한국통신학회 1983년도 추계학술발표회 논문집 (19983· 10· 29)

콘볼루션 코드를 이용한 에러 정정 디코더 설계

83328

○최 상 일 강 창 언
연 세 대 학 교

Design of an Error Correcting Decoder Using Convolutional Codes

○Sang Il CHOI Chang Eon KANG
YON SEI UNIVERSITY

Abstract.

In this paper, a general procedure is formulated for decoding any convolutional code with the decoding delay m blocks that corrects all bursts confined to m-1 or fewer consecutive blocks followed by a guard space of at least m-1 consecutive error free blocks. It is shown that this procedure can be implemented by a simple logical circuitry of the same order of complexity as a parity checking circuit for a block linear code.

Using the Type-B_2 codes analysis, the error correcting decoder for (18,12) convolutional code with double errors between messages is physically designed.

1. Introduction

The convolutional codes for correcting burst errors falls into two categories, the Type-B_1 burst error correcting codes and the Type-B_2 burst error correcting codes. [1]

The Type-B_2 codes were first studied by Wyner and Ash in 1963 and they found the Optimal Type-B_2 (n_0, n_0-1) codes for $n_0=2,3,4$. Based on the Wyner and Ash's work, Berlekamp formulated a general procedure constructing the Optimal Type-B_2 (n_0, n_0-1) codes for any n_0. Preparata discovered the same class of codes independently at about the same time. In 1965, Massey devised a decoding procedure for the Berlekamp - Preparata codes. [2,3,4]

For the burst error correction, the (mn_0, mk_0) convolutional code is analyzed by the BPM codes' analysis in the Type-B_2 burst error correcting codes.

2. Convolutional codes

(1) Matrix description

Since the parity bits in a convolutional code check the information sycmbols in blocks preceding the present block, the basic parity check matrix is given by [5]

$$h = \left[P_m^T 0 \quad P_{m-1}^T 0 \; . \; . \; . \; . \; P_1^T I \right] \qquad (1)$$

where the (n_0-k_0) by ko matrices P_i^T are arbitrary, and 0 and I represent the all zero and identity matrices of order (n_0-k_0) respectively. An n-tuple c is a code word in the convolutional code encoded by eq. (1) if and only if

$$CH^T = 0 \qquad (2)$$

where H^T denote the transpose of the parity check matrix

$$H = \begin{bmatrix} P_1^T I & & & \\ P_2^T 0 & P_1^T I & & \\ & & & \\ P_m^T 0 & P_{m-1}^T 0 & \ldots\ldots & P_1^T I \end{bmatrix} \qquad (3)$$

where blank spaces in matrix represent zeros. [1,5]

(2) Encoding

The convolutional encoder[1] by eq.(3) shown in the figure 1 operates as follows. With the D-P switch in position D(Data), the k_o data bits of the block to be encoded are shifted into the DATA REGISTER and directly out to the channel. The switch is then thrown to position P and the n_o-k_o parity checks which complete the block are read out to the channel. At this point the encoder is ready to encode the next block.

(3) Syndrome calculation

Let T be the transmitted code sequence and let E be the noise sequence added by the noise channel. Then the received sequence, R at the output of the channel is

$$R = T + E \qquad (4)$$

R can be rewritten as [1]

$$R=\left(r_1(1)\ r_1(2)\ldots r_1(n_o)\ r_2(1)\ r_2(2)\ldots r_2(n_o)\ \ldots r_m(1)\ r_m(2)\ldots r_m(n_o)\right)$$

where $r_m(i) = t_m(i)+e_m(i)$ for $i=1,2\ldots n_o$

The syndrome of this received sequences, S is defined as

$$S = RH^T \qquad (5)$$

Thus, S is also a semi infinte sequence which consists of ordered blocks

$$S=\left(s_1(1)\ s_1(2)\ldots\ s_1(n_o-k_o)\ s_2(1)\ldots s_2(n_o-k_o)\ldots\ s_m(1)\ s_m(2)\ldots s_m(n_o-k_o)\right) \qquad (6)$$

where the mth block consists of (n_o-k_o) syndrome digits

$$s_m(1)\ s_m(2)\ \ldots\ s_m(n_o-k_o)$$

From eq.(6), the (n_o-k_o) syndrome digits of the m-th block are obtained as follows

$$s_m(j)=r_m(k_o+j) + \sum_{i=1}^{k_o} r_m(i)g_1(i,j) + \sum_{i=1}^{k_o} r_{m-1}(i)g_2(i,j)+ \ldots + \sum_{i=1}^{k_o} r_1(i)\, g_m(i,j) \qquad (7)$$

for $j=1,2,\ \ldots,\ (n_o-k_o)$. Since R=T+E and $TH^T = 0$, we obtain

$$S = RH^T = EH^T \qquad (8)$$

3. Type-B_2 Codes

(1) Bounds

The burst correcting ability b of a code with Type-B_2 burst correcting ability b_2 is bounded by

$$b_2+(n_o-1) \geq b \geq b_2 - (n_o-1) \qquad (9)$$

This upper bound on b and the upper bound on b_2 given by Wyner-Ash[5] yield

$$b \leq \frac{(m-1)\ (n_o-k_o)}{\left[1 + \frac{k_o}{n_o}\right]} + n_o-1 \qquad (10)$$

(2) BPM Codes

In BPM codes no code word can have one burst confined to the m-th block while burst is in some other n_o-bit block. That is,

$$E\ H^T \neq 0 \qquad (11)$$

where E is of the form

$$E_1\ 000\ ..\ E_j\ \ldots 0$$

where E_1 represents a (nonzero) burst confined to the 1st block and E_j represents a burst confined to the j-th block.

The parity check matrix of such a code is the form

$$H = \left[B_1\ B_2\ \ldots\ B_m\right] \qquad (12)$$

where B_i is a down-shifted truncated of B_{i-1}. [2,3,4]

With the upper half of B_1 specified as $\bar{I}$, elementary row operations can transform $\left[B_1 B_j\right]$ to the form.

$$\begin{bmatrix} \bar{I} & X_j \\ 0 & Y_j \end{bmatrix} \qquad (13)$$

Thus $\left[B_1 B_j\right]$ is nonsingular if and only if the n_o by n_o matrix Y_j is nonsingular. [6] In Berlekamp's work[4] it is shown that it is possible to choose B_1 such that all Y_j are rendered simultaneously nonsingular.

$$B_1 = \begin{bmatrix} & \bar{I} & \\ 0\ 0 & \ldots\ldots & 0\ 0 \\ 1 & & \\ 1\ 1 & & \\ 1\ 1\ 1 & & \\ \vdots & & \end{bmatrix} \qquad (14)$$

4. Hardware design

(1) Block diagram

The block diagram of the burst correcting decoder is shown in Fig 2. The syndrome is calculated as usual and stored in the Syndrome Register. If the burst occurs in a block, the relationships among the syndrome bits hold, the output of the Error Correction Logic Circuits is a 1; otherwise it is a 0. If it is a 1 and if all the errors affecting the bits in the decoder are confined to a single n_0-bit block, these errors occurs in the block. The n_0 data bits affected by the burst can be corrected by adding to each the appropriate syndrome bit. As a result, the errors are eliminated in the decoder before the receiver.

(2) A Decoder by the (18,12) convolutional code.

For the (18,12) convolutional code, the parity check matrix is given by eqs. (12) and (14).

$$H = \begin{array}{llllll} 001 & & & & & \\ 010 & 001 & & & & \\ 100 & 010 & 001 & & & \\ 000 & 100 & 010 & 001 & & \\ 100 & 000 & 100 & 010 & 001 & \\ 110 & 100 & 000 & 100 & 010 & 001 \end{array} \qquad (15)$$

If the burst occurs in the first block, then the second half of the syndrome must be

$$S_2 = E_1 \left[B_{12}\right]^T \qquad (16)$$

Where E_1 denotes the burst, B_{12} denotes the lower half of the matrix B_1, and S_2 is the second half of the syndrome. In other words, since the first half of the syndrome in reverse order $\bar{S}_1$ is identical to E_1, if

$$S_2 + \bar{S}_1 \left[B_{12}\right]^T = 0 \qquad (17)$$

the burst occurs in the first block; otherwise it does not occur.

From eqs. (15) and (17)

$$S_2 + \bar{S}_1 \begin{bmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} = 0 \qquad (18)$$

that is

$$S_{21}=0,\ S_{22}=S_{13},\ S_{23}=S_{12}+S_{13} \qquad (19)$$

where S_{ij} represents the j-th bit of the i-th half of the syndrome. Similarly, if a burst occurs in the second block, third block . . . . , the three binary equations are given by the same way.

The complete decoder for the (18,12) convolutional code is shown in Fig 3.

5. Computer simulation and Experiments

(1) Computer Simulation

```
!LIST-1300

3  PRINT "------------------------------
   --------"
4  PRINT "** ERROR CORRECTING DEC
   ODER **"
5  PRINT "------------------------------
   --------"
10  PRINT "*INPUT*","*OUTPUT*"
50  DIM A(12),S(6)
55  FOR Y = 1 TO 6
60  GOSUB 1000
70 X = A(3) + A(6) + A(10) + A(11
   ) + A(12
80  IF X / 2 = INT (X / 2) THEN
    GOTO 100
90 X1 = 1: GOTO 110
100 X1 = 0
110  IF X1 = F3 THEN  GOTO 130
120 E = 1: GOTO 140
130 E = 0
140  FOR K = 6 TO 2 STEP - 1
150 S(K) = S(K - 1)
160  NEXT K
170 S(1) = E
180  PRINT F1;"";F2;"";F3,D1;"";D
    2
190  NEXT Y
195  PRINT "------------------------
    --------"
200  FOR Z = 1 TO 6
210  GOSUB 1000
220  FOR L = 6 TO 2 STEP - 1
230 S(L) = S(L - 1)
240  NEXT L
250 S(1) = F3
260  PRINT F1;"";F2;"";F3,D1;"";D
    2
270  NEXT Z
275  PRINT "------------------------
    --------"
280  GOTO 55
290  END
1000  IF S(2) = S(4) THEN  GOTO 1
      020
1010 P1 = 1: GOTO 1030
1020 P1 = 0
1030  IF (S(1) + S(4) + S(5)) / 2
       = INT ((S(1) + S(4) + S(5)
      ) / 2) THEN  GOTO 1050
1040 P2 = 1: GOTO 1060
1050 P2 = 0
1060  IF P1 + P2 + S(3) > = 1 THEN
       GOTO 1080
1070 P3 = 1: GOTO 1090
1080 P3 = 0
1090 P5 = S(4) * S(5)
1100  IF P3 + P5 > = 1 THEN  GOTO
      1120
1110 P6 = 0: GOTO 1130
1120 P6 = 1
1130 P7 = S(4) * P6
1140  IF A(12) = P7 THEN  GOTO 11
      60
1150 D1 = 1: GOTO 1170
1160 D1 = 0
1170  READ F1,F2,F3
1180  FOR I = 12 TO 2 STEP - 1
1190 A(I) = A(I - 1)
1200  NEXT I
1210 A(1) = F1
1220  IF A(12) = P7 THEN  GOTO 12
      40
1230 D2 = 1: GOTO 1250
1240 D2 = 0
1250  FOR J = 12 TO 2 STEP - 1
1260 A(J) = A(J - 1)
1270  NEXT J
1280 A(1) = F2
1290  RETURN
```

(2) Experiments

Code words can be calculated by eqs. (2) and (15)

$$C = \begin{array}{llllll} 100 & 100 & 111 & 010 & 011 & 111 \\ 010 & 001 & 100 & 110 & 010 & 001 \\ 100 & 110 & 100 & 001 & 010 & 111 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{array} \qquad (20)$$

In Fig 3 if the code words given by eq.(20) were received sequences, the syndrome sequences 000000 stored in the syndrome register. These are the correct code words. If the burst occurs in the first block, second block, third block,......, the syndrome sequences 010110, 101100, 011000,... are respectively stored in the syndrome register.

If the received sequences were as follows

R= 100 100 111 100 011 111 (21)

100 001 100 110 010 001

000 011 100 100 110 110

the corrected data are given by the experianent as follows.

D= 10 10 11 01 01 11 (22)

01 00 10 11 01 00

11 01 10 10 11 11

By the experiment, the decoder shown in Fig 3 is found to correct errors.

6. Conclusions

In case of using the parity check matrix given by wyner , Ash and Berlekamp, the syndrame sequences are found to be shifted by one bit respectively whenever the burst occurs in the first block, 2-nd block,.... Therefore this parity check matrix can be used in an error correcting decoder for the (mn_0, mk_0) convolutional codes. These decoders using convolutional codes are found to be able to the Error correction.

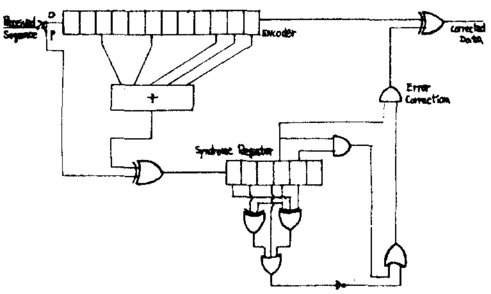

Fig 3. The complete decoder for (18,12) convolutional code.

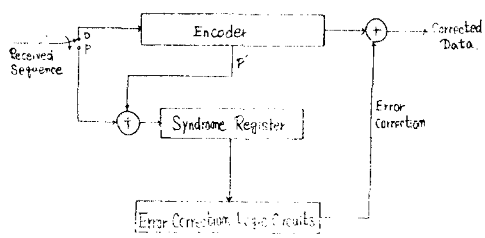

Fig 2. The block diagram of the error correcting decoder.

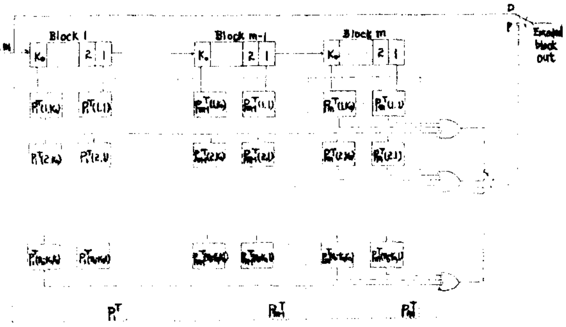

Fig 1. A K-nk_0 stage encoder for an (mn_0, mk_0) convolutional code.

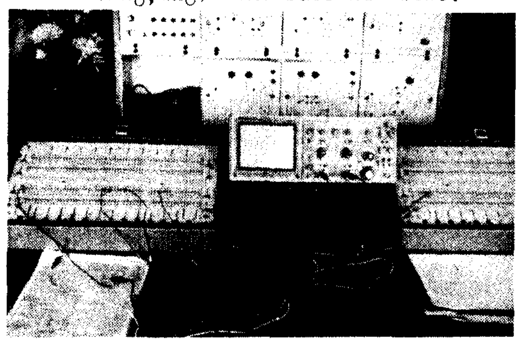

Fig 4. The experiment of the error correcting decoder.

REFERENCES

1 Shu Lin, "An Introduction to Error-Correcting Codes" pp.12-30, 185-244, 281-306. 1970

2 Yoshihiro Iwadare, "On Type-B_1 Burst-Error-Correcting convolutional Codes", IEEE Trans. on Information Theory, vol. IT-14 pp. 577-583 1968

3 J.L. Massey, "Implementation of Burst-Correcting Convolutional Codes", IEEE Trans. on Information Theory, vol. IT-11 pp.416-422 1965

4 E.R. Berlekamp, "Note on Recurrent Codes", IEEE Trans. on Information Theory, vol. IT-10 pp. 257-259 1964.

5 W.W. Peterson, "Error-Correcting Codes" pp.19-38. 144-204, 392-446 1972.

6 F.R. Gantmacher, "The theory of Matrices" vol. 1 pp. 12-19, 130-174 vol.2 pp.24-49 1974.

7 J.L.Massey, And M.K. SAIN; "Inverses of Linear Sequential Circuits" IEEE Trans. on Computers vol.C-17 pp.330-337 1968.