STEP-BY-STEP DECODING OF ALTERNANT CODES

C. G. Omidyar and H. J. Helgert
Department of Electrical Engineering and Computer Science
The George Washington University
Washington, D. C. 20052 U.S.A.

SUMMARY

In this paper we present a decoding scheme for Alternant codes. The syndromes are calculated from the received vector and the parity check matrix H. Let t be the error correcting capability of the decoder. Then we determine a Key Equation by adding t columns of the parity check matrix H. We raise this equation t-1 times to the power of n, where n is the number of columns of H.

Next we consider a matrix A_t whose elements are the set of coefficients from the Key Equations which we obtained. We make a decision based on the determinant of the matrix A_t. If the matrix A_t is singular, then we test the matrix A_{t-1} for singularity and continue up to A_{t-t+1} which in fact the decoder can correct one error. if any one of the matrices A_t through A_{t-t+1} is nonsingular we change the first digit of the received vector, then recompute the syndromes and recheck Δ_t'. If Δ_t' is zero the change is retained. If not, the digit is changed again. The Algorithm then proceeds to the next digit.

This Algorithm for decoding Alternant codes has significant improvements over previous schemes since the step-by-step decoding can be carried out at selected areas of the received word.

INTRODUCTION

DESCRIPTION OF ALTERNANT CODES

A linear (n, k) code is described by its parity check matrix H whose entries are elements from GF(q), where the integer n is the length of the code and k is the number of information symbols. The nonprimitive Alternant codes [1] are defined by the parity check matrix

$$H = \begin{bmatrix} z_1/(x_1-\omega) & z_2/(x_2-\omega) & \cdots\cdots\cdots\cdots & z_n/(x_n-\omega) \\ z_1/(x_1-\omega)^2 & \cdots\cdots\cdots\cdots\cdots & & z_n/(x_n-\omega)^2 \\ \vdots & & & \vdots \\ z_1/(x_1-\omega)^r & \cdots\cdots\cdots\cdots\cdots & & z_n/(x_n-\omega)^r \end{bmatrix}$$

Let $m=\mu\lambda$, where μ and λ are integers greater than one and $n \leq q^{\lambda}$, each x_i is a different element of $GF(q^{\lambda})$, $z_i \epsilon GF(q^{\lambda})$-0 and ω is any element of $GF(q^m)$ not in a proper subfield of $GF(q^m)$. The syndromes are calculated from the received vector and the parity check matrix H by $S=VH^t=eH^t$, where e is the error vector.

SPECIAL CASE

Let q=2, m=20, μ=4, λ=5, z_i =1 for i=1, 2,....32 and n=32. ω is a primitive element of $GF(2^{20})$ which is a root of $x^{20}+x^3+1$. The parity check matrix is

$$H = \begin{bmatrix}
0&0&1&0&1&0&0&0&0&0&1&1&1&1&1&1&1&1&0&0&1&1&1&1&1&0&1&1&0&1 \\
0&0&0&0&1&1&0&0&1&1&0&0&0&0&0&1&0&1&1&1&0&0&1&0&0&1&1&0&0&0&1&1 \\
1&0&0&1&1&0&0&1&0&0&0&1&1&1&1&0&0&1&1&1&0&1&0&1&0&1&1&0&0&1&0&1 \\
0&1&1&1&1&1&0&1&1&1&0&0&0&1&1&1&1&1&1&1&1&0&1&0&0&1&0&1&0&0&0&0 \\
0&1&0&0&0&1&0&0&1&1&1&0&1&0&1&0&1&1&0&1&1&0&1&0&1&0&1&1&0&0&1&0 \\
0&1&0&0&0&1&0&0&1&0&0&1&1&0&1&1&1&0&1&0&0&1&0&1&1&0&0&0&1&0&1&1 \\
0&1&0&0&0&0&1&1&1&1&0&0&1&1&1&1&1&1&0&1&1&0&0&0&1&1&1&1&1&0&1&0 \\
0&1&1&0&0&0&0&0&0&0&1&1&1&1&0&1&1&1&1&1&0&1&1&1&1&1&1&0&1&1&1&0 \\
0&1&0&0&0&0&1&0&0&1&1&1&0&0&1&0&1&0&1&0&1&0&0&0&1&0&0&0&1&1&1&0 \\
0&1&0&0&1&1&0&1&1&0&0&0&1&0&0&1&0&0&1&1&0&0&1&0&0&0&1&1&0&1&0&1 \\
0&1&0&0&1&1&0&1&0&1&0&1&0&1&1&0&1&0&0&1&0&0&1&0&0&1&0&0&1&1&0&1 \\
0&1&0&0&0&1&0&0&1&1&1&0&0&0&0&1&1&1&1&0&1&0&0&0&1&1&0&1&0&0&0&0 \\
0&1&1&1&1&1&0&0&1&0&1&0&0&1&0&0&0&1&0&0&1&1&0&1&0&1&1&1&0&0&0&1 \\
0&1&0&1&0&1&0&0&0&0&1&1&1&0&1&0&0&0&0&0&0&1&0&0&1&0&1&0&0&0&0&0 \\
0&1&0&1&1&0&0&1&1&0&0&0&1&1&1&1&0&0&0&0&1&0&1&1&0&1&0&0&0&0&0&1 \\
0&1&1&1&0&1&1&1&1&1&0&1&0&0&1&0&0&0&0&0&1&0&1&0&0&0&1&1&1&1&1&0 \\
0&1&1&0&0&1&0&1&0&1&1&1&1&1&0&1&0&1&1&1&1&0&1&1&1&0&0&1&0&1&1&1 \\
0&1&1&0&0&1&0&0&1&0&0&1&1&1&1&0&0&0&0&0&1&1&1&1&1&1&1&0&1&0&1&1 \\
0&1&1&0&0&0&0&0&0&1&1&1&1&0&1&1&0&0&0&0&0&0&1&0&0&0&1&0&1&1&0&0 \\
1&1&0&1&1&1&0&0&1&1&0&1&1&1&0&0&1&0&1&0&0&1&0&1&0&1&0&0&1&1&1&1
\end{bmatrix}$$

H has 32 columns, exactly n-k=20 linearly independent rows and the total number of code words is 2^{12} =4096.

WEIGHT SPECTRUM OF ALTERNANT CODES

Let β_i denote the number of code words of weight i in a linear (n, k) code. The numbers β_i, i = l, 2, ...n, form the weight spectrum of the code. Table 1 shows the weight spectrum of the above code.

TABLE 1

i =	9	10	11	12	13	14	15	16	17	18	19	20	21	22
β_i =	44	78	104	202	312	463	584	363	328	457	312	210	136	57
i =	23	24	25	26										
β_i =	24	14	4	1										

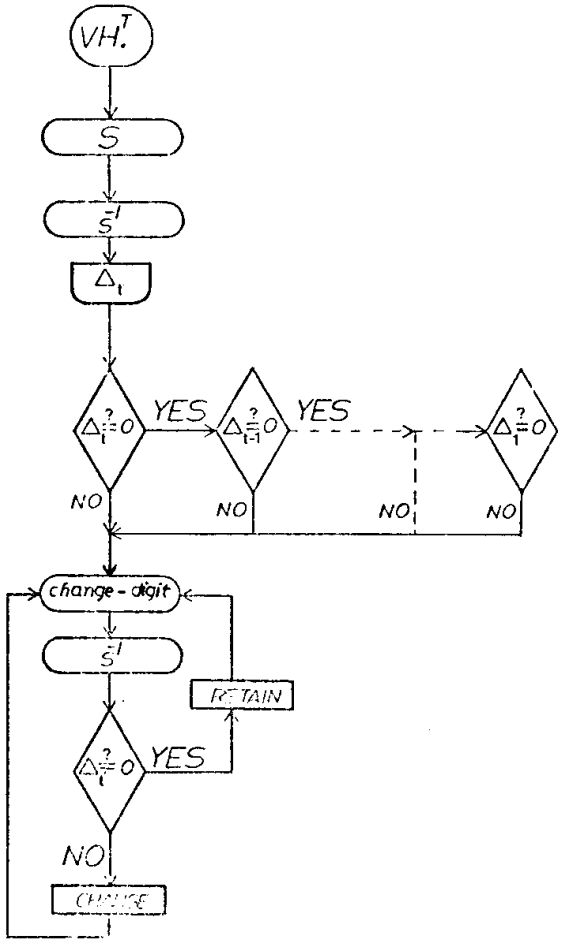
The operation of error correction can be described as follows. (see flow chart)

STEP 1 Test Δ_t through Δ_1; if any Δ_t, $\neq 0$ go to step 2.

STEP 2 Change first digit by adding the error vector e=1, 0, 0 ...
Recompute s-1 and set $c=w+s^{-1}$

STEP 3 Test $\Delta_{t'}$, if $\Delta_{t'}$ =0 the change is retained. If not the digit is changed back.

STEP 4 Transfer the control to step 2 for the next digit.



FLOWCHART OF DECODING ALGORITHM

DESCRIPTION OF DECODING ALGORITHM

The Key Equation for decoding Alternant codes is derived by adding t columns of the parity check matrix H. An additional t-1 equations are required to correct t errors. We raise the Key Equation t-1 times to the power of n, where n is the number of columns of H.

In general, the columns of H are designated by $1/(x_i-\omega)$ i=1, 2 ... n. Therefore we set $S=\sum_{i=1}^{t} 1/(x_i-\omega)$, where t is the number of errors on the received vector and S is the syndrome vector.

In order to decode the received sequence, the decoder has to make its decision based on the following determinants. The Algorithm for decoding Alternant codes and its design equations are as follows. Let $c=w+s^{-1}$.

ERROR-CORRECTING

1 Error-Correcting

$$S=\frac{1}{(x_1+w)},\ \partial_1=x_1 \longrightarrow \partial_1=c$$

2 Error-Correcting

$$S=\sum_{i=1}^{\tilde{}} \frac{1}{(x_i+w)},\ \text{where } \partial_1=\sum_{i=1}^{2} x_i,\ \partial_2=\prod_{i=1}^{2} x_i \rightarrow \begin{bmatrix} c & 1 \\ c^n & 1 \end{bmatrix} \begin{bmatrix} \partial_1 \\ \partial_2 \end{bmatrix} = \begin{bmatrix} w^2 \\ w^{2n} \end{bmatrix}$$

3 Error Correcting

$$S=\sum_{i=1}^{3} \frac{1}{(x_i+w)},\ \text{where } \partial_1=\sum_{i=1}^{3} x_i,\ \partial_2=\sum_{\substack{i,j=1 \\ i\neq j}}^{2} x_i x_j,\ \partial_3=\prod_{i=1}^{3} x_i \rightarrow$$

$$\begin{bmatrix} w^2 & c & 1 \\ w^{2n} & c^n & 1 \\ w^{2n^2} & c^{n^2} & 1 \end{bmatrix} \begin{bmatrix} \partial_1 \\ \partial_2 \\ \partial_3 \end{bmatrix} = \begin{bmatrix} w^3 \\ w^{3n} \\ w^{3n^2} \end{bmatrix}$$

4 Error-Correcting

$$S=\sum_{i=1}^{4} \frac{1}{(x_i+w)}, \text{ where } \partial_1=\sum_{i=1}^{4} x_i,\ \partial_2=\sum_{\substack{i,j=1\\ i\neq j}}^{3} x_i x_j,\ \partial_3=\sum_{\substack{i,j,k=1\\ i\neq j\neq k}}^{3} x_i x_j x_k,\ \partial_4=\prod_{i=1}^{4} x_i$$

$$\rightarrow \begin{bmatrix} w^2c & w^2 & c & 1 \\ w^{2n}c^n & w^{2n} & c^n & 1 \\ w^{2n}c^{n^2} & w^{2n^2} & c^{n^2} & 1 \\ w^{2n^3}c^{n^3} & w^{2n^3} & c^{n^3} & 1 \end{bmatrix} \begin{bmatrix} \partial_1 \\ \partial_2 \\ \partial_3 \\ \partial_4 \end{bmatrix} = \begin{bmatrix} w^4 \\ w^{4n} \\ w^{4n^2} \\ w^{4n^3} \end{bmatrix}$$

APPENDIX A

The computation of the determinants is carried out on an IBM 370 computer. As an example for 4 error-correcting the algebraic solution is as follows. (Special case)

$$\Delta_{t=4} = w^2c\,[c^n(w^{2n^2}+w^{2n^3})+c^{n^2}(w^{2n}+w^{2n^3})+c^{n^3}(w^{2n}+w^{2n^2})] +$$
$$w^{2n}c^n[c(w^{2n^2}+w^{2n^3})+c^{n^2}(w^2+w^{2n^3})+c^{n^3}(w^2+w^{2n^2})] +$$
$$w^{2n^2}c^{n^2}[c(w^{2n^3}+w^{2n})+c^n(w^2+w^{2n^3})+c^{n^3}(w^2+w^{2n})] +$$
$$w^{2n^3}c^{n^3}[c(w^{2n}+w^{2n^2})+c^n(w^2+w^{2n^2})+c^{n^2}(w^2+w^{2n})]$$

(← nth Power; ← nth Power; nth Power)

We compute the first row and raise it to the power of n. Each row is the nth power of the previous row. For brevity, the coefficients are not reported here.

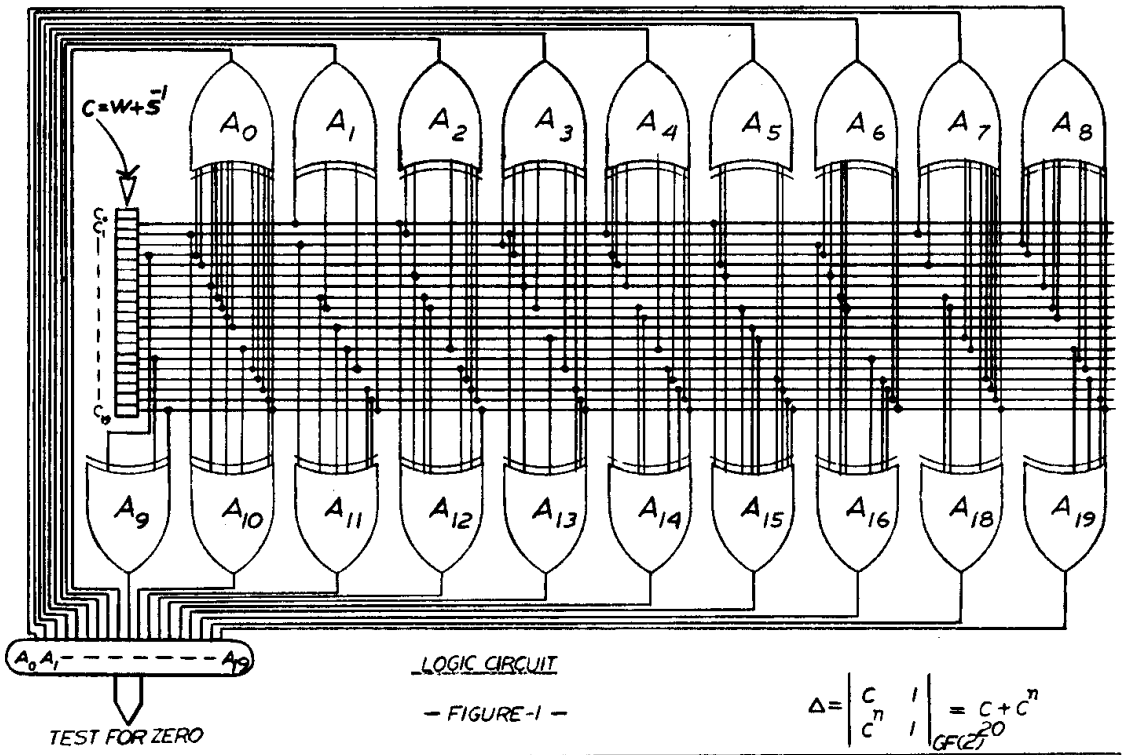
The decoder connections are shown on logic circuit cards for 2 and 3 error-correcting. (See Figure 1 and 2)

REFERENCES

[1] H. J. Helgert, “Alternant Codes,” Information and control, vol. 26, December 1974.

[2] H. J. Helgert, “Decoding of Alternant Codes,” IEEE. Vol. IT-23, July 1977.

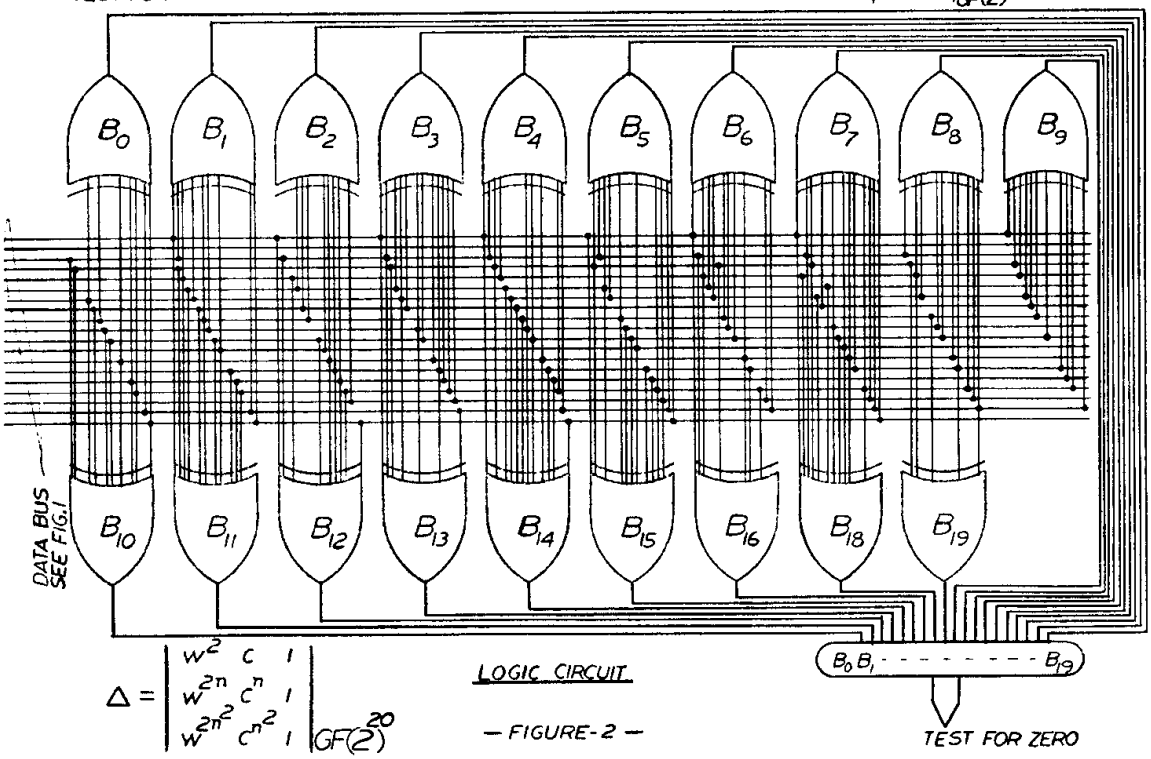
[3] E. R. Berlekamp, “Algebraic Coding Theory,” McGraw-Hill, New York, 1968



LOGIC CIRCUIT

— FIGURE-1 —

$$\Delta = \begin{vmatrix} c & 1 \\ c^n & 1 \end{vmatrix}_{GF(2^{20})} = c + c^n$$



$$\Delta = \begin{vmatrix} w^2 & c & 1 \\ w^{2n} & c^n & 1 \\ w^{2n^2} & c^{n^2} & 1 \end{vmatrix}_{GF(2^{20})}$$

LOGIC CIRCUIT

— FIGURE-2 —