the first three equations or the last three, from the four equations given above in matrix form. For example, if the last three are taken, calculation will be made of the three components e_i of the error in the aggregate indexed by x by solving the system

$$\begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix} = \begin{bmatrix} y_1 & y_2 & y_3 \\ y_1^2 & y_2^2 & y_3^2 \\ y_1^3 & y_2^3 & y_3^3 \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix}.$$

The first of the four equations is then usable to verify the consistency of the values e_i so calculated.

The block diagram of **Figure 2** represents an apparatus for reading encoded digital signals 70, incorporating the decoder 10. This apparatus 70 comprises a keyboard 711, a screen 709, an external recipient of information 109, a data reader 105 and a demodulator 106, conjointly connected to input/output ports 703 of the decoder 10 which is produced here in the form of a logic unit.

The decoder 10 comprises, connected together by an address and data bus 702:

- a central processing unit 700,
- 25 - a random access memory (RAM) 704,
- a read only memory (ROM) 705; and
- said input/output ports 703.

Each of the elements illustrated in Figure 2 is well known to the person skilled in the art of microcomputers and mass storage systems and, more generally, of information processing systems. These known elements are therefore not described here. It should be noted, however, that:

- 5 - the information recipient 109 could be, for example, an interface peripheral, a display, a modulator, an external memory or another information processing system (not shown), and could be adapted to receive sequences of signals representing speech, service messages or multimedia data in particular of the IP or ATM type, in the form of sequences of binary data,
- 10 - the reader 105 is adapted to read data recorded on a carrier such as a magnetic or magneto-optical disk.

The random access memory 704 stores data, variables and intermediate processing results, in memory registers bearing, in the description, the same names as the data whose values they store. The random access

15 memory 704 contains in particular the following registers:

- registers "*received_words*", in which the received words are kept,
- a register "*estimated_symbols*", in which are stored the symbols from a received word in course of correction,
- a register "*associated_words*", in which are stored the symbols of the
- 20 "*associated codewords*", and
- a register "*information_symbols*", in which are stored the symbols resulting from the redundancy removal.

The read only memory 705 is adapted to store, in registers which, for convenience, have the same names as the data which they store:

- 25 - the operating program of the central processing unit 700, in a register "*program*",
- the length of each codeword in a register "*n*",
- the number of information symbols in each code word, in a register "*k*", and
- 30 - a table containing each word $\underline{Y^j X^i}$ of which the components are equal to the value taken by the monomial $Y^j X^i$ at the points of the locating set,

for $i = 0, \dots, 2L - 1$ and $j = 0, \dots, \Lambda_{\max} - 1$, where L is the total number of aggregates and $\Lambda_{\max}$ is the size of the maximum aggregate, in a register "W".

An application of the invention to the mass storage of data has been described above by way of example, but it is clear that the methods according
5 to the invention may equally well be implemented within a telecommunications network, in which case unit 105 could for example be a receiver adapted to implement a protocol for data packet transmission over a radio channel.

CLAIMS

1. A method of decoding a one-point algebraic geometric code
 5 defined on an algebraic curve of type $C(a,b)$ represented by an equation
 $F(X,Y)=0$ of degree b in X and of degree a in Y over $\mathbf{F}_q$, characterized in
 that it comprises the following steps:

- calculating extended error syndromes $(\sigma_j(i))$ associated with a received
 word $(\underline{r})$;
- 10 - determining the values of errors in each component $r(x, y_p(x))$ of the
 received word $\underline{r}$, on the basis of the extended error syndromes calculated.

2. A decoding method according to claim 1, characterized in that, the
 algebraic curve being partitioned into aggregates, the step of determining the
 15 values of errors in each component on the basis of the extended error
 syndromes comprises the following steps for each aggregate:

- calculating, on the basis of the extended error syndromes,
 compound error values $(\tau_j(h), E_j(x))$ each representing a linear combination of
 the errors in the components of the aggregate;
- 20 - determining that the errors in the components of the aggregate are
 zero if all said compound error values are zero;
- calculating the errors in the components of the aggregate if at least
 one of said compound error values is not zero.

25 3. A decoding method according to claim 2, characterized in that the
 calculation of the extended error syndromes $\sigma_j(i)$ is performed for
 $j=0, \dots, j_0-1$, where j_0 is at least equal to the maximum value $\lambda_{\max}$ of the
 cardinals $\lambda(x)$ of the aggregates of the code.

4. A decoding method according to one of claims 1 to 3, characterized in that determining the values of errors comprises the following steps:

- calculating $\underline{\tau}_j = \underline{\sigma}_j \cdot V^{-1}$ for $j = 0, \dots, a - 1$, where V^{-1} is the matrix of dimensions $(q - 1) \times (q - 1)$ of which the element in position (i, j) , $1 \leq i, j \leq q - 1$, is $\alpha^{-(i-1)(j-1)}$, α being a primitive element of $\mathbf{F}_q$;

- for all h such that there is j such that $\tau_j(h) \neq 0$, calculating the error $\hat{e}(x, y_p(x))$ of the components of the aggregate defined by $x = \alpha^h$ in the received word by:

$$10 \quad \begin{bmatrix} \hat{e}(\alpha^h, y_1(\alpha^h)) \\ \hat{e}(\alpha^h, y_2(\alpha^h)) \\ \vdots \\ \hat{e}(\alpha^h, y_{\lambda(\alpha^h)}(\alpha^h)) \end{bmatrix} = \begin{bmatrix} 1 & \dots & \dots & 1 \\ y_1(\alpha^h) & \dots & \dots & y_{\lambda(\alpha^h)}(\alpha^h) \\ \vdots & \vdots & \vdots & \vdots \\ y_1^{\lambda(\alpha^h)-1}(\alpha^h) & \dots & \dots & y_{\lambda(\alpha^h)}^{\lambda(\alpha^h)-1}(\alpha^h) \end{bmatrix}^{-1} \begin{bmatrix} \tau_0(h) \\ \tau_1(h) \\ \vdots \\ \tau_{\lambda(\alpha^h)-1}(h) \end{bmatrix}.$$

5. A decoding method according to claim 4, characterized in that, for each j , the calculation of the elements of the vector $\underline{\tau}_j$ is performed by a circuit comprising a shift register and receiving as input the elements of the vector $\underline{\sigma}_j$.

15

6. A decoding method according to one of claims 1 to 3, characterized in that determining the values of errors comprises the following steps:

- implementing, for $j = 0, \dots, \lambda_{\max} - 1$, by means of the error syndromes polynomial

20

$$S_j(Z) = \sum_{i=0}^{q-2} \sigma_j(i) Z^i,$$

an error correction algorithm adapted to Reed-Solomon codes, so as to calculate the error $E_j(x)$ in each component labeled by the element x of $\mathbf{F}_q$ of a word of Reed-Solomon code defined over $\mathbf{F}_q$, and

25

- calculating the estimations $\hat{e}(x, y_p(x))$ of the respective errors in the components $r(x, y_p(x))$ of $\underline{r}$ by solving the system of equations of type

$$E_j(x) = \sum_{p=1}^{\lambda(x)} [y_p(x)]^j \hat{e}(x, y_p(x)) , \quad (3)$$

where equations are chosen that are associated with consecutive values of j at least equal in number to the number of components $\lambda(x)$ of the aggregate considered.

5

7. A decoding method according to claim 6, characterized in that the calculation of the estimations $\hat{e}(x, y_p(x))$ is performed only for the values of x such that there is at least one value of j for which $E_j(x)$ is non-zero.

10

8. A decoding method according to any one of claims 4 to 7, characterized in that, if there is at least one value of $x = \alpha^h$ associated with an erroneous aggregate of cardinal $\lambda(x) < a$, comparison is then made between the members of at least one equation

$$E_j(x) = \sum_{p=1}^{\lambda(x)} [y_p(x)]^j \hat{e}(x, y_p(x)) , \quad \text{where } \lambda(x) \leq j \leq a-1 ,$$

15

associated with said value of x , if need be after having calculated $E_j(x)$ by means of an error correction algorithm adapted to Reed-Solomon codes.

20

9. An error correction device for a one-point algebraic geometric code defined on an algebraic curve of type $C(a, b)$ represented by an equation $F(X, Y) = 0$ of degree b in X and of degree a in Y over $\mathbf{F}_q$, characterized in that it comprises:

- means for calculating extended error syndromes $(\sigma_j(i))$ associated with a received word $(\underline{r})$;

25

- means for determining the values of errors in each component $(r(x, y_p(x)))$ of the received word $\underline{r}$, on the basis of the extended error syndromes calculated.

10. An error correction device according to claim 9, characterized in that, the algebraic curve being partitioned into aggregates, the means for determining the values of errors in each component on the basis of the extended error syndromes comprise:

5 - means for calculating, for each aggregate and on the basis of the extended error syndromes, compound error values $(\tau_j(h), E_j(x))$ each representing a linear combination of the errors in the components of the aggregate;

10 - means for determining that the errors in the components of an aggregate are zero if all said compound error values relative to that aggregate are zero;

 - means for calculating the errors in the components of an aggregate if at least one of said compound error values relative to that aggregate is not zero.

15

11. An error correction device according to claim 10, characterized in that the means for calculating the extended error syndromes $\sigma_j(i)$ are implemented for $j=0, \dots, j_0-1$, where j_0 is at least equal to the maximum value $\lambda_{\max}$ of the cardinals $\lambda(x)$ of the aggregates of the code.

20

12. An error correction device according to one of claims 9 to 11, characterized in that the means for determining the values of errors comprise:

25 - means for calculating $\underline{\tau}_j = \underline{\sigma}_j \cdot V^{-1}$ for $j = 0, \dots, a-1$, where V^{-1} is the matrix of dimensions $(q-1) \times (q-1)$ of which the element in position (i, j) , $1 \leq i, j \leq q-1$, is $\alpha^{-(i-1)(j-1)}$, α being the primitive element of $\mathbf{F}_q$;

 - means for calculating, for all h such that there is j such that $\tau_j(h) \neq 0$, the error $\hat{e}(x, y_p(x))$ of the components of the aggregate defined by $x = \alpha^h$ in the received word $\underline{r}$ by:

$$\begin{bmatrix} \hat{e}(\alpha^h, y_1(\alpha^h)) \\ \hat{e}(\alpha^h, y_2(\alpha^h)) \\ \vdots \\ \hat{e}(\alpha^h, y_{\lambda(\alpha^h)}(\alpha^h)) \end{bmatrix} = \begin{bmatrix} 1 & \cdots & \cdots & 1 \\ y_1(\alpha^h) & \cdots & \cdots & y_{\lambda(\alpha^h)}(\alpha^h) \\ \vdots & \vdots & \vdots & \vdots \\ y_1^{\lambda(\alpha^h)-1}(\alpha^h) & \cdots & \cdots & y_{\lambda(\alpha^h)}^{\lambda(\alpha^h)-1}(\alpha^h) \end{bmatrix}^{-1} \begin{bmatrix} \tau_0(h) \\ \tau_1(h) \\ \vdots \\ \tau_{\lambda(\alpha^h)-1}(h) \end{bmatrix}$$

13. An error correction device according to claim 12, characterized in that, for each j , it comprises a circuit comprising a shift register and receiving as
5 input the elements of the vector $\underline{\sigma}_j$ to calculate the elements of the vector $\underline{\tau}_j$.

14. An error correction device according to one of claims 9 to 11, characterized in that the means for determining the values of errors comprise:

- means for implementing, for $j = 0, \dots, \lambda_{\max} - 1$, by means of the error
10 syndromes polynomial

$$S_j(Z) = \sum_{i=0}^{q-2} \sigma_j(i) Z^i ,$$

an error correction algorithm adapted to Reed-Solomon codes, so as to calculate the error $E_j(x)$ in each component labeled by the element x of $\mathbf{F}_q$ of a word of Reed-Solomon code defined over $\mathbf{F}_q$, and

- means for calculating the estimations $\hat{e}(x, y_p(x))$ of the respective errors
15 in the components $r(x, y_p(x))$ of $\underline{r}$ by solving the system of equations of type

$$E_j(x) = \sum_{p=1}^{\lambda(x)} [y_p(x)]^j \hat{e}(x, y_p(x)) , \quad (3)$$

where equations are chosen that are associated with consecutive values of j at least equal in number to the number of components $\lambda(x)$ of the aggregate
20 considered.

15. An error correction device according to claim 14, characterized in that the means for calculating the estimations $\hat{e}(x, y_p(x))$ are implemented only for the values of x such that there is at least one value of j for which $E_j(x)$ is
25 non-zero.

16. An error correction device according to any one of claims 12 to 14, characterized by means for comparing, if there is at least one value of $x = \alpha^h$ associated with an erroneous aggregate of cardinal $\lambda(x) < a$, between the members of at least one equation

5
$$E_j(x) = \sum_{p=1}^{\lambda(x)} [y_p(x)]^j \hat{e}(x, y_p(x)) , \text{ where } \lambda(x) \leq j \leq a-1 ,$$

associated with said value of x .

17. A decoder (10), characterized in that it comprises:

- at least one error correction device according to any one of claims 9 to 10 16, and
- at least one redundancy removal unit (108).

18. An apparatus for receiving encoded digital signals (70), characterized in that it comprises a decoder according to claim 17, and in that it 15 comprises means (106) for demodulating said encoded digital signals.

19. A computer system (70), characterized in that it comprises a decoder according to claim 17, and in that it further comprises:

- at least one hard disk, and
- 20 - at least one means (105) for reading that hard disk.

20. Non-removable data storage means, characterized in that it comprises computer program code instructions for the execution of the steps of a method according to any one of claims 1 to 8.

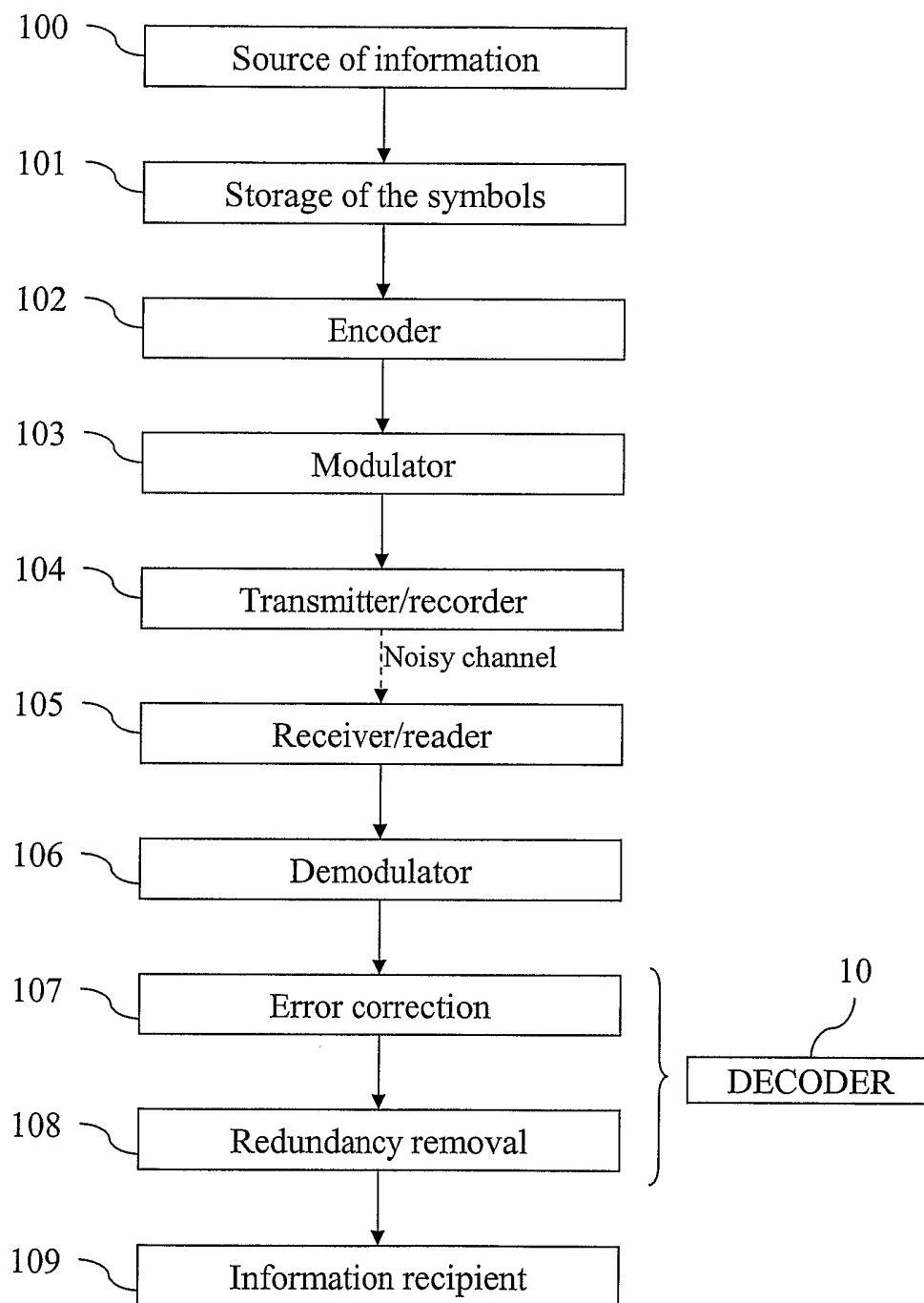
25

21. Partially or wholly removable data storage means, characterized in that it comprises computer program code instructions for the execution of the steps of a method according to any one of claims 1 to 8.

30 22. A computer program, characterized in that it contains instructions such that, when said program controls a programmable data processing device,

said instructions cause said data processing device to implement a method according to any one of claims 1 to 8.

1/3

**FIG. 1**

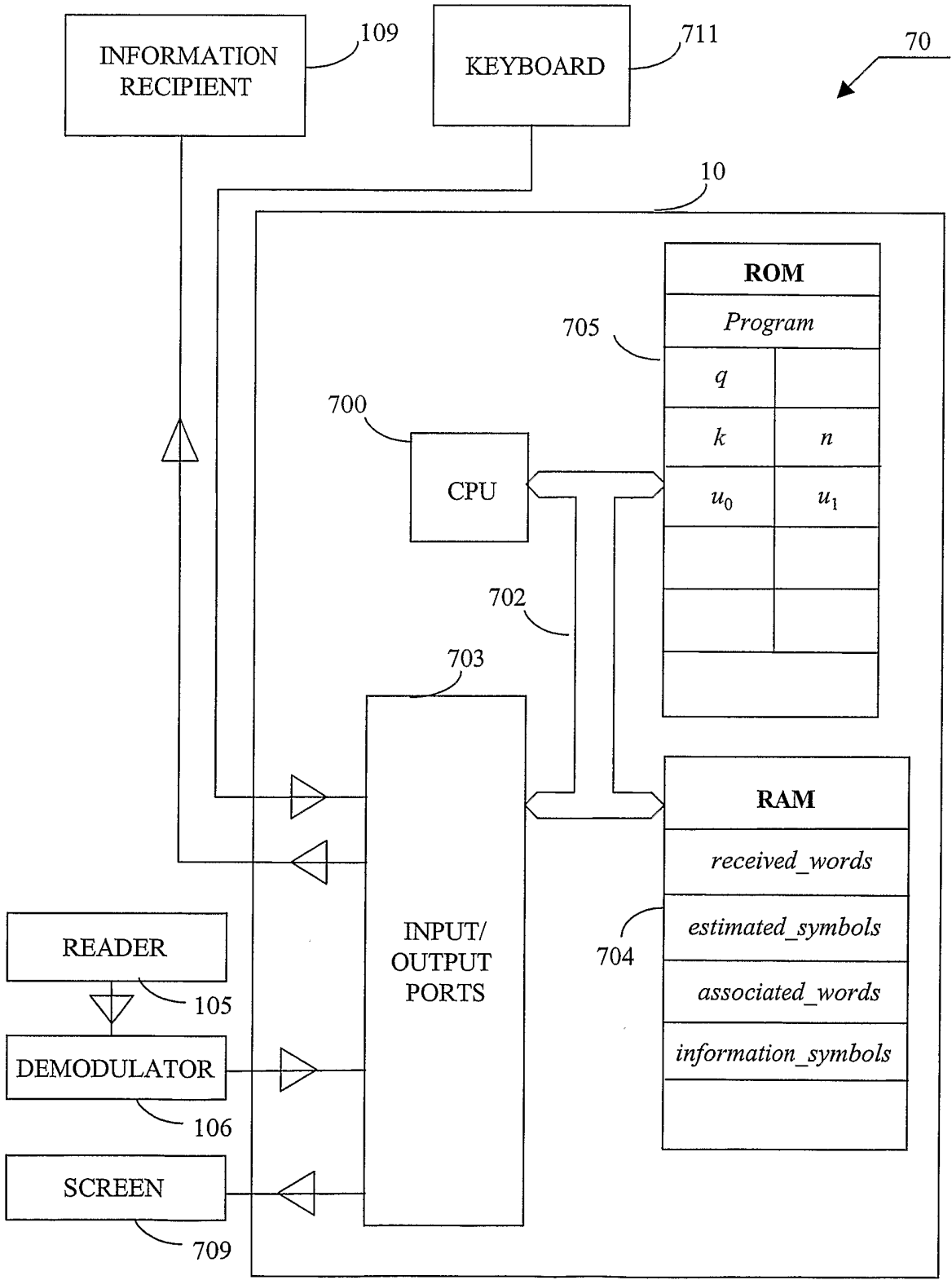
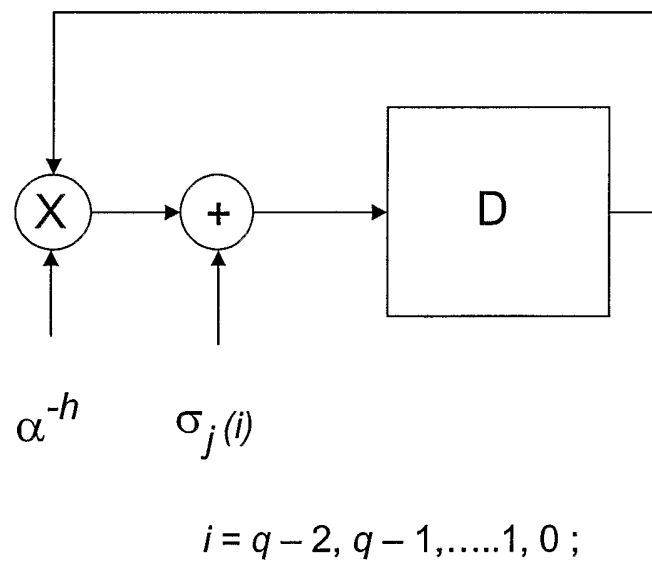


FIG. 2

3/3

**FIG. 3**

INTERNATIONAL SEARCH REPORT

International application No
PCT/IB2005/004034

A. CLASSIFICATION OF SUBJECT MATTER
INV. H03M13/13

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
H03M

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

EPO-Internal

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2004/117719 A1 (LEHOBEY FREDERIC ET AL) 17 June 2004 (2004-06-17) claim 1 -----	1,9

☐ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* Special categories of cited documents :

- *A* document defining the general state of the art which is not considered to be of particular relevance
- *E* earlier document but published on or after the international filing date
- *L* document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
- *O* document referring to an oral disclosure, use, exhibition or other means
- *P* document published prior to the international filing date but later than the priority date claimed

- *T* later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
- *X* document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
- *Y* document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.
- *Z* document member of the same patent family

Date of the actual completion of the international search

19 April 2006

Date of mailing of the international search report

03/05/2006

Name and mailing address of the ISA/
European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,
Fax: (+31-70) 340-3016

Authorized officer

Farman, T

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/IB2005/004034

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2004117719 A1	17-06-2004	FR 2845220 A1	02-04-2004