Copyright Notice

More on Hauck's warning.

In EWD525 "On a warning from E.A.Hauck" I mentioned without proof that with $n=2^m$ bit there exist 2^{n-m-1} different messages --I called them "codes", but that is an unusual terminology for which I apologize-- , such that any two different messages differ in at least four bit positions, thus allowing correction of one-bit errors and detection of two-bit errors. Since then I have been shown a proof of that theorem; I report that proof because it is so nice, and because it gives some further insights.

For the sake of brevity I shall demonstrate the theorem for $16=2^4$ bits (in a way which is readily generalized for other values of m). We consider 16 bits numbered from 0 through 15 , writing their index in binary:

$$d_{0000}, d_{0001}, d_{0010}, d_{0011}, \ldots, d_{1111}$$

With "xxx1" we denote the set of odd indices, with "xx1x" the set {0010, 0011, 0110, 0111, 1010, 1011, 1110, 1111}, in general the set obtained by all possible substitutions of a 0 or a 1 at a place marked "x", and define

$$h0 = \text{parity}(d_{xxx1}) \;,\; h1 = \text{parity}(d_{xx1x}) \;,\; h2 = \text{parity}(d_{x1xx}) \;,\; h3 = \text{parity}(d_{1xxx})$$

where the function "parity" is = 0 if among the (8) bits with an index from the indicated set, the number of 1's is even, and = 1 if it is odd. Further we introduce $h = \text{parity}(d_{xxxx})$, which is just the sum of all the 16 bits modulo 2.

The 2^{11} correct messages are then characterized by the equations

$$h0 = h1 = h2 = h3 = h = 0 .$$

Note. The above equations have indeed 2^{11} different solutions: the 11 bits d_3, d_5, d_6, d_7, d_9, d_{10}, d_{11}, d_{12}, d_{13}, d_{14}, and d_{15} can be chosen freely, we then solve h0 for d_1, h1 for d_2, h2 for d_4, and h3 for d_8, and finally h for d_0 .

We now denote by "a" the binary number formed by "h3 h2 h1 h0" and observe:

0) for each correct message we have

$$h = 0, \; a = 0$$

1) for a one-bit error at bit position i we have

$$h = 1, \; a = i$$

2) for a two-bit error at bit positions i and j

$$h = 0, \; a = \text{the bit-wise sum of } i \text{ and } j$$

(because $i \neq j$, we conclude that $a \neq 0$, thereby distinguishing this case from a correct message)

3) for a three-bit error at positions i, j, and k

$$h = 1, \; a = \text{the bit-wise sum of } i, j, \text{ and } k.$$

4) for a four-bit error at positions i, j, k, and l

$$h = 0, \; a = \text{the bit-wise sum of } i, j, k, \text{ and } l.$$

etc.

Under the assumption that one- and two-bit errors are the <u>only</u> errors that can occur, the rules are

$h = 0$ and $a = 0$: accept the bit sequence as given

$h = 1$: invert bit d_a

$h = 0$ and $a \neq 0$: alarm, as two-bit error has been detected.

From the above, however, we see that all errors in 3, 5, 7, ... bits will then erroneously be interpreted as one-bit errors, i.e. in those cases our error correction indeed increases the probability of a wrong result being produced as if it were a correct one. The above gives a clear demonstration of the possible "harmfulness" of error correction alluded to in EWD525's last paragraph. Hence this note.

Plataanstraat 5
4565 - NUENEN
The Netherlands

prof.dr.Edsger W.Dijkstra
Burroughs Research Fellow