D. T. Brown
F. F. Sellers, Jr.

Error Correction for IBM 800-bit-per-inch Magnetic Tape

Abstract: The 800-bit-per-inch magnetic tape units that are components of IBM System/360 can correct error bursts of unlimited length in any one of their nine tape tracks. The correction technique employs a modified cyclic code in conjunction with a parity bit in each nine-bit character. There are nine check bits in the modified cyclic code and these form a check character at the end of every record. To perform the correction, a record in which an error has been detected must be reread. Errors involving more than one track within the same record are detected but are not correctable. This error correction technique operates during both read-forward and read-backward operations, and on records of any length.

Introduction

The error correction technique described here is employed by models I, II and III of the IBM 2401, 2402, 2403 and 2404 magnetic tape units. These units write characters composed of eight information bits and one parity bit in parallel across one-half-inch tape. The character parity bit is called Vertical Redundancy Check or VRC. At the end of a record consisting of any number of characters, two characters of check information are written. One of the check characters is the Longitudinal Redundancy Check or LRC character; it is used only for error detection and plays no role in the correction process. Each bit in this character is a parity check of all the data bits in the corresponding track. (VRC and LRC are the same as those used on IBM 729 tape units for error detection.) The other check character is called the Cyclic Redundancy Check (CRC) character.

There are several causes of errors on magnetic tape. These include oxide particles on the head, oxide particles on the tape, voids in the oxide coating of the tape, dust and other foreign particles on the tape and head, and tape damage due to handling. These defects usually affect only one track. However, the spacing between track centers is about 40 times the spacing between bits on a track, and it is common for more than one bit along a track to be affected. Accordingly, the error correction procedure is designed to correct almost any pattern of erroneous bits along a track, but to correct errors only in a single track within a record. When an erroneous record is read, the CRC character and VRC (character parities) are used jointly to locate the track in which the errors occurred. For error correction the record is reread. When the VRC determines that a character is in error during the rereading, the bit in that character in the track that contained errors during the original reading is corrected. Moving tape across the head sometimes changes the pattern of errors. If the pattern along the track changes between the first and second reading, error correction can still be performed. If the errors change track, error correction will be done improperly; but this situation will be detected and the new track in error can be located.

The CRC character is formed during the writing operation in the CRC Register (CRCR) as shown in Fig. 1(a). As each character is written, it is also entered into the CRCR, and the CRCR is shifted. After all the data characters have been written, the CRCR is given one more shift. The final content is the CRC character, which is then written on tape. Parallel-entry shift registers have been proposed[1] for implementation of conventional cycle codes. The CRCR is also a parallel-entry shift register, but it is constructed to create a superposition of nine cyclic codes.

During reading of the tape, the characters are again entered into the CRCR, which is shifted after each character is read. If no errors occur during reading, the CRC character is regenerated in the CRCR; and then, when the CRC character is read from the tape, it is compared with the CRC character in the CRC register by a bit-by-bit EXCLUSIVE OR operation. This results in a final register content of all 0's.* When an erroneous record is read, the result will not be all 0's, but will depend on which

D. T. Brown is at the IBM Systems Development Division Laboratory in Poughkeepsie, New York, and F. F. Sellers Jr. is currently at the MIT center for Advanced Engineering Studies.

* In the implementation, the CRC character is modified to give a final pattern of 111010111 instead of 0's. This modification is explained in the section, Altering the CRCR Character Parity, but for simplicity is omitted at this point. It does not affect the properties of the code.

track contains the errors as well as on the pattern of errors along the track. As erroneous characters are read, they are detected by the VRC and recorded in the Error Pattern Register (EPR) which is shown in Fig. 1(b). The EPR is identical to the CRCR except that it has only one input. It is shifted simultaneously with the CRCR and a 1 is entered whenever the character being entered into the CRCR has an improper VRC.

Consider that a track in error contains the pattern that would be produced by a bit-by-bit EXCLUSIVE OR of the correct data with an error pattern that is all 0's except for 1's in the bit positions in error. The final content of the CRCR would be all 0's except for the effect of the error pattern. The input to the EPR is a 1 only when an error occurs; therefore, this input is the error pattern. When the entire record is read, and the correct data have cancelled out in the CRCR, the EPR and CRCR will differ only in that the error pattern has entered the EPR in the bottom position, while in the CRCR, it has entered in the position of the track containing the error. If the error were in the bottom track, the CRCR and the EPR would match at this time. If, after the erroneous record is read, the CRCR is shifted (with the inputs set to all 0's), the CRCR content will be the same as if the errors had occurred in the next lower track. Consequently, if the CRCR is shifted and compared with the EPR after every shift, the number of shifts required until a match is obtained identifies the track containing the errors. If the CRCR and the EPR have not matched after eight shifts (and nine comparisons), errors have occurred in more than one track and are therefore uncorrectable.

An example of the correction process

Consider writing the five-character record shown in Fig. 2(a). Each character is entered into the CRCR at the time the character is written onto tape. The successive contents of the CRCR are shown in Fig. 2(b). After the last character has been entered into the CRCR, an additional shift is made (with the register inputs set at 0's); this result is entered onto tape as the CRC character. After this, the LRC character is written. The LRC character is used for error detection; it is not needed for the correction process. Figure 2(c) gives the record as it appears on tape. Note that the LRC character makes the longitudinal parity even for the entire record, including the CRC character. The CRC character is constructed to have the same parity as the parity of all the data bits; and since there is an odd number of odd parity data characters in the example, the CRC character is odd. The LRC character parity is determined from the data bits plus the CRC bits and this parity is always even.

It will be assumed that the three bits indicated by the darkened corners in Fig. 2(c) will be read erroneously [they are correct in Fig. 2(c)]. Figure 2(d) shows successive

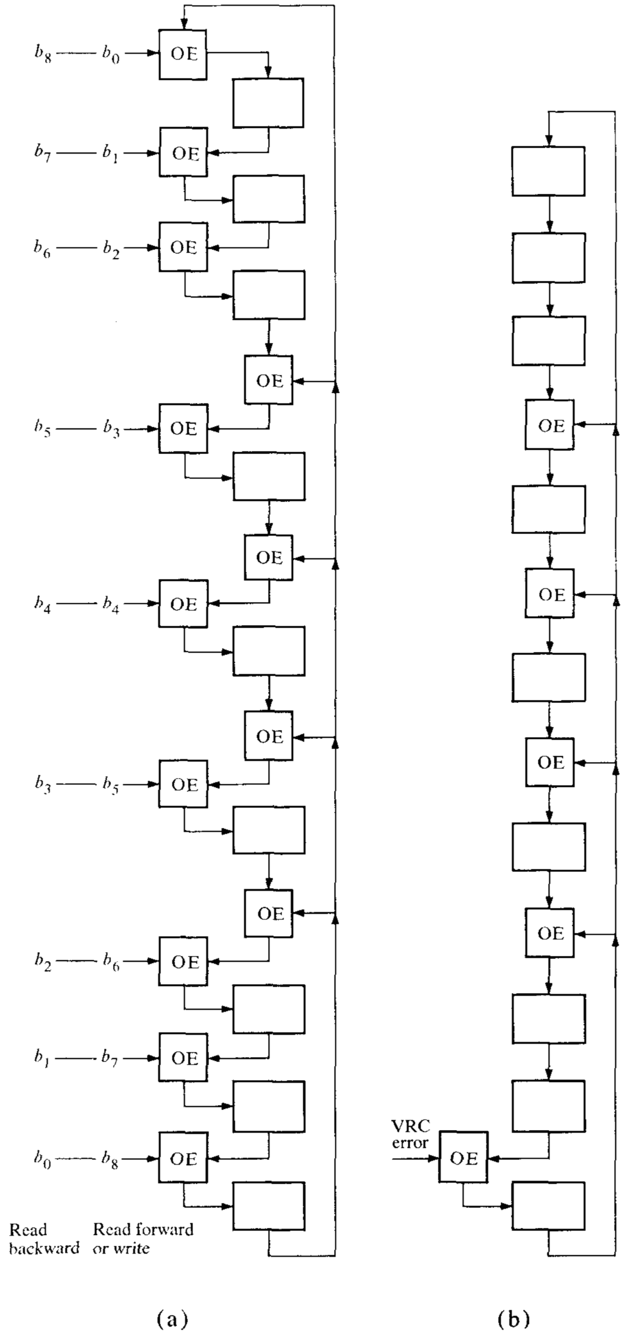

Figure 1 (a) Cyclic Redundancy Check Register; (b) Error Pattern Register. The OE boxes represent EXCLUSIVE OR's (mod 2 adders) and the open boxes represent shift cells. On a shift each bit moves down to the next cell.

contents of the CRCR when the erroneous record is read. After the second character is entered, the result is still the same as in the writing operation shown in Fig. 2(b); but, when the erroneous characters are read, the pattern differs. The last column of Fig. 2(d) gives the CRCR content that exists after the erroneous record has been read. (If no errors had occurred, the CRCR content, after

Bit	Characters 1	2	3	4	5
0	1	0	0	0	0
1	1	0	0	0	1
2	0	1	1	0	0
3	0	0	0	1	1
4	0	1	1	1	0
5	0	0	0	1	1
6	0	1	0	1	1
7	1	1	1	1	1
8	0	1	0	0	0

(a)

Shift pattern	Initial setting	Enter 1	Shift enter 2	Shift enter 3	Shift enter 4	Shift enter 5	Shift
8 → 0	0	1	0	0	1	0	1
0 → 1	0	1	1	0	0	0	0
1 → 2	0	0	0	0	0	0	0
8+2 → 3	0	0	0	0	0	1	1
8+3 → 4	0	0	1	1	0	0	0
8+4 → 5	0	0	0	1	1	1	1
8+5 → 6	0	0	1	0	1	0	0
6 → 7	0	1	1	0	1	0	0
7 → 8	0	0	0	1	0	1	0

(b)

Bit	Characters 1	2	3	4	5	CRC	LRC
0	1	0	0	0	0	1	0
1	1	0	0	0	1	0	0
2	0	1	1	0	0	0	0
3	0	0	0	1	1	1	1
4	0	1	1		0	0	1
5	0	0	0	1	1	1	1
6	0	1	0	1	1	0	1
7	1	1	1	1	1	0	1
8	0	1	0	0	0	0	1

(c)

Shift pattern	Initial setting	Enter 1	Shift enter 2	Shift enter 3	Shift enter 4	Shift enter 5	Shift enter CRC
8 → 0	0	1	0	0	1	0	0
0 → 1	0	1	1	0	0	0	0
1 → 2	0	0	0	0	0	0	0
8+2 → 3	0	0	0	0	0	1	0
8+3 → 4	0	0	1	1	0	0	0
8+4 → 5	0	0	0	0	1	0	1
8+5 → 6	0	0	1	0	0	0	1
6 → 7	0	1	1	0	1	1	0
7 → 8	0	0	0	1	0	1	1

(d)

Shift pattern	Initial setting	Shift	Shift	Shift enter	Shift	Shift enter	Shift enter
8 → 0	0	0	0	0	1	0	1
0 → 1	0	0	0	0	0	1	0
1 → 2	0	0	0	0	0	0	1
8+2 → 3	0	0	0	0	1	0	1
8+3 → 4	0	0	0	0	1	1	1
8+4 → 5	0	0	0	0	1	1	0
8+5 → 6	0	0	0	0	1	1	0
6 → 7	0	0	0	0	0	1	1
7 → 8	0	0	0	1	0	1	0

(e)

Shift pattern	Initial setting	Shift 1	Shift 2	Shift 3
8 → 0	0	1	0	1
0 → 1	0	0	1	0
1 → 2	0	0	0	1
8+2 → 3	0	1	0	1
8+3 → 4	0	1	1	1
8+4 → 5	1	1	1	0
8+5 → 6	1	0	1	0
6 → 7	0	1	0	1
7 → 8	1	0	1	0

(f)

Figure 2 An example of the correction procedure. (a) Set of data to be written onto tape. (b) Contents of CRCR after each character has been written. The right column is the CRC character. (c) The record, including the LRC and CRC characters, as written. For this example, assume the bits indicated by the darkened corners will be read erroneously. (d) Contents of CRCR after reading each character. (e) Contents of EPR after reading each character. (f) Contents of CRCR during correction. The fact that four compares are made before the CRCR matches the EPR indicates that the errors exist in the fourth track from the bottom; i.e., the 5-bit position.

character 5 had been read, would be the same as the next-to-last column in Fig. 2(b). Then, shifting this and entering the CRC character would give all 0's, verifying that there had been no error.)

During reading, the EPR is also shifted at each character time, and each time the VRC detects that an erroneous character is being read, a 1 is entered into the bottom position of the register. The result is shown in Fig. 2(e).

To understand the next step, notice that when characters that have the 5-bit incorrect are entered into the CRCR, the final CRCR content is the same as if a 1 had been entered into the 5-bit position whenever an erroneous character was read and data had never been entered. This is so because the correct data and the CRC character cancel to give all 0's; so, the final pattern is due to the erroneous bits only. Therefore, the EPR final content and the CRCR final content differ only because the errors were not in the 8-bit track. Shifting the CRCR after the record has been entered moves the entire pattern down one position. This is the same as moving the erroneous track down one position. After three shifts, the CRCR content matches the content of the EPR, indicating that the errors were three tracks above the 8-bit track (i.e., in the 5-bit track). This is shown in Fig. 2(f). If the errors had been in the 8-bit track, there would have been an immediate match, while eight shifts would have been needed if the errors had been in the 0-bit track. Eight shifts with no match would indicate that errors had occurred in more than one track and were, hence, uncorrectable.

After the track in error has been located, the record is reread, and when a character with a VRC error is read, the 5-bit is inverted. The corrected data are entered into the CRCR and the final result will be all 0's, indicating proper correction.

Mathematical description of the correction process

In this section, familiarity with the polynomial description of cyclic codes will be helpful. Reference 2 gives an elementary exposition; Ref. 3 is more complete.

A character can be represented by the polynomial:

$$M_i = b_{0i}X^0 + b_{1i}X^1 + b_{2i}X^2 + \cdots + b_{8i}X^8.$$

In a record of n data characters, M_n is the first character and M_1 is the last. The CRC character is M_0.

When a character (represented by a polynomial) in the CRCR is shifted, the effect is to multiply the polynomial by X, and reduce the result modulo a generator polynomial,

$$G = 1 + X^3 + X^4 + X^5 + X^6 + X^9.$$

If the original polynomial has an X^8 term (a 1 in the high order or X^8 position), shifting puts this 1 on the feedback paths of the register. The shift increases the exponent of every term by one, so the 1 in feedback is X^9. Reduction modulo G means that G equals 0. As mod 2 arithmetic is performed on the coefficients, subtraction and addition are identical, and are equivalent to the logical operation EXCLUSIVE OR; therefore, $X^9 = 1 + X^3 + X^4 + X^5 + X^6$. The 1 in the feedback EXCLUSIVE OR's with the register content as it is shifted, to add $1 + X^3 + X^4 + X^5 + X^6$ as shown in Fig. 1(a).

When a new character is entered into the CRCR, it EXCLUSIVE OR's with the previous content. Now the successive contents of the CRCR, as a record is written, can be listed as follows:

$$\begin{array}{ll} M_n & \\ M_{n-1} + XM_n & \mod G \\ M_{n-2} + XM_{n-1} + X^2M_n & \mod G \\ \vdots & \\ \sum_{i=1}^{n} X^{i-1}M_i & \mod G. \end{array}$$

Character index, i

Track index, j	n	$n-1$		2	1	0
0	b_{0n}	$b_{0(n-1)}$	$\cdots$	b_{02}	b_{01}	b_{00}
1	b_{1n}	$b_{1(n-1)}$	$\cdots$	b_{12}	b_{11}	b_{10}
2	b_{2n}	$b_{2(n-1)}$	$\cdots$	b_{22}	b_{21}	b_{20}
3	b_{3n}	$b_{3(n-1)}$	$\cdots$	b_{32}	b_{31}	b_{30}
4	b_{4n}	$b_{4(n-1)}$	$\cdots$	b_{42}	b_{41}	b_{40}
5	b_{5n}	$b_{5(n-1)}$	$\cdots$	b_{52}	b_{51}	b_{50}
6	b_{6n}	$b_{6(n-1)}$	$\cdots$	b_{62}	b_{61}	b_{60}
7	b_{7n}	$b_{7(n-1)}$	$\cdots$	b_{72}	b_{71}	b_{70}
8	b_{8n}	$b_{8(n-1)}$	$\cdots$	b_{82}	b_{81}	b_{80}

Forward tape motion $\longleftarrow$

Figure 3 Notation used to identify bits in a tape record. The first character written has index n.

The last line is the CRCR content after the last character of data has been entered. After one more shift, with the inputs set to 0's, the content is the CRC character.

Define the polynomial M by

$$M = \sum_{i=1}^{n} X^i M_i.$$

Then the CRC character can be written as:

$$M_0 = M \quad \mod G. \qquad (1)$$

The relation between M and the record, as it appears on tape, is shown in Fig. 3. Each column is a character and each row is a track. Each bit position is defined by a character index i and a track index j. M can be calculated by associating a power of X equal to the sum of the character and track indices with each bit and performing polynomial addition.

When an error-free record is read, the characters enter the CRCR in the same way as in writing, except that instead of an input of all 0's for the final shift, the CRC character is read in. This gives $M + M_0 = 0 \mod G$; that is, the final CRCR content is all 0's.

A record with errors in the track with index j is represented by $M + M_0 + X^jE$, where

$E = \sum X^i$, summed over the indices of all characters in error.

The polynomial E represents the error pattern along the track. If a bit of the CRC character is in error, it contributes a term X^0. The final CRCR content on reading an erroneous record is:

$$M + M_0 + X^jE = X^jE \quad \mod G \quad (0 \le j \le 8).$$

The EPR differs from the CRCR only in that its only input is in the high order (X^8) position. It also multiplies by X and reduces modulo G when it is shifted; but the only possible inputs are 0 and X^8. Because the EPR is shifted as every character is read, and X^8 is entered every time there is a VRC (character parity) error, the content of the EPR at the end of the record is $X^8E \bmod G$.

At this time the CRCR and EPR are compared and if they do not match, the CRCR is shifted until they do (a maximum of eight shifts). To uniquely locate the erroneous track, it is necessary and sufficient that

$$X^{j+k}E = X^8E \quad \bmod G, \tag{2}$$

only for $j + k = 8$ and not for $j + k < 8$. The number of times that the CRCR is shifted to give a match is k. Since k is known, the index j of the track in error can be determined from $j + k = 8$.

Equation (2) can be rewritten as

$$X^{j+k}(1 + X^{8-j-k})E = 0 \quad \bmod G. \tag{3}$$

For Eq. (3) to be satisfied for $j + k < 8$, G must evenly divide the left-hand side. The irreducible factors of G are $G_1 = (1 + X)$ and $G_2 = (1 + X + X^2 + X^4 + X^6 + X^7 + X^8)$. While $(1 + X^{8-j-k})$ contains the factor $(1 + X)$ only E could contain the factor G_2. Therefore, the only (single-track) errors for which the track cannot be determined are errors of the form

$$E_{\text{uncorrectable}} = FG_2, \tag{4}$$

where F is an arbitrary polynomial.

In this case, the CRCR content after the record has been read but before shifting k times will be $X^jFG_2 \bmod G$.

The following shows that the value of this must be 0 or G_2.

$$\begin{aligned} XG_2 &= XG_2 + G && \bmod G, \text{ or} \\ XG_2 &= XG_2 + (1 + X)G_2 && \bmod G, \text{ so} \\ XG_2 &= G_2 && \bmod G, \end{aligned}$$

and so

$$X^kG_2 = G_2 \qquad \bmod G. \tag{5}$$

From Eq. (5) it follows that

a) if X^jF has an even number of terms, $X^jFG_2 = 0 \bmod G$, because G_2 is added to itself an even number of times, or
b) if X^jF has an odd number of terms, $X^jFG_2 = G_2 \bmod G$, because G_2 is added to itself an odd number of times.

The CRCR is tested for the two patterns (all 0's or G_2), and if one of the two is found, a match with the EPR is not attempted. Uncorrectable errors with $G = 0$ are distinguishable from correct records, which also have $G = 0$, by the occurrence of character parity errors that are detected by VRC.

Since $E_{\text{uncorrectable}}$ must have the factor G_2, an uncorrectable pattern of errors must have a separation between the first bit in error and the last bit in error of at least nine bit positions. For longer "error bursts" the fraction of errors of a given length that are uncorrectable is 2^{-8}. This is derived by counting the different values F can assume.

If G had been chosen to be irreducible instead of having the factor $1 + X$, fewer error patterns would result in an uncorrectable error. This was not done for two reasons. First, to correct errors in the CRC character, the correct parity of this character must be known. The following shows that this requires the $1 + X$ factor in G.

Rewriting Eq. (1) gives:

$$M_0 + FG = M.$$

Only if $G = (1 + X)G_2$ will FG have an even number of terms, i.e., even parity (see Ref. 1, theorem 2). If this is the case, then the parity of M_0 (the CRC character) equals the parity of M, because the sum of two even polynomials (polynomials with an even number of terms) is an even polynomial, and the sum of an even and an odd polynomial is an odd polynomial. From the definition of M, its parity must be the parity of all the data bits in the record. If VRC (character parity) is even, the parity of M must be even. If VRC is odd, the parity of M will equal the parity of the count of data characters in the message. Normally, odd VRC is used, so a mod 2 count of characters is made during reading. If there is an even number of characters, the parity of M, and hence M_0, should be even; and if the count is odd, the parity of M_0 should be odd. If the parity of M_0 is not as calculated, an error has been detected, and a 1 is entered into the EPR.

The second reason for choosing a G with a $1 + X$ factor is given in the next section.

- *Read backward*

The 2400 Series tape units can read records forward or backward. To locate the track in error during the read-backward operation, a constraint must be placed on the choice of G, and a minor change must be made to the way data is entered into the CRCR.

The record, including the CRC character, is

$$M + M_0 = \sum_{i=0}^{n} X^i M_i,$$

which is a sum of terms, one for every 1 in the record, and each with an exponent equal to the sum of row and column indices as shown in Fig. 3. When the tape unit is operating in the read-backward mode, the inputs to the CRCR are "twisted" so that bit b_{0i} enters the high order

(X^8) position, bit b_{8i} enters the low order (X^0) position, and so forth. The input, in polynomial form, is

$$M_i^* = b_{8i}X^0 + b_{7i}X^1 + b_{6i}X^2 + \cdots + b_{0i}X^8,$$

where M_i^* is called the reciprocal of M_i.

Reading the record backward gives a final CRCR content of

$$(M + M_0)^* = \sum_{i=0}^{n} X^{n-i} M_i^* \quad \text{mod} \quad G.$$

The polynomial $(M + M_0)^*$ is the reciprocal of $M + M_0$ because the terms are all reversed; X^0 goes to X^{n+8}, X^1 goes to X^{n+7}, and so forth.

Because $M + M_0 = 0 \bmod G$,

$$M + M_0 = FG,$$

so

$$(M + M_0)^* = (FG)^*. \tag{6}$$

The generator polynomial G was chosen to be symmetric so that $G = G^*$. Because of this,

$$(FG)^* = F^*G. \tag{7}$$

This can be seen by examining the process of multiplication, taking G as the multiplier.

Equations (6) and (7) give $(M + M_0)^* = 0 \bmod G$. Therefore, when the unit is reading backward, the final CRCR content is all zeros if there has been no error.

If errors do occur, the final EPR content will be X^8E^* mod G. The final CRCR content will be $X^{8-j}E^*$ mod G. Because of the "twisted" inputs, the number of CRCR shifts to give a match is j, the index of the track in error, rather than $8 - j$ as was the case for reading forward. Because $G_2 = G_2^*$, it follows from Eq. (4), that E^* is uncorrectable if and only if E is uncorrectable.

The requirement $G = G^*$ can be satisfied (for degree nine polynomials) only if G has the factor $1 + X$.

- *Altering the CRC character parity*

It is desirable that the LRC character contain at least one 1 to identify the start of the record during backward reading. This is achieved by adding the polynomial $G_2 = 1 + X + X^2 + X^4 + X^6 + X^7 + X^8$ to the content of the CRCR at the finish of data writing. The CRC character becomes

$$M_0' = M_0 + G_2.$$

This modified CRC character will have parity opposite the parity of M_0. This in turn causes the LRC parity to be odd, so the LRC character is never all zeros.

The CRC character parity could be altered by adding to it any polynomial with an odd number of terms (i.e., by inverting an odd number of bits), but the use of G_2 prevents a complication to the procedure for locating the track in error. The final CRCR content, on reading a correct record, will be G_2 instead of all 0's. When a record does have track j in error, the final content will be

$$X^jE + G_2 \quad \text{mod } G.$$

At this point, it would be possible to subtract G_2 from the CRCR content and proceed to compare the result with the EPR content, which is X^8E mod G. However, this is not necessary. If G_2 is not subtracted, after k shifts the CRCR contains

$$X^{j+k}E + X^kG_2 \quad \text{mod } G.$$

But, from Eq. (5), this is equal to

$$X^{j+k}E + G_2 \quad \text{mod } G.$$

In other words, the G_2 term remains unaltered and can be subtracted after the shifting rather than before. Subtraction after shifting is much easier to implement because the register content need not be changed. It is only necessary to use inverted signals from the CRCR positions corresponding to terms in G_2 when doing the comparisons with the EPR.

Uncorrectable single-track errors will still result in a final CRCR content of either G_2 or all 0's because $G_2 + G_2 = 0$ and $G_2 + 0 = G_2$.

Conclusions

The CRC error correction technique corrects most common errors on 800-bit-per-inch magnetic tape. Each tape record requires only nine check bits more than the number used for error detection in previous tape systems. The amount of hardware needed is modest. The CRC causes a negligible increase in the time needed to write a record or to read an error-free record. When errors occur, a backspace and a reread are required.

References

1. M. Y. Hsiao, "Theories and Applicaitons of Parallel Linear Feedback Shift Registers," Doctoral Thesis at the University of Florida, December 1967, also IBM Technical Report TR 00.1708, Systems Development Division Laboratory, Poughkeepsie, New York.
2. W. W. Peterson and D. T. Brown, "Cyclic Codes for Error Detection," *Proc. IRE* **49,** 228 (1961).
3. W. W. Peterson, *Error Correcting Codes,* John Wiley & Sons, Inc., New York, 1961.

Received October 14, 1969