Optimization of Cyclic Redundancy-Check Codes with 24 and 32 Parity Bits

Guy Castagnoli, Stefan Bräuer, and Martin Herrmann

***Abstract*—The method of Fujiwara *et. al.* for efficiently computing the minimum distance of shortened Hamming codes via the weight distribution of their dual codes is extended to treat arbitrary shortened cyclic codes. Using this extended method implemented on a high-speed special-purpose processor, several classes of CRC codes with 24 and 32 parity bits have been investigated. The CRC codes of each class are known to have the same minimum distance $d_{\min,L}$ in a certain range L of block lengths n, and within each class, that CRC code has been determined whose minimum distance exceeds $d_{\min,L}$ up to the largest block length. The $d_{\min}$ profiles of the resulting codes, i.e., the minimum distances $d_{\min}$ as a function of the block length n, are presented and compared with the $d_{\min}$ profiles of recent suggestions of Merkey and Posner, as well as with the $d_{\min}$ profile of the widely used 32 parity-bit standard code recommended in IEEE-802. The optimization of the $d_{\min}$ profile is warranted by the fact that given the number of p of parity bits and the block length n, it is the minimum distance that determines the order of magnitude of the probability P_{ue} of undetectable errors on low-noise binary symmetric channels (BSC's). This is seen from plots of P_{ue} versus n for our codes and for the IEEE-802 code on a low-noise BSC and on a fairly noisy BSC.**

I. Introduction

CYCLIC redundancy-check codes (CRC codes) are used for detecting errors that occur during transmission of digital (binary) information. This is done by appending to a block

$$i = [i_0, i_1, \cdots, i_{k-1}] \tag{1}$$

of k binary information digits i_l, $l = 0, 1, \cdots, k-1$, a block

$$r = [r_0, r_1, \cdots, r_{p-1}] \tag{2}$$

of p parity bits r_j, $j = 0, 1, \cdots, p-1$, yielding a frame (or codeword) $c = [r, i]$ consisting of $n = k + p$ binary digits. The block r of parity bits is computed from i, using a linear feedback shift register (LFSR) in such a way that

$$r(x) = (x^p \cdot i(x)) \bmod g(x) \tag{3}$$

where

$$i(x) \stackrel{\text{def}}{=} i_0 + i_1 \cdot x + \cdots + i_{k-1} \cdot x^{k-1} \tag{4a}$$

and

$$r(x) \stackrel{\text{def}}{=} r_0 + r_1 \cdot x + \cdots + r_{p-1} \cdot x^{p-1} \tag{4b}$$

are the information and parity bits, respectively, interpreted as polynomials, and where $g(x)$ is the generator polynomial of the code and is implemented in the LFSR. Error detection at the receiving end is achieved by computing the parity bits from the received information block $\hat{i}$ and comparing these with the received parity bits $\hat{r}$. Any discrepancy between these two sets of parity bits then indicates the presence of transmission errors in the received frame. It is possible, however, that transmission errors reside in the received frame that cannot be detected by this procedure. Whenever channel noise affects both the information block i and the block r of parity bits in such a way that the received frame $\hat{c} = [\hat{r}, \hat{i}]$ satisfies

$$\hat{r}(x) = \left(x^p \cdot \hat{i}(x)\right) \bmod g(x), \tag{5}$$

the errors cannot be detected by parity checking. The performance of a CRC code is thus measured by the probability P_{ue} of undetectable channel errors occurring. An undetectable error pattern e can also be viewed as one transforming a given codeword c into a different codeword $c' = c + e$. Due to the linearity of CRC codes, the undetectable error patterns are precisely the nonzero codewords. The undetected-error probability P_{ue} is thus the probability that channel noise produces an error pattern equal to a nonzero codeword of the CRC code. On a low-noise binary symmetric channel, which tends to produce low-weight error patterns more frequently than error patterns with a large Hamming weight, it thus appears reasonable to use a CRC code that has a maximum minimum distance. Indeed, as a computation of P_{ue} for specific CRC codes shows, $d_{\min}$ does dominate the undetected-error probability on low-noise BSC's. Correspondingly, our following investigation of CRC codes aims at determining $d_{\min}$ at various block lengths and at optimizing the $d_{\min}$ profile, i.e., $d_{\min}$ as a function of the block length n.

In Section II, we present our extension of Fujiwara's algorithm. While the method of Fujiwara *et al.* [1] is for determining the minimum distance of a shortened Hamming code via the weight distribution of its dual code, we can perform Fujiwara's algorithm with no restrictions on the factorization of the generator polynomial $g(x)$ of C. In Section III, we present the $d_{\min}$ profiles, i.e., the minimum distances at all block lengths, of the best CRC codes that resulted from our search for good CRC codes with 24 and 32 parity bits. For reasons of comparison, the $d_{\min}$ profiles of the CRC codes

Paper approved by the Editor for Coding and Applications of the IEEE Communications Society. Manuscript received January 24, 1990; revised February 21, 1992.

G. Castagnoli is at Rautistrasse 159, 8048 Zurich, Switzerland.
S. Bräuer is with Staefa Control System AG, CH-8712 Stafa, Switzerland.
M. Herrmann is with the Reliability Laboratory, ETH-Zentrum, 8092 Zurich, Switzerland.

IEEE Log Number 9210330.

proposed by Merkey and Posner [2] and of the widely used IEEE-802 standard are also presented. Moreover, we have plotted the probability P_{ue} of undetectable errors for our codes and for the standard code IEEE-802 on both a low-noise binary symmetric channel (BSC) and a fairly noisy BSC. Section IV, finally, contains some concluding remarks on the usefulness and properness of our codes.

II. A Method for Computing the Minimum Distance of CRC Codes Via the Dual Code's Weight Distribution

The minimum distance of a CRC code at a given block length can be determined in principle by explicitly computing the Hamming weight of each nonzero codeword and keeping track of the smallest determined Hamming weight. However, in general, there are far too many codewords to check, e.g., if $k = 1000$, then there are $2^{1000} \approx 10^{300}$ codewords, and so this procedure is computationally infeasible. Fortunately, though, there exist the MacWilliams' identities (6) [3], [4]

$$A^{(n)}(z) = 2^{-p}(1+z)^n B^{(n)}\left(\frac{1-z}{1+z}\right) \tag{6}$$

which express the weight distribution $A^{(n)}(z) \stackrel{\text{def}}{=} A_0^{(n)} + A_1^{(n)}z + \cdots + A_n^{(n)}z^n$ of a CRC code at length n in terms of the weight distribution $B^{(n)}(z) \stackrel{\text{def}}{=} B_0^{(n)} + B_1^{(n)}z + \cdots + B_n^{(n)}z^n$ of its dual code $C_\perp$, which is the code orthogonal to C in $GF(2)^n$. Thus, knowing how many codewords of the dual code have a Hamming weight of i, for $i = 0, 1, \cdots, n$, we can apply MacWilliams' identities to determine the weight distribution of the CRC code itself at the respective block length, and thereby also its minimum distance $d_{\min}$. Since the MacWilliams' identities constitute a linear transformation, they are not overly time consuming to apply, and since the dual code of a CRC code has "only" 2^p codewords at every block length, we see that for CRC codes C with $p = 24$ or $p = 32$ parity bits, it is indeed possible to compute $d_{\min}$ via the weight distribution of its dual code $C_\perp$. Since there are many CRC codes from which to choose an optimum, this computation must be repeated very often. Therefore, it is still material to compute the 2^p codewords of the dual code efficiently. While the direct computation of each codeword requires $2^p \cdot n$ "steps," the method presented in this paper takes between 2^p and $2^p \cdot 2$ steps, depending on the block length n. For typical n (e.g., $n = 1000$), this constitutes a significant gain, especially when dealing with CRC codes with 32 parity bits where the factor $2^p \approx 4.3 \cdot 10^9$ alone indicates that either method is at the verge of computational intractability.[1,2]

Consider the $(n-p) \times n$ binary generator matrix G of the shortened cyclic code C of length n generated by $g(x) = g_0 + g_1 x + \cdots + g_{p-1}x^{p-1} + x^p$:

$$G = \begin{pmatrix} g_0 & g_1 & \cdots & g_{p-1} & 1 & & & & & \\ & g_0 & g_1 & \cdots & g_{p-1} & 1 & & & & \\ & & g_0 & g_1 & \cdots & g_{p-1} & 1 & & & \\ & & & \ddots & & & \ddots & & & \\ & & & & \ddots & & & \ddots & & \\ & & & & & g_0 & g_1 & \cdots & g_{p-1} & 1 \end{pmatrix}. \tag{7}$$

The generator matrix G is at the same time the parity-check matrix of the dual code $C_\perp$ of C, and we can see that the digits c_i of the codewords in the dual code constitute a linear recurring sequence with the recurrence relation

$$c_i = g_0 c_{i-p} + g_1 c_{i-p+1} + \cdots g_{p-1}c_{i-1}, \quad i = p, p+1, \cdots, n. \tag{8}$$

The recurrence relations (8), which hold for the infinitely continued sequence $c_0, c_1, c_2, \cdots$, can be expressed through the rationality of the formal power series

$$c\left(\frac{1}{x}\right) \stackrel{\text{def}}{=} \sum_{i=0}^{\infty} \frac{c_i}{x^{i+1}} \tag{9}$$

by

$$c\left(\frac{1}{x}\right) \cdot g(x) = u(x) \tag{10}$$

where $u(x)$ is a polynomial with $\deg u(x) < \deg g(x)$. Given $u(x)$, we can obtain $c(1/x)$ by expanding the rational function $u(x)/g(x)$ in powers of x^{-1}. This can be realized by a linear feedback shift register (LFSR) in which a shift operation transforms its state $u(x)$ into $(x \cdot u(x)) \bmod g(x)$.

$$\begin{aligned} \frac{u(x)}{g(x)} &= \frac{u_0 + u_1 x + \cdots + u_{p-1}x^{p-1}}{g_0 + g_1 x + \cdots + g_{p-1}x^{p-1} + x^p} \\ &= \frac{1}{x} \cdot \frac{u_0 x + u_1 x^2 + \cdots + u_{p-1}x^p}{g_0 + g_1 x + \cdots + g_{p-1}x^{p-1} + x^p} \\ &= \frac{1}{x} \cdot \frac{u_{p-1}g(x) + x \cdot u(x) - u_{p-1}g(x)}{g(x)} \\ &= \frac{u_{p-1}}{x} + \frac{x \cdot u(x) \bmod g(x)}{g(x)}. \end{aligned} \tag{11}$$

The outputs of this shift register constitute the digits of a dual codeword. For every initial LFSR content, we obtain a different codeword, and since there are 2^p initial LFSR states, we can compute in this way all codewords of the dual code to the CRC code at any desired length. Rather than initializing the LFSR 2^p times, i.e., once for each new codeword, we proceed as follows. Having computed the weight of one codeword, we can determine the weight of a further codeword without reinitializing the LFSR by simply shifting the LFSR one step and discarding the first digit of the old codeword. The change in weight between the old and new codeword (either 0, +1, or −1) is determined from the new digit and the discarded digit. The most efficient way to realize this idea is by operating two LFSR's as follows. Load a particular initial state $u_0(x)$ into two LFSR's, and then compute the Hamming weight w of the corresponding codeword in $C_\perp$ of desired length by operating the first of these LFSR's n times and counting the

[1] Our method was implemented on a special-purpose processor that runs with a 40 MHz clock, and thus takes between 107 and 215 s to determine $d_{\min}$ of a 32 parity-bit CRC code at a given block length.

[2] Fujiwara *et al.* [12] used a method to determine the minimum distance of a CRC code directly. It is more efficient than our method, but does not yield the dual code's weight distribution which is needed to determine the properness of the codes.

number of output 1's. The first LFSR state is thereby set to $u_n(x)$. Now, shift both LFSR's simultaneously. The first LFSR [the "leading LFSR," starting at $u_n(x)$] produces a new digit, while the second LFSR [the "trailing LFSR," starting at $u_0(x)$] outputs the discarded digit. Using this method, which is due to Fujiwars *et al.* [1], and which was independently discovered by Günther [5], we can compute the Hamming weight of further codewords with only one shift per codeword. However, we compute the weights of new codewords only as long as we reach new LFSR states $u_i(x)$. Depending on $u_0(x)$ and $g(x)$, we can determine the number of codewords that are encountered starting at $u_0(x)$ by first noting that the state of the trailing LFSR after N shifts will be

$$u_N(x) = x^N \cdot u_0(x) \bmod g(x)\,. \tag{12}$$

Now, $u_N(x) = u_0(x)$ holds exactly when $g(x)$ divides $u_0(x)\cdot\left(x^N-1\right)$. This is equivalent to $g(x)/\gcd(g(x),u_0(x))$ dividing x^N-1. Thus, the number N of different codewords produced will be

$$N = \operatorname{ord}\left(\frac{g(x)}{\gcd(g(x),u_0(x))}\right). \tag{13}$$

Therefore, if and only if $g(x)$ is a primitive polynomial, i.e., if and only if $g(x)$ satisfies

$$\operatorname{ord}\, g(x) = 2^{\deg g(x)} - 1 \tag{14}$$

will all nonzero codewords be encountered when starting from a single nonzero initial state $u_0(x)$. When $g(x)$ is not primitive, the nonzero LFSR states are partitioned into several equivalence classes that we will call cyclic cosets and that are defined by the equivalence relation

$$\begin{aligned} u(x) \equiv v(x) \iff\ & \exists\, j \\ \text{such that } v(x) &= \left(x^j \cdot u(x)\right) \bmod g(x)\,. \end{aligned} \tag{15}$$

In this case, several initial states (coset leaders) $u_0(x)$ must be used, and we now show how to determine these.

Consider the factorization $g(x) = \prod_{i=1}^{J} g_i(x)^{e_i}$ of $g(x)$ into irreducible polynomials. With any shift register state $u(x)$, associate the partial fraction decomposition of $u(x)/g(x)$:

$$\frac{u(x)}{g(x)} = \sum_{i=1}^{J} \frac{u_i(x)}{g_i(x)^{e_i}} \tag{16}$$

where

$$\deg\, u_i(x) < \deg\, g_i(x)^{e_i} \qquad \forall\, i\,. \tag{17}$$

The J-tuples

$$\begin{aligned} (u_1(x), u_2(x), \cdots, u_J(x)) \in\ & GF(2)[x]/(g_1(x)^{e_1}) \times \cdots \\ & \times GF(2)[x]/(g_J(x)^{e_J}) \end{aligned} \tag{18}$$

as isomorphous as a commutative ring to the residue class ring

$$GF(2)[x]/(g(x))\,. \tag{19}$$

The isomorphism σ is established through

$$\sigma : u(x) \mapsto (u_1(x), \cdots, u_J(x)) \tag{20a}$$

with

$$u_i(x) \stackrel{\text{def}}{=} u(x) \bmod g_i(x)^{e_i}\,. \tag{20b}$$

The inverse isomorphism σ^{-1} is defined by

$$u(x) = \left(\sum_{i=0}^{J} u_i(x)\cdot v_i(x)\frac{g(x)}{g_i(x)^{e_i}}\right) \bmod g(x) \tag{20c}$$

where $v_i(x)$ is the multiplicative inverse of $g(x)/g_i(x)^{e_i}$ in $GF(2)[x]/(g_i(x)^{e_i})$.

By this isomorphism, the J components of the elements $x^l u(x) \bmod g(x)$, $l = 1, 2, \cdots$, of a cyclic coset of each constitute cyclic cosets of their own. The problem of finding a set of representatives (coset leaders) of the cyclic cosets in $GF(2)[x]/(g(x))$ is thus reduced to finding such representatives of the J component cyclic cosets in $GF(2)[x]/(g_i(x)^{e_i})$ for $i = 1, \cdots, J$, and then to combining these in all possible ways. Assume that we have already solved the problem in each of the component residue class rings separately. Furthermore, let $\langle c_j\rangle$ be a cyclic coset in the jth residue class ring for $j = 1, \cdots, J$. Now, we wish to determine all cyclic cosets $\langle c\rangle$ in the ring $GF(2)[x]/(g(x))$ that have as component cyclic cosets exactly the given cosets $\langle c_j\rangle$, $j = 1, 2, \cdots, J$. If d_j denotes the number of elements in the jth component cyclic coset $\langle c_j\rangle$, and if $c_j^{(k)}$ denotes the element $\left(x^k \cdot c_j(x)\right) \bmod g_j(x)^{e_j}$, then a set of coset leaders of these cyclic cosets $\langle c\rangle$ in $GF(2)[x]/(g(x))$ is furnished by

$$\left(c_1, c_2^{(k_2)}, \cdots, c_J^{(k_J)}\right) \tag{21a}$$

where

$$0 \le k_i \le K_i - 1 \qquad \text{for } i = 2, \cdots, J \tag{21b}$$

and

$$K_i \stackrel{\text{def}}{=} \gcd(d_i, \operatorname{lcm}(d_1, d_2, \cdots, d_{i-1}))\,. \tag{21c}$$

Hence, the total number of equivalence classes is

$$\begin{aligned} N = \prod_{i=2}^{J} K_i &= \prod_{i=2}^{J} \frac{d_i \cdot \operatorname{lcm}(d_1, \cdots, d_{i-1})}{\operatorname{lcm}(d_1, \cdots, d_i)} \\ &= \frac{d_1 \cdot d_2 \cdot \cdots \cdot d_J}{\operatorname{lcm}(d_1, d_2, \cdots, d_J)}\,. \end{aligned} \tag{22}$$

We can understand this by the following principle.

Action of a Cyclic Group on a Cartesian Product:

If, under the action of a cyclic group $G = (c)$ on a finite set, we have two orbits T_1 and T_2 of sizes d_1 and d_2, respectively, then on the Cartesian product $T \stackrel{\text{def}}{=} T_1 \times T_2$, which has $d_1 \cdot d_2$ points, the group G induces an action that partitions T into orbits of $\operatorname{lcm}(d_1, d_2)$ points. There are, hence,

$$N = \frac{d_1 \cdot d_2}{\operatorname{lcm}(d_1, d_2)} = \gcd(d_1, d_2) \tag{23}$$

orbits. As coset leaders for these orbits, one can take

$$\left(t_1, t_2^{(k)}\right) \qquad \text{where } 0 \le k \le N-1 \tag{24}$$

and where $t_i^{(k)}$ is the image of t_i under the action of $c^k \in G$.

Proof: Since the orbit of t_i under the action of the cyclic group $G = (c)$ contains d_i points, we have

$$t_i^{(l)} = t_i \iff d_i | l \qquad \text{for } i = 1, 2 . \tag{25}$$

Therefore,

$$t_i^{(l)} = t_i, \qquad i = 1, 2 \iff \operatorname{lcm}(d_1, d_2) | l \tag{26}$$

and so the product orbit contains $\operatorname{lcm}(d_1, d_2)$ points. The elements $(t_1, t_2^{(k)})$, with $0 \le k < N - 1$, are on distinct orbits since the subgroup of elements in G that keep t_1 fixed is (c^{d_1}), and hence these elements map $t_2^{(j)}$ onto elements $t_2^{(j+l \cdot \gcd(d_1, d_2))}$ whose "indexes" $(j + l \cdot \gcd(d_1, d_2))$ are congruent modulo $N = \gcd(d_1, d_2)$.

We are left with the problem of finding the coset leaders of the cyclic cosets defined by (15) in the case where $g(x) = f(x)^e$ is a power of an irreducible polynomial $f(x)$. In this case, the ring

$$A = GF(2)[x]/(f(x)^e) \tag{27}$$

is irreducible, i.e., it cannot be decomposed further into a direct sum of other rings. The simplest instance is when $g(x) = f(x)$ is primitive. Then there are only two equivalence classes of LFSR states, that of the zero state $u(x) = 0$, and the cyclic coset of all nonzero states for which we can take $u(x) = 1$ as coset leader. This was the situation considered by Fujiwara *et al.* [1] who investigated shortened Hamming codes.

The next simplest case is when $g(x)$ is irreducible, but not primitive. Let $m = \deg g(x)$ and $n = \operatorname{ord} g(x)$. Apart from the cyclic coset of the zero state, which is always present, we have several cyclic cosets of nonzero states. Let $\alpha(x)$ be a representative of a primitive element of the finite field

$$F = GF(2)[x]/(g(x)) \tag{28}$$

such that the root x of $g(x)$ in $F[x]$ can be expressed in F as $x = \alpha(x)^t$ with

$$t \stackrel{\text{def}}{=} \frac{2^m - 1}{n} . \tag{29}$$

If we write the nonzero state $u(x)$ as $u(x) = \alpha(x)^i \bmod g(x)$, the cyclic coset

$$\left\{u(x), (x \cdot u(x)) \bmod g(x), \left(x^2 \cdot u(x)\right) \bmod g(x), \cdots\right\}, \tag{30}$$

written in terms of indexes with respect to $\alpha(x)$, reads

$$i, i + t, i + 2t, \cdots , \tag{31}$$

from which we see that the indexes of the states equivalent to $u(x)$ are precisely those indexes that are congruent to i modulo t. Hence, a set of coset leaders of the cyclic cosets of nonzero states is given by

$$c_i(x) = \alpha(x)^i \bmod g(x) \qquad \text{with } i = 0, 1, \cdots, t - 1 . \tag{32}$$

We now come to the most general case where $g(x)$ is a power (at least 2) of an irreducible polynomical $f(x)$. In this case, the multiplicative structure of the irreducible ring (27) of residue classes modulo $g(x)$ is no longer a cyclic group, as in the case of the finite field F, but is only a semi-group. Fortunately, x is coprime with $g(x) = f(x)^e$, so that the greatest common divisor of the elements of a cyclic coset with $g(x)$ is the same for all elements, namely, $f(x)^s$ for some s with $0 \le s < e$. The nonzero cyclic cosets can therefore be grouped according to the multiplicity of $f(x)$ in their elements. If

$$u(x) = \overline{u}(x) \cdot f(x)^s \tag{33}$$

with $f(x), \nmid \overline{u}(x)$, then the elements equivalent to $u(x)$, $\left(x^l \cdot u(x)\right) \bmod g(x)$ with $l = 0, 1, 2, \cdots$, can be computed as

$$\begin{aligned} \left(x^l \cdot u(x)\right) \bmod f(x)^e &= \left(x^l \overline{u}(x) \cdot f(x)^s\right) \bmod f(x)^e \\ &= \left(x^l \overline{u}(x) \bmod f(x)^{e-s}\right) \cdot f(x)^s , \end{aligned} \tag{34}$$

i.e., these elements are $f(x)^s$ times the elements of equivalence classes of polynomials that are coprime with $f(x)$ and that are taken from the irreducible ring $GF(2)[x]/(f(x)^{e-s})$. We have thus reduced the problem of finding the coset leaders in $GF(q)[x]/(f(x)^e)$ to the problem of finding them in the multiplicative groups M_t of elements coprime to $f(x)$, where

$$M_t = \left(GF(2)[x]/f(x)^t\right)^* \qquad t = 1, 2, \cdots, e . \tag{35}$$

The groups M_t, which are commutative and finite, posses a decomposition

$$M_t = (a_1) \times (a_2) \times \cdots \times (ar) \tag{36}$$

as a direct product of cyclic subgroups (a_i); see [6, p. 7]. Hence, we must find generators a_i of these subgroups of M_t. Then, in principle, we could determine a set of coset leaders using properties of the action of a cyclic group on Cartesian products. This would require, however, that we determine the representation of $x \in M_t$ as a product of powers of the a_i's, which is a little more than is needed in the method that we actually use here.

Generators of $\left(GF(2)[x]/\left(f(x)^t\right)\right)^*$ *[7]:* The multiplicative group M_t of elements coprime of $f(x)$ of $GF(2)[x]/(f(x)^t)$, for $t \ge 2$, possesses the following set of generators. The "primitive" generator $a_p = (a_p(x))$, which has order $2^{\deg f} - 1$, and the generators $a_{j,k} = (a_{j,k}(x))$ of the 2-Sylov subgroup[3] $P^{(t)}$ of M_t, where

$$a_{j,k} = 1 + x^j \cdot f(x)^{1+2k} \tag{37}$$

for $0 \le j \le \deg f - 1$ and $0 \le k \le \lfloor (t-2)/2 \rfloor$. The order of $a_{j,k}$ is

$$\eta_{j,k} = 2^{\lceil \log_2 \frac{t}{1+2k} \rceil} \tag{38}$$

Lemma [7]: The order of $x \in M_t$ is

$$\operatorname{ord} x = \operatorname{ord} f \cdot 2^{\lceil \log_2 t \rceil} . \tag{39}$$

[3]The p-Sylov subgroup of a finite commutative group G is that subgroup of G consisting of all elements whose order is a power of the prime p.

Following a suggestion of Günther [5], we can write (x), the cyclic group in M_t that is generated by x, as the direct product of (x_2) and (x_p), where

$$x_2 \stackrel{\text{def}}{=} x^{\text{ord}f} \tag{40}$$

and

$$x_p \stackrel{\text{def}}{=} x^{2^{\lceil \log_2 t \rceil}}. \tag{41}$$

Therefore, as follows from (39), x_2 has order $2^{\lceil \log_2 t \rceil}$, and thus $x_2 \in P^{(t)}$. With respect to the generators $a_{i,j}$ of $P^{(t)}$, x_2 has the representation

$$x_2 = \prod_{j,k} a_{j,k}^{s_{j,k}}. \tag{42}$$

The order of x_p is ord f and (x_p) is a subgroup of (a_p). Without loss of essential generality, we can assume that

$$x_p = a_p^{\frac{2^{\deg f}-1}{\text{ord}\, f}}. \tag{43}$$

In this special case, the equivalence relation (15) defining our cyclic cosets is

$$u(x) \equiv v(x) \iff \exists\ k,l \text{ such that } v(x) = \left(x_p(x)^k \cdot x_2(x)^l \cdot u(x)\right) \bmod f(x)^t. \tag{44}$$

Since the order of x_2 is $2^{\lceil \log_2 t \rceil}$, i.e., maximal in $P^{(t)}$, it follows from (38) that in (42), at least one of the $a_{j,k}$'s with $k = 0$ must have an odd exponent $s_{j,k}$. Therefore, if we can find a j_0 such that the exponent $s_{j_0,0}$ of $a_{j_0,0}$ in (42) is odd, then we can replace $a_{j_0,0}$ by x_2 as a generator in $P^{(t)}$ and still have a direct product decomposition of M_t. In this case, a set of coset leaders of the cyclic cosets with elements coprime to $f(x)$ is given by

$$c_{i_p,j,k}(x) = a_p(x)^{i_p} \cdot \prod_{(j,k)\neq(j_0,0)} a_{j,k}(x)^{i_{j,k}} \bmod f(x)^t \tag{45}$$

where $0 \le i_p \le \left(2^{\deg f} - 1\right)/\text{ord}f - 1$ and $0 \le i_{j,k} \le n_{j,k} - 1$.

We are now left with determining a value j_0 such that $2 \nmid s_{j_0,0}$. Considering (42) in terms of representatives in $GF(2)[x]$ and taking it modulo $f(x)^2$, we obtain

$$x_2(x) = \prod_{j,k} a_{j,k}(x)^{s_{j,k}} = \prod_j a_{j,0}(x)^{s_{j,0}} = \prod_j a_{j,0}(x)^{s_{j,0}\bmod 2} \bmod f(x)^2. \tag{46}$$

Now the elements $y \in P^{(t)}$, i.e., of order equal to a power of 2, have representatives of the form

$$y(x) = 1 + \sigma(x) \cdot f(x); \tag{47}$$

see [7]. The arithmetic of $P^{(2)}$, when expressed in terms of the polynomials $\sigma(x)$ in (47), is polynomial addition [7], e.g., if $y_i(x) = 1 + \sigma_i(x) \cdot f(x)$ for $i = 1, 2$, then $\overline{y} \stackrel{\text{def}}{=} y_1(x) \cdot y_2(x) \bmod f(x)^2$, written in the form $\overline{y}(x) = 1 + \overline{\sigma}(x) \cdot f(x)$, has $\overline{\sigma}(x) = \sigma_1(x) + \sigma_2(x)$. From $a_{j,0}(x) = 1 + x^j \cdot f(x) = 1 + \sigma_j(x) f(x)$, we see that $\sigma_j(x) = x^j$, i.e., $\sigma_j(x)$ corresponds to a "unit vector." Hence, if the jth coefficient of $\sigma(x)$ in the representation

$$w(x) = 1 + \sigma(x) \cdot f(x) \tag{48}$$

of $w(x) \stackrel{\text{def}}{=} x_2(x) \bmod f(x)^2$ is nonzero, then $s_{j,0}$ is odd, and we can set $j_0 = j$.

III. Optimization of CRC Codes with 24 and 32 Parity Bits

Using an implementation on a high-speed special-purpose processor of the method of Section II for determining the minimum distance of CRC codes, an extensive search for "good" CRC codes has been conducted among several classes of generator polynomials. These classes of generator polynomials consist of polynomials that generate shortened cyclic codes of a given minimum distance $d_{\min,\mathrm{L}}$ in a certain range L of block lengths n. Among two generators in such a class, that one is considered better for which the minimum distance $d_{\min}(n)$ of the length n shortened cyclic code is greater than $d_{\min,\mathrm{L}}$ up to a larger block length n. In the following, we present the best CRC codes that resulted from our search. We also compare these with several recent suggestions of Merkey and Posner [2], as well as with the standard code IEEE-802. In describing the classes of generators in which our search was carried out, we make frequent use of the following lemma.

Lemma [7]: Consider the generator polynomial $g(x) \in GF(2)[x]$ with simple roots and order n such that $(x+1)|g(x)$. Let the minimum distance of its generated CRC code C at block length n be d. Then the polynomial $\overline{g}(x) \stackrel{\text{def}}{=} (x+1) \cdot g(x)$ generates a CRC code $\overline{C}$ with minimum distance $\overline{d} = d$ at block length $n+1$ and $d_{\min} = 4$ at block lengths ranging from $n+2$ to $2n$.

A. CRC Codes with 24 Parity Bits

The code CRC-24/6.1 was obtained starting from the class of CRC codes with 23 parity bits whose generator polynomial factors into $(x+1)$ and two distinct primitive polynomials of degree 11 in $GF(2)[x]$. This class contains, in particular, the primitive BCH codes of length $2^{11} - 1 = 2047$ whose generator polynomials have the factors $(x+1)$, $M_{11}^{(i)}(x)$ and $M_{11}^{(3i)}(x)$,[4] and thus by the BCH bound [9, p. 269], have minimum distance $d_{\min} \ge 6$. Now, generator polynomials of degree p and order n can be partitioned into equivalence classes defined by

$$\overline{g}(x) \equiv g(x) \iff \exists\, t \in \boldsymbol{N} \text{ such that } \overline{g}(x) = \gcd\left(g\left(x^t\right), x^n - 1\right) \text{ where } \gcd(t,n) = 1. \tag{49}$$

Equivalent polynomials, in the sense of (49), generate cyclic codes of the same rate, length, and minimum distance. Hence,

[4] $M_{11}^{(j)}(x) \in GF(2)[x]$ denotes the minimal polynomial of α^j, where α is a primitive element of $GF(2^{11})$.

TABLE I
$d_{\min}(n)$ VERSUS n FOR THE CODES CRC-24/6.1 AND CRC-24/6.2 AS WELL AS THE CODES MP-CRC-24/6.1 AND MP-CRC-24/6.2

$d_{\min}$	n CRC-24/6.1	n CRC-24/6.2	n MP-CRC-24/6.1	n MP-CRC-24/6.2
16		25, ⋯, 26		
14	25			
12	26			
10	27, ⋯, 36	27, ⋯, 41		
8	37, ⋯, 83	42, ⋯, 95	25, ⋯, 68	25, ⋯, 55
6	84, ⋯, 2050	96, ⋯, 2048	69, ⋯, 2048	56, ⋯, 2048
4	2051, ⋯, 4098	2049, ⋯, 4094	2049, ⋯, 4094	2049, ⋯, 4094
2	≥4099	≥4095	≥4095	≥4095

from each class of CRC codes generated by equivalent polynomials, only one member was tested at its cyclic block length. The CRC codes in those classes that meet the BCH bound, in particular the class containing the BCH codes, were then further considered. In order to obtain generators of degree 24, they were first multiplied by $(x + 1)$, which at the same time increases the block length n up to which $d_{\min} \geq 6$ to $n = 2048$. By our lemma and by the equivalence relation (49) of cyclic codes, these polynomials are all generators of CRC codes of the given form that have $d_{\min} \geq 6$ up to $n = 2048$. Hence, having investigated only the equivalence classes (49) of simple-root cyclic codes whose generators of degree 23 factor as indicated, we could determine all generators with $(x + 1)^2$ as a factor and two distinct primitive polynomials of degree 11 that generate shortened cyclic codes that have $d_{\min} \geq 6$ at $n = 2048$. These resulting generators were then compared in terms of the $d_{\min}$ profile of their generated CRC code. The best code thereby found, our code CRC-24/6.1, has $d_{\min} \geq 8$ up to the largest block length, $n = 95$, among all these codes.

The code CRC-24/6.2 is the optimal code in the class of CRC-24 codes generated by polynomials $g(x)$ that are the product of $(x + 1)^2$ with an irreducible polynomial $f(x)$ of degree 22 and order $2^{11} + 1 = 2049$. As follows from Tzeng and Hartmann [10], cyclic codes with generator polynomials of the form $(x + 1) \cdot f(x)$, where $f(x) \in GF(2)[x]$ is irreducible of degree 22 and order 2049, have $d_{\min} = 6$. Therefore, by our lemma, the CRC-24 codes with generators $(x + 1)^2 \cdot f(x)$, $f(x)$ as before, have $d_{\min} = 6$ up to the block length $n = 2050$. Our code CRC-24/6.2, which is the best CRC code in this class, has $d_{\min} \geq 8$ up to the block length $n = 83$, and hence is not quite as good as the code CRC-24/6.1. The minimum distance profiles of the two CRC-24 codes CRC-24/6.1 and CRC-24/6.2 are listed in Table I, which also contains the $d_{\min}$ profiles of two corresponding CRC codes that have been suggested by Merkey and Posner in [2].

The code CRC-24/5.1 was optimized among the codes generated by an irreducible polynomial of degree 24 and order $2^{12} + 1 = 4097$. Codes of this type were first suggested by Zetterberg; see [10]. The Zetterberg codes have minimum distance 5 at their cyclic block length and are optimal at the block length, i.e., there exists no binary (4097, 4073) block code with minimum distance greater than 5, as follows from the Hamming bound. In [7], the optimallity is proven in all cases of binary cyclic codes with $4k$ parity bits and order $2^{2k} + 1$. The $d_{\min}$ profile of the code CRC-24/51. is given in Table II. For reasons of comparison, Table II also contains the $d_{\min}$ profile of the code CRC-24/5.2, which is the best code in the class of CRC-24 codes with a generator polynomial $g(x)$ that factors into two irreducible polynomials of degree 12 and orders $2^{12} - 1 = 4095$ or 4095 or $(2^{12} - 1)/3 = 1365$, respectively. Note that the code CRC-24/5.2 has $d_{\min} = 6$ only up to $n = 217$, whereas CRC-24/5.1 has $d_{\min} = 6$ up to $n = 252$.

TABLE II
$d_{\min}(n)$ VERSUS n FOR THE CODES CRC-24/5.1 AND CRC-24/5.2

$d_{\min}$	CRC-24/5.1	CRC-24/5.2
15	25	25, ⋯, 26
12		27, ⋯, 28
11		29, ⋯, 31
10	26, ⋯, 33	32, ⋯, 33
9		34, ⋯, 35
8		36, ⋯, 41
7	34, ⋯, 37	42, ⋯, 77
6	38, ⋯, 252	78, ⋯, 217
5	253, ⋯, 4097	218, ⋯, 4095
2	≥4098	≥4096

TABLE III
$d_{\min}(n)$ VERSUS n FOR THE CODE CRC-24/4

$d_{\min}$	CRC-24/4
12	25, ⋯, 30
10	31, ⋯, 36
8	37, ⋯, 61
6	62, ⋯, 846
4	847, ⋯, 8 388 607
2	≥8 388 608

The code CRC-24/4 is the CRC code that has $d_{\min} = 6$ up to the largest block length among all generator polynomials of the form $(x + 1) \cdot p(x)$ where $p(x)$ is a primitive polynomial of degree 23. The $d_{\min}$ profile of this code is given in Table III. CRC codes with a generator polynomial $g(x)$ of the form $g(x) = (x + 1) \cdot q(x)$, where $q(x)$ is an irreducible polynomial of order $(2^{23} - 1)/47 = 178\,481$, were found to have $d_{\min}(n) = 6$ only up to $n = 763$. These codes are therefore inferior to CRC-24/4.

Table IV lists the generator polynomials of CRC-24 codes treated in this paper, together with their most important parameters. In Fig. 1, we give typical values (see Note 1

TABLE IV
GENERATOR POLYNOMIALS $g(x)$ OF THE CRC-24 CODES

CRC Code	n_c	$g(x)$	Factorization of $g(x)$
CRC-24/6.1	4094	127 266 713	$5711 \cdot 5657 \cdot 3 \cdot 3 = M_{11}^{(49)}(x) \cdot M_{11}^{(373)}(x) \cdot \left(M_{11}^{(0)}(x)\right)^2$
CRC-24/6.2	4098	136 600 675	$23\,325\,331 \cdot 3 \cdot 3 = M_{22}^{(354\,131)}(x) \cdot \left(M_{22}^{(0)}(x)\right)^2$
MP-CRC-24/6.1	4094	114 430 011	$7745 \cdot 7371 \cdot 3 \cdot 3 = M_{11}^{(163)}(x) \cdot M_{11}^{(489)}(x) \cdot \left(M_{11}^{(0)}(x)\right)^2$
MP-CRC-24/6.2	4094	120 013 007	$5253 \cdot 5001 \cdot 3 \cdot 3 = M_{11}^{(341)}(x) \cdot M_{11}^{(1023)}(x) \cdot \left(M_{11}^{(0)}(x)\right)^2$
CRC-24/5.1	4097	114 377 431	$114\,377\,431 = M_{24}^{(1\,363\,635)}(x)$
CRC-24/5.2	4095	126 742 365	$14\,227 \cdot 14\,667 = M_{12}^{(101)}(x) \cdot M_{12}^{(1527)}(x)$
CRC-24/4	8 388 607	114 505 543	$73\,474\,441 \cdot 3 = M_{23}^{(288\,373)}(x) \cdot M_{23}^{(0)}(x)$

The polynomials are given in octal notation, e.g., octal 13 stands for binary 001 011 which is the polyomial $x^3 + x + 1$. $M_m^{(i)}(x)$ denotes the minimal polynomial of α^i, where α is the primitive element of $GF(2^m)$ used in Appendix C of [9]. n_c is the length of the cyclic code generated by $g(x)$.

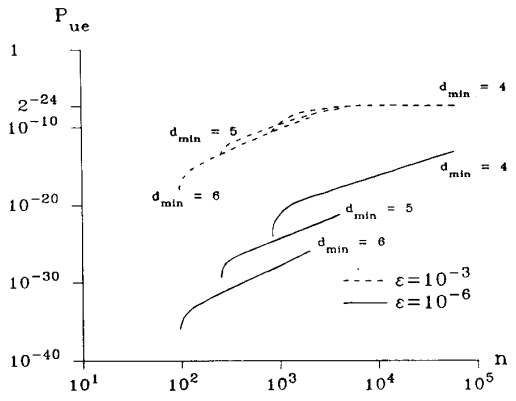

Fig. 1. $P_{ue}(n)$ versus n for CRC-24 codes with various minimum distances $d_{\min}$ on the BSC's with $\epsilon = 10^{-3}$ and 10^{-6}.

below) of undetectable-error probabilities P_{ue} of CRC-24 codes with various minimum distances at all block lengths. Thereby, P_{ue} is given on a quite noisy BSC (crossover probability $\epsilon = 10^{-3}$) and on a low-noise BSC (crossover probability $\epsilon = 10^{-6}$).

B. CRC Codes with 32 Parity Bits

The code CRC-32/8 was found to be the best CRC-32 code of the class of codes whose generator factors into $(x + 1)^2$ and three distinct irreducible polynomials of degree 10, two of order $2^{10} - 1 = 1023$, and one of order 341. The generators of the codes equivalent to the primitive BCH codes of length 1023 and $d_{\min} = 8$, when multiplied with $(x+1)$, yield CRC-32 codes in this class. Hence, by our lemma, this class of generator polynomials of degree 32 contains CRC-32 codes that satisfy $d_{\min}(n) \geq 8$ up to the block length $n = 1024$. The $d_{\min}$ profile of the code CRC-32/8 is listed in Table V, together with the $d_{\min}$ profiles of the corresponding CRC-32 codes suggested by Merkey and Posner [2].

TABLE V
$d_{\min}(n)$ VERSUS n FOR THE CODES CRC-32/8, MP-CRC-32/8.1, MP-CRC-32/8.2

$d_{\min}$	n CRC-32/8	n MP-CRC-32/8.1	n MP-CRC-32/8.2
14	$33, \cdots, 44$		
12	$45, \cdots, 48$		
10	$49, \cdots, 98$	$33, \cdots, 78$	$33, \cdots, 79$
8	$99, \cdots, 1024$	$79, \cdots, 1023$	$80, \cdots, 1023$
4	$1025, \cdots, 2046$		
2	≥ 2047	≥ 1024	≥ 1024

The code CRC32/6 is the best code in the class of CRC-32 codes whose generator factors into $(x + 1)^2$ and two distinct primitive polynomials of degree 15, and hence of order $2^{15} - 1 = 32\,767$. Similarly to the class within which CRC-32/8 was selected, this class contains CRC codes whose generators are $(x + 1)$ times the generator of primitive BCH codes of length 32 767 and $d_{\min} = 6$. The $d_{\min}$ profile of the code CRC-32/6 is contained in Table VI. The codes CRC-32/8 and CRC-32/6 were both found in a way similar to the search for CRC-24/6.1, i.e., by first finding all equivalence classes (49) of simple-rood generators of degree 31 whose $d_{\min}$ at the cyclic block length was equal to $d_{\min}$ of the BCH code of the same rate and length, and then optimizing the $d_{\min}$ profile of the resulting augmented degree-32 generators.

The code CRC-32/5.1 is the optimum Zetterberg code with 32 parity bits, i.e., it was determined to have $d_{\min} = 6$ up to the largest possible block length among all codes generated by an irreducible polynomial of degree 32 and order $2^{16} + 1 = 65\,537$. Its minimum distance profile is listed in Table VII. This table also contains the code CRC-32/5.2, which is the best code in the class of CRC-32 codes whose generator polynomial $g(x)$ has two irreducible factors of orders $2^{16} - 1 = 65\,535$ and 65 535 or $(2^{16} - 1)/3 = 21\,845$, respectively. The code CRC-32/5.2 has $d_{\min} = 6$ up to slightly shorter block lengths than CRC-32/5.1, i.e., up to $n = 1029$ as opposed to $n = 1092$

TABLE VI
$d_{\min}(n)$ VERSUS n FOR THE CODE CRC-32/6

$d_{\min}$	CRC-32/6
20	33
18	34, ⋯, 35
16	36
14	37
12	38, ⋯, 43
10	44, ⋯, 56
8	57, ⋯, 306
6	307, ⋯, 32 768
4	32 769, ⋯, 65 534
2	≥65 535

TABLE VII
$d_{\min}(n)$ VERSUS n FOR THE CODES CRC-32/5.1 AND CRC-32/5.2

$d_{\min}$	IEEE-802	CRC-32/5.2
17		33, ⋯, 34
15	33, ⋯, 35	
14		35
13		36, ⋯, 38
12	36, ⋯, 49	
11	50, ⋯, 53	39, ⋯, 52
10	54, ⋯, 59	53, ⋯, 68
9		69, ⋯, 80
8	60, ⋯, 90	81, ⋯, 110
7	91, ⋯, 113	111, ⋯, 266
6	114, ⋯, 1092	267, ⋯, 1029
5	1093, ⋯, 65 537	1030, ⋯, 65 535
2	≥65 538	≥65 536

for the code CRC-32/5.1. Note that also for $p = 16$ [8] and $p = 24$, the best Zetterberg code is slightly better than the best code in the corresponding class of CRC codes whose generator factors into two irreducible polynomials of the same degree.

The code CRC-32/4 is the best code that resulted from an optimization of the $d_{\min}$ profile of some 47 000 CRC codes with a generator that factors into $(x + 1)$ times a primitive polynomial of degree 31. Since there are $2 \cdot (2^{30} - 1)/31$ primitive polynomials of degree 31 in $GF(2)[x]$, it was impossible to test all CRC codes of this class. Our CRC-32/4 code has $d_{\min} = 6$ up to the block length $n = 5275$. Its $d_{\min}$ profile is given in Table VIII.

The standard code IEEE-802, whose generator polynomial was suggested by Hammond *et al.* [11] and which is used in the IEEE 802.3, 4, 5, and 6 protocols (Ethernet, Token Passing Bus, Token Ring, and the Metropolitan Area Network) as well as in the FDDI protocol, was also investigated for reasons of comparison. Its generator is a primitive polynomial of degree 32, and thus has $d_{\min} = 3$ up to $n_c = 2^{32} - 1 = 4\,294\,967\,295 \approx 4.3 \cdot 10^9$, which safely exceeds any frame length ever to be encountered in practice. Its $d_{\min}$ profile is contained in Table VIII, and its P_{ue} performance on a quite noisy BSC and on a low-noise BSC is given in Fig. 2. The IEEE-802 code was investigated earlier by Fujiwara *et al.* [12].

Table IX contains a list of all CRC codes with 32 parity bits that are treated in this paper, and in Fig. 3 we give typical values (see Note 1 below) of undetectable-error probabilities P_{ue} of CRC-32 codes with various $d_{\min}$ at all block lengths.

TABLE VIII
$d_{\min}(n)$ VERSUS n FOR THE CODES CRC-32/4 AND THE STANDARD IEEE-802

$d_{\min}$	IEEE-802	CRC-32/4
18		33
16		34, ⋯, 38
15	33, ⋯, 42	
14		39, ⋯, 40
12	43, ⋯, 44	41, ⋯, 52
11	45, ⋯, 53	
10	54, ⋯, 66	53, ⋯, 79
9	67, ⋯, 89	
8	90, ⋯, 123	80, ⋯, 209
7	124, ⋯, 203	
6	204, ⋯, 300	210, ⋯, 5275
5	301, ⋯, 3006	
4	3007, ⋯, x[a]	5276, ⋯, $2^{31} - 1$
3	$x + 1, \cdots, 2^{32} - 1$	
2	$\geq 2^{32}$	$\geq 2^{31}$

[a] $x \geq 64\,000$.

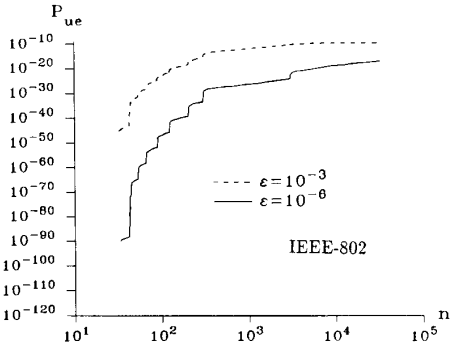

Fig. 2. $P_{ue}(n)$ versus n for the IEEE-802 code on the BSC's with $\epsilon = 10^{-3}$ and 10^{-6}.

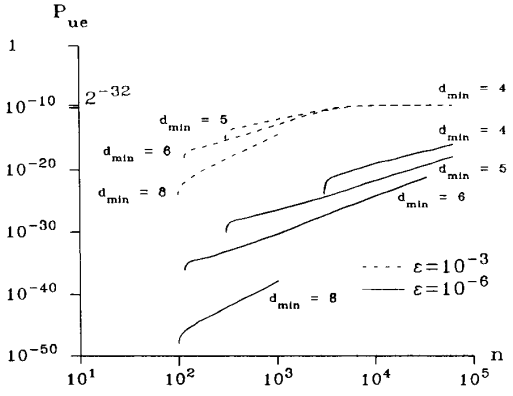

Fig. 3. $P_{ue}(n)$ versus n for CRC-32 codes with various minimum distances $d_{\min}$ on the BSC's with $\epsilon = 10^{-3}$ and 10^{-6}.

The values of P_{ue} are given for a quite noisy BSC (crossover probability $\epsilon = 10^{-3}$) and a low-noise BSC ($\epsilon = 10^{-6}$).

Note 1: In Figs. 1–3, we have presented $P_{ue}(n)$ as computed via the dual code's weight distribution according to [13]. The values for P_{ue} have therefore not been obtained by any approximation. However, it should be emphasized that

TABLE IX
GENERATOR POLYNOMIALS $g(x)$ OF THE CRC-32 CODES

CRC Code	n_c	$g(x)$	Factorization of $g(x)$
CRC-32/8	2046	1F1922815	$465 \cdot 557 \cdot 787 \cdot 3 \cdot 3 = M_{10}^{(95)}(x) \cdot M_{10}^{(239)}(x) \cdot M_{10}^{(47)}(x) \cdot \left(M_{10}^{(0)}(x)\right)^2$
MP-CRC-32/8.1	1023	1404098E2	$531 \cdot 4C9 \cdot 747 \cdot 3 \cdot 2 = M_{10}^{(29)}(x) \cdot M_{10}^{(87)}(x) \cdot M_{10}^{(73)}(x) \cdot M_{10}^{(0)}(x) \cdot x$
MP-CR-32/8.2	1023	10884C512	$42D \cdot 463 \cdot 7B1 \cdot 3 \cdot 2 = M_{10}^{(101)}(x) \cdot M_{10}^{(189)}(x) \cdot M_{10}^{(191)}(x) \cdot M_{10}^{(0)}(x) \cdot x$
CRC-32/6	65 534	1F6ACFB13	$8011 \cdot C85F \cdot 3 \cdot 3 = M_{15}^{(4779)}(x) \cdot M_{15}^{(1093)}(x) \cdot \left(M_{15}^{(0)}(x)\right)^2$
CRC-32/5.1	65 537	1A833982B	$1A833982B = M_{32}^{(127\,989\,855)}(x)$
CRC-32/5.2	65 535	1572D7285	$15BAB \cdot 10FEB = M_{16}^{(1533)}(x) \cdot M_{16}^{(273)}(x)$
CRC-32/4	$2^{31} - 1$	11EDC6F41	$F5B4253F \cdot 3 = M_{31}^{(90\,577\,445)}(x) \cdot M_{31}^{(0)}(x)$
IEEE-802	$2^{32} - 1$	104C11DB7	$104C11DB7 = M_{32}^{(7)}(x)$

The polynomials are given in hexadecimal notation, e.g., hex 3B stands for binary 0011 1011 which is the polyomial $x^5 + x^4 + x^3 + x + 1$. $M_m^{(i)}(x)$ denotes the minimal polynomial of α^i, where α is the primitive element of $GF(2^m)$ used in [9, Appendix C]. n_c is the length of the cyclic code generated by $g(x)$.

in Figs. 1 and 3, we present only "typical values" of P_{ue}: for fixed values of ϵ ($\epsilon = 10^{-3}$ and $\epsilon = 10^{-6}$), these figures show P_{ue} as a function of the block length n *and* the minimum distance $d_{\min}$ for CRC codes with 24 and 32 parity bits, respectively. The presented data on P_{ue} thus do not refer to any particular code. We have chosen this representation in order to save space. Hence, for specific codes, these data lack certain details, i.e., the behavior of P_{ue} at the "jump" points of $d_{\min}(n)$. Our investigations have shown that at a "jump" point of a given CRC code, its P_{ue} value jumps from the lower to the higher level of P_{ue} displayed in Figs. 1 and 3 in a saturation-like manner, as shown in Fig. 2 for IEEE-802.

Note 2: Since the results on $d_{\min}$ of the various CRC codes investigated for this paper have been obtained by a fast special-purpose processor, they are comparatively difficult to reproduce. It is therefore important to note that our results on $d_{\min}$ can be seen to be consistent in all of those cases where the minimum distance of a shortened cyclic code is known from a different source [7]. Notably, our results for the IEEE-802 code, whose $d_{\min}$ profile cannot be determined by theoretical means, are consistent with the results presented in [12] that were computed by a different method, which is particularly suited to compute the minimum distance of a CRC code, but does not offer any values of P_{ue}.

IV. CONCLUDING REMARKS

Our investigations of CRC codes have yielded several interesting new codes with 24 and 32 parity bits. In the case of 24 parity bits, where so far there has been no standardization, our suggestions should provide a solid basis for future standards. We have also found CRC codes with 32 parity bits that improve upon earlier suggestions, as well as upon the standard IEEE-802.

In view of the sometimes heuristic approaches that underlay previous work in defining CRC codes for error detection, the superiority of our codes is not surprising. On the other hand, however, it should be observed that it is mainly on low-noise channels, for which the standards on the whole perform satisfactorily, that our codes improve upon these standards. The improvements over the standard codes are due to increases of the minimum distance at various block lengths. On very noisy channels, e.g., during error bursts, the error-detecting capability depends solely on the number of parity bits, which is not subject to optimization. We also mention that most of the new CRC codes that we have presented in this work have been investigated for properness at block lengths up to approximately $n = 2^{15} - 1$. A CRC code is said to be proper at a certain block length n if on BSC's the probability P_{ue} of undetectable errors increases monotonically as a function of the crossover probability $\epsilon \in [0, 0.5]$. Except for the codes CRC-24/5.2 and CRC-32/5.2, all CRC-24 and CRC-32 codes were investigated for properness. The codes were checked at those block lengths not exceeding $n = 2^{15} - 1$ that appear in the respective tables of Section III, showing $d_{\min}$ as a function of n. Except for the code CRC-24/5.1 at $n = 37$ and $n = 38$ and the code CRC-32/5.1 at $n = 113$ and $n = 114$, where a slight improperness was detected, as well as for the code CRC-32/8, where improperness was detected at $n = 2046$, all of these checks revealed that P_{ue} increases monotonically with ϵ for the respective CRC codes and block lengths.

Whether one should replace an existing standard code by one of our codes is both a technical and economical question. On new and isolated networks that are not likely ever to be coupled to existing nets, it is certainly reasonable to choose one of our codes. Moreover, should any future specifications require a number of parity bits different from 24 or 32 and which does not (substantially) exceed 32, then

our algorithm together with its implementation on our special-purpose processor appears as a reasonable means for obtaining a good solution.

ACKNOWLEDGMENT

The authors would like to thank the ETH, notably Prof. J. L. Massey, for supporting this project as well as G. Funk of ABB for posing the question of finding good CRC codes and Dr. Ch. Günther of ABB for pointing out the possibility of efficient generation of the codewords of the dual code of a shortened cyclic code by using two LFSR's. Finally, they are indebted to the reviewers for constructive criticism.

REFERENCES

[1] T. Fujiwara, T. Kasami, A. Kitai, and S. Lin, "On the undetected error probability for shortened Hamming codes," *IEEE Trans. Commun.*, vol. COM-33, pp. 570–574, June 1985.
[2] P. Merkey and E. C. Posner, "Optimum cyclic redundancy codes for noisy channels," *IEEE Trans. Inform. Theory,* vol. IT-30, pp. 865–867, Nov. 1984.
[3] F. J. MacWilliams, "A theorem on the distribution of weights in a systematic code," *Bell Syst. Tech. J.,* vol. 42, pp. 79–94, 1963.
[4] S. C. Chang and J. K. Wolf, "A simple derivation of the MacWilliams identity for linear codes," *IEEE Trans. Inform. Theory,* vol. IT-26, pp. 476–477, July 1980.
[5] Ch. Günther, oral communication.
[6] B. L. van der Waerden, *Algebra II, vol. 5.* Auflage: Springer-Verlag, 1967.
[7] G. Castagnoli, "On the minimum distance of long cyclic codes and cyclic redundancy-check codes," Ph.D. dissertation Inst. for Signal and Inform. Processing, Swiss Federal Inst. Technol., Zurich, 1989.
[8] G. Castagnoli, J. Ganz, and P. Graber, "Optimum cyclic redundancy-check codes with 16-bit redundancy," *IEEE Trans. Commun.,* vol. 38, pp. 111–114, Jan. 1990.
[9] W. W. Peterson and E. J. Weldon, Jr., *Error-Correcting Codes,* 2nd ed. Cambridge, MA: M.I.T. Press, 1972.
[10] K. K. Tzeng and C. R. P. Hartmann, "On the minimum distance of certain reversible cyclic codes," *IEEE Trans. Inform. Theory,* vol. IT-16, pp. 644–646, Sept. 1970.
[11] J. L. Hammond, J. E. Brown and S. S. Liu, "Development of a transmission error model and an error-control model," Tech. Rep., Rome Air Develop. Cen., RADC-TR-75-138, May 1975.
[12] T. Fujiwara, T. Kasami, and S. Lin, "Error detecting capabilities of the shortened Hamming codes adopted for error detection in IEEE Standard 802.3," *IEEE Trans. Commun.,* vol. 37, pp. 986–989, Sept. 1989.
[13] J. K. Wolf, A. M. Michelson, and A. H. Levesque, "On the probability of undetected error for linear block codes," *IEEE Trans. Commun.,* vol. COM-30, pp. 317–324, Feb. 1982.
[14] W. W. Peterson and D. T. Brown, "Cyclic codes for error detection," *Proc. IRE,* vol. 49, pp. 228–235, Jan. 1961.
[15] C. Leung, E. R. Barnes, and D. U. Friedman, "On some properties of the undetected-error probability of linear codes," *IEEE Trans. Inform. Theory,* vol. IT-25, pp. 110–112, Jan. 1979.
[16] K. A. Witzke and C. Leung, "A comparison of some error detecting CRC-code standards," *IEEE Trans. Commun.,* vol. COM-33, pp. 996–998, Sept. 1985.

Guy Castagnoli was born in Zurich, Switzerland, on December 8, 1960. He received the diploma in mathematics and the Ph.D. degree in electrical engineering in 1989, both from the Swiss Federal Institute of Technology.

He is currently with Vita Life Insurance Company as an actuary.

Stefan Bräuer was born in St. Gall, Switzerland, on July 6, 1963. He received the diploma in electrical engineering from the Swiss Federal Institute of Technology in 1989.

He is currently with Staefa Control System as a Development Engineer in HVAC control.

Martin Herrmann was born in Hetzeldorf, Siebenbürgen/Rumania, on August 13, 1962. He received the diploma in electrical engineering from the Swiss Federal Institute of Technology in 1989.

He is presently with the Reliability Laboratory of the Swiss Federal Institute of Technology, where he is working on his Ph.D. dissertation in the area of EPROM reliability.