Nirakar Shrestha Assignment 11

1)

Data bits with value 1 are in bit positions 12, 11, 5, 4, 2, and 1:

Poistion 12 11 10 9 8 7 6 5 4 3 2 1

Bits D8 D7 D6 D5 C8 D4 D3 D2 C4 D1 C2 C1

Block 1 1 0 0 0 0 1 0

Codes 1100 1011 0101

The check bits are in bit numbers 8, 4, 2, and 1.

Check bit 8 calculated by values in bit numbers: 12, 11, 10 and 9

Check bit 4 calculated by values in bit numbers: 12, 7, 6, and 5

Check bit 2 calculated by values in bit numbers: 11, 10, 7, 6 and 3

Check bit 1 calculated by values in bit numbers: 11, 9, 7, 5 and 3

Thus, the check bits are: 0 0 1 0.de, k par

Due to the limited redundancy that Hamming codes add to the data, they can only detect and correct errors when the error rate is low. This is the case in computer memory (ECC memory), where bit errors are extremely rare and Hamming codes are widely used. Extended Hamming codes achieve a Hamming distance of four, which allows the decoder to distinguish between when at most one one-bit error occurs and when any two-bit errors occur.

In the Hamming code, k parity bits are added to an n-bit data word, forming a new word of n k bits. The bit positions are numbered in sequence from 1 to n k. Those positions numbered with powers of two are reserved for the parity bits. The remaining bits are the data bits. The code can be used with words of any length.

2)

Check bits needed: 2k-1 M+K, where M = data bits and k = check bits

If K = 10, and M = 1024, then

2k-1 M+K or, 210-1 1024+10 or 1023 < 1034

Thus, it does not satisfy the condition when K=10.

If K = 11, and M = 1024, then

2k-1 M+K or, 211-1 1024+10 or 2047 > 1034

Thus, to satisfy the required condition k=11, or 11 check bits re needed.

Therefore, k = 11 is the minimum no. of check bits  that satisfies the requirement.

3)

If there are 2 bits of error, then a 1-bit parity scheme will not detect any errors, since the parity will match the data with two errors. (Actually, a 1-bit parity scheme can detect any odd number of errors; however, the probability of having 3 errors is much lower than the probability of having two, so, in practice, a 1-bit parity code is limited to detecting a single bit of error.) Of course, a parity code cannot correct errors, which Hamming wanted to do as well as to detect them. If we used a code that had a minimum distance of 3, then any single bit error would be closer to the correct pattern than to any other valid pattern. He came up with an easy to understand the mapping of data into a distance 3 code that we call Hamming Error Correction Code (ECC) in his honor. We use extra parity bits to allow the position identification of a single error. Here are the steps to calculate Hamming ECC

a. Start numbering bits from 1 on the left, as opposed to the traditional

The numbering of the rightmost bit is 0.

b. Mark all bit positions that are powers of 2 as parity bits (positions 1, 2, 4, 8,

16, …).

c. All other bit positions are used for data bits (positions 3, 5, 6, 7, 9, 10, 11, 12,

13, 14, 15, …).

d. The position of parity bit determines the sequence of data bits that it checks

(Figure 5.24 shows this coverage graphically) is:

■ Bit 1 (0001two) checks bits (1,3,5,7,9,11,...), which are bits where rightmost

bit of address is 1 (0001two, 0011two, 0101two, 0111two, 1001two, 1011two,…).

■ Bit 2 (0010two) checks bits (2,3,6,7,10,11,14,15,…), which are the bits

where the second bit to the right in the address is 1.

■ Bit 4 (0100two) checks bits (4–7, 12–15, 20–23,...) , which are the bits where

the third bit to the right in the address is 1.

■ Bit 8 (1000two) checks bits (8–15, 24–31, 40–47,...), which are the bits

where the fourth bit to the right in the address is 1.

Note that each data bit is covered by two or more parity bits.

e. Set parity bits to create even parity for each group.

