Maciej Ciepiela, 347677, Sieci Komputerowe, Ćwiczenia 2

1. *W 10-Mbitowym Ethernecie sygnał rozchodzi się z prędkością 10^8 m/s. Standard ustala, że maksymalna odległość między dwoma komputerami może wynosić co najwyżej 2,5 km. Oblicz, jaka jest minimalna długość ramki (wraz z nagłówkami).*

Dane:

- Przepływność: 10^7 bitów/s
- Prędkość propagacji: $v = 10^8$ m/s
- Maksymalna odległość: $L = 2500$ m

Czas propagacji:

$t_p = L/v = 2500 / 10^8 = 25 * 10^{-6}$ s

Minimalna długość ramki (aby w czasie transmisji zdążyć rozprzestrzenić się po całym kablu):

minimalna długość ramki = $10^7 * 25 * 10^{-6} * 2$ = **500 bitów**

razy 2, bo z wykładu aby wykryć kolizję podczas transmisji, ramka musi trwać co najmniej 2× czas propagacji.

2. *Rozważmy rundowy protokół Aloha we współdzielonym kanale, tj. w każdej rundzie każdy z n uczestników usiłuje wysłać ramkę z prawdopodobieństwem p. Jakie jest prawdopodobieństwo P(p,n), że jednej stacji uda się nadać (tj. że nie wystąpi kolizja)? Pokaż, że P(p,n) jest maksymalizowane dla p =1/n. Ile wynosi $\lim_{n\to\infty} P(1/n,n)$?*

Protokół ALOHA wysyła bez uprzedniego sprawdzania czy ktoś nadaje i sprawdzania czy będzie kolizja. Mamy dużo wysyłających nadających rzadko, a pakiety mają te same długości. Pakiet wybieramy z prawdopodobieństwem p.
Skoro każdy wysyła z prawdopodobieństwem p, to prawdopodobieństwo, że dokładnie jedna stacja wyśle ramkę wynosi:

$P(p, n) = n * p * (1 - p)^{n-1}$

Licząc pochodną po ***p***:

$P'(p) = n * (1 - p)^{n-1} - (n - 1) * n * (1 - p)^{n-2} * p$

Miejsce zerowe: $p = 1/n$

Zatem $P(p, n)$ dla $p = 1/n$ przyjmuje wartość maksymalną.

$\lim_{n\to\infty} P(1/n,n) = (1 - 1/n)^{n-1} = (1 - 1/n)^{n} * (1 - 1/n)^{-1} = 1/e * 1 = 1/e$

3. Jaka suma kontrolna CRC zostanie dołączona do wiadomości 1010 przy założeniu że CRC używa wielomianu $x^2 + x + 1$? A jaka jeśli używa wielomianu $x^7 + 1$?

a) Na podstawie wielomianu wiemy, że g = 111, stopień = 2

Wykonujemy dzielenie w modulo 2 (XOR), wynik jest nieistotny, interesuje nas reszta (powinna mieć stopień równy stopniowi CRC):

```
 1110
101000 : 111
111
 100
 111
  110
  111
  010
```

Zatem do wiadomości zostanie dołączona suma kontrolna równa 10 (reszta z obliczeń), cała wiadomość: 101010

b) Na podstawie wielomianu wiemy, że g = 10000001, stopień = 7

Wykonujemy obliczenia analogicznie jak w poprzednim przykładzie:

```
10100000000 : 10000001
10000001
0010000100
   10000001
   00001010
```

Reszta(suma kontrolna): 1010, Cała wiadomość: 10101010

***4.** Pokaż, że CRC-1, czyli 1-bitowa suma obliczana na podstawie wielomianu G(x) = x + 1, działa identycznie jak bit parzystości.*

Chcemy pokazać, że reszta z dzielenia $x * M(x)$ przez G(x) zwróci bit parzystości M(x), czyli jest równa 0 gdy liczba jedynek jest parzysta oraz 1 w p.p.

Podstawa:
Jeśli m = 0: liczba 1 jest parzysta, a reszta z dzielenia 0 przez x + 1 to 0,
Jeśli m = 1: liczba jedynek nieparzysta, reszta z dzielenia x przez x + 1 to 1.

Krok:
Załóżmy, że dla dowolnego ciągu m długości n reszta z dzielenia przez x+1 jest równa liczbie jedynek w m modulo 2.
Rozważmy ciąg m’ = mb, gdzie m jest opisana jak powyżej a b to ostatni bit równy 1 lub 0.
reszta(m’) = (reszta(m) + b) mod 2, gdzie reszta oznacza resztę z dzielenia w modulo 2,
z zał. ind.:
reszta(m’) = (liczba jedynek w m + b) mod 2 = liczba jedynek w m’ mod 2. ■

***5.** Załóżmy, że wielomian G(x) stopnia n stosowany w CRC zawiera składnik x^0. Pokaż, że jeśli wybierzemy dowolny odcinek długości n z wiadomości i dowolnie go zmodyfikujemy (zmienimy dowolną niezerową liczbę bitów w nim), to zostanie to wykryte. Czy taka własność zachodzi, jeśli G(x) nie zawiera składnika równego x^0?*

CRC: $G(x) = a_n x^n + \ldots + 1$, $\deg(G(x)) = n$
Załóżmy nie wprost że $G(x) \mid E(x)$, wtedy $\exists Q(x)$: $E(x)=G(x)\cdot Q(x)$, ale:
$\deg(E(x)) < n$ i $\deg(G(x)) = n$, a $\deg(Q(x)) = \deg(E(x)) - \deg(G(x)) < 0$, sprzeczność
zatem $G(x) \nmid E(x)$ wykrywa błąd.

Weźmy dowolne G(x) bez składnika x^0: $G(x) = x^2$,
dowolną wiadomość: m = 10,
Z obliczeń wiemy że wiadomość do wysłania to będzie 1000 ($B(x) = x^3$).

Teraz wprowadzamy błąd: 1100, czyli $B'(x) = x^3 + x^2$, $E(x) = x^2$
sprawdźmy czy $G(x) \mid B'(x)$ oraz $G(x) \mid E(x)$
$x^3 + x^2 = x^2 * (x + 1)$, $x^2 = x^2 * 1$
zatem dzieli, a z tego wynika, że G(x) nie zawierające składnika x^0 może nie wykryć błędu.

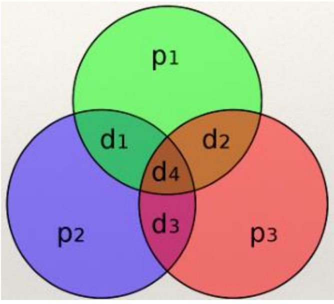
***8.** Pokaż, że kodowanie Hamming(7,4) umożliwia skorygowanie jednego przekłamanego bitu.*
Z wykładu wiemy, że wystarczy pokazać, że odległość Hamminga >= 3,
czyli możemy pokazać, że nie jest równa 1 ani 2.
Odległość Hamminga między dwoma kodami to ilość zamian potrzebna do uzyskania z jednego kodu drugiego (ilość jedynek w słowie a XOR b).
Weźmy macierz generującą G kodu (7,4) oraz macierz kontroli parzystości H.

$$\mathbf{G}^{\mathbf{T}} := \begin{pmatrix} 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \mathbf{H} := \begin{pmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}$$

Korzystając z własności macierzy kontroli H:
1) Każda kolumna w H jest niezerowa, bo 0 oznaczałoby, że 1-bitowy błąd w tej pozycji nie wpływałby na syndrom => byłby niewykrywalny.
2) Żadne dwie kolumny nie są równe, bo dwie identyczne kolumny oznaczałyby, że błąd w tych pozycjach byłby niewidoczny (ich syndromy się znoszą).
(syndrom to $s = H * r^T$, wynik mnożenia macierzy kontrolnej i otrzymanego słowa)
Zatem nie istnieją wektory kodowe o wadze 1 (bo kolumna ≠ 0) oraz o wadze 2 (bo każda para kolumn jest liniowo niezależna — nie mogą się znosić)

Natomiast istnieją 3 kolumny, które będą liniowo zależne co pokazuje możliwość rozpoznania 3-bitowego błędu.

Także patrząc na diagram z wykładu, można zauważyć, że zmiana jakiegokolwiek z bitów danych(d1,d2,d3,d4) wymusza zmianę 2 innych bitów parzystości, np. dla prawidłowego kodowania, zmiana d1 dodatkowo zmienia p1 i p2, czyli każde uszkodzone słowo leży co najmniej 3 kroki od innych słów kodowych.



To zapewnia, że minimalna odległość Hamminga wynosi 3, a co za tym idzie, kodowanie umożliwia skorygowanie jednego błędnego bitu.