ZESZYTY NAUKOWE POLITECHNIKI ŚLĄSKIEJ 1966
Seria: AUTOMATYKA z. 7 Nr kol. 167

ZYGMUNT SZWAJA
Katedra Automatyki
i Elektroniki Przemysłowej
Politechniki Poznańskiej

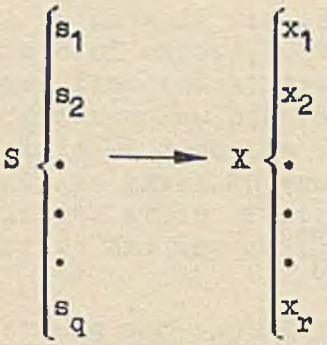
PIERŚCIENIE WIELOMIANÓW I CIAŁA GALOIS W ZASTOSOWANIU DO SYNTEZY KODÓW CYKLICZNYCH

Streszczenie. Zagadnienie syntezy liniowych korekcyjnych kodów binarnych można ująć następująco: dana jest długość kodu n i znana jest liczba t błędów, które chcemy skorygować; należy znaleźć kod, którego liczba znaków informacyjnych k będzie największa, przy czym waga w każdego ciągu kodowego musi spełniać zależność $w \geq 2t+1$. Dla rozwiązania tego zagadnienia stosuje się struktury algebraiczne takie, jak grupy, pierścienie, ciała i przestrzenie wektorowe. Niniejsza praca daje systematyczny przegląd tych struktur ze szczególnym uwzględnieniem pierścieni wielomianów i ciał Galois stanowiących podbudowę matematyczną kodów cyklicznych. Praca ilustrowana jest przykładami.

1. Wstęp

Zagadnienie kodowania, czyli odwzorowywania ciągów symboli źródła S na ciągi symboli kodu X (rys. 1), znane jest już od czasu wprowadzenia telegrafii. W najprostszym przypadku alfabet źródła S stanowią litery i cyfry systemu dziesiętnego, a alfabet kodu X stanowią symbole 0 i 1.

W związku z rozwojem maszyn cyfrowych oraz ich zastosowaniem do przetwarzania i transmisji danych wierność przesyłania informacji cyfrowych stała się zagadnieniem podstawowym: błędny odbiór choćby jednego elementu może całkowicie zmienić sens



Rys. 1. Alfabet źródła S i alfabet kodu X

wiadomości. Powstaje więc potrzeba konstruowania kodów, umożliwiających tylko wykrywanie albo wykrywanie i korekcję błędów powstałych w czasie przekazywania informacji przez kanał. Proste kody, umożliwiające wykrywanie pojedynczych błędów były znane i stosowane już od dawna. Przykładem jest kod C_5^2 lub C_7^3. Pierwszy wykorzystuje tylko te 10 kombinacji kodu 5-pozycyjnego, w których liczba jedynek jest równa 2. W kodzie C_7^3 z $2^7 = 128$ wszystkich możliwych kombinacji kodu 7-pozycyjnego wybiera się tylko $\binom{7}{3} = 35$ kombinacji, w których liczba jedynek jest równa 3.

Z tego można wyciągnąć wniosek, że zasadniczą cechą kodów detekcyjnych jest nadmiarowość, zwana również rozwlekłością lub redundacją. Jeśli nadmiarowość jest odpowiednio duża, to oprócz stwierdzenia wystąpienia błędu możliwe jest określenie, na której pozycji wyrazu kodowego wystąpił błąd i tym samym skorygowanie błędu.

Kody korekcyjne są bardziej złożone niż kody detekcyjne i tym samym oprzyrządowanie związane z ich stosowaniem jest bardziej skomplikowane i kosztowne. Powstaje pytanie, czy w układzie przenoszenia informacji, przedstawionym na rys. 2, nie wystarczyłyby kody umożliwiające wykrywanie błędów. Stwierdzenie błędu może bowiem spowodować żądanie powtórzenia informacji i tym samym wyeliminowanie błędu.

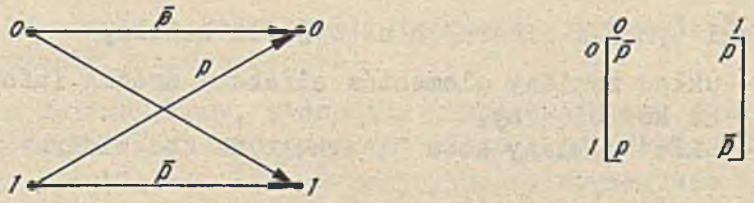
Taka operacja jest jednak niemożliwa, jeśli w torze przenoszenia informacji znajduje się układ pamięciowy. Mianowicie

wtedy nadejście informacji do punktu odbiorczego może mieć miejsce po kilku godzinach czy dniach. Stwierdzenie błędu, który może wystąpić zarówno w kanale, jak i w pamięci (np. na taśmie magnetycznej), nie może już spowodować powtórzenia informacji, a więc tym samym omówioną metodą nie można błędu skorygować. Te i podobne momenty uzasadniają stosowanie kodów korekcyjnych.

Rys. 2. Blokowy układ przenoszenia informacji

Kanał informacyjny można opisać przez:

1) podanie alfabetu wejściowego $A = \{a_i\}$; $i = 1, \ldots r$,
2) podanie alfabetu wyjściowego $B = \{b_j\}$; $j = 1, \ldots s$,
3) określenie prawdopodobieństw $P = (b_j/a_i)$ dla wszystkich i i j, tj. prawdopodobieństw pojawienia się symbolu b_j przy założeniu, że wysłano symbol a_i.



Rys. 3. Opis graficzny i macierzowy binarnego kanału symetrycznego

W przypadku kanału binarnego alfabet zarówno wejściowy, jak i wyjściowy składa się z 2 elementów: 0 i 1. Poszczególne praw-

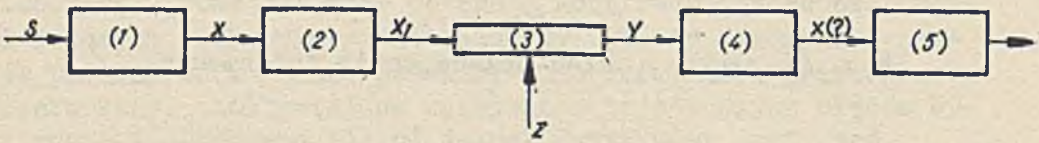
dopodobieństwa będą wtedy: P(0/0), P(0/1), P(1/1) i P(1/0). Jeżeli zachodzi

$$P(0/0) = P(1/1)$$

$$P(1/0) = P(0/1),$$

to otrzymujemy tzw. kanał binarny symetryczny, przedstawiony graficznie na rys. 3. Kanał taki można opisać również za pomocą macierzy podanej obok rys. 3.

Przy założeniu kodu binarnego pełny schemat blokowy układu transmisji informacji jest taki jak na rys. 4.



Rys. 4. Pełny schemat blokowy układu przenoszenia informacji

Poszczególne symbole i bloki na tym rysunku oznaczają:

S - źródło,

X - kod binarny,

X_1 - kod binarny rozszerzony,

Z - zakłócenia,

Y - ciągi symboli binarnych na wyjściu kanału,

blok (1) - układ zamiany elementów alfabetu źródła informacji na kod binarny,

blok (2) - układ zamiany kodu binarnego na kod binarny rozszerzony,

blok (3) - kanał informacyjny,

blok (4) - układ zamiany odebranych ciągów binarnych na zwykły kod binarny,

blok (5) - układ zamiany kodu binarnego na alfabet źródła.

Przy określonym kanale informacyjnym problem kodowania sprowadza się do znalezienia zależności między ciągami X i X_1 oraz do budowy bloków (2) i (4).

Pierwszą publikacją, dotyczącą kodów korekcyjnych, była praca W. Hamminga [1]. Istotne punkty tej pracy to:

1. Określenie wagi ciągów kodowych i odległości między ciągami. Mianowicie dla ciągu kodowego

$$\alpha = (a_1, a_2, \ldots a_n);\ a_i = 0 \text{ lub } 1.$$

wagę Hamminga określa się przez

$$a = \|\alpha\| = \sum_{i=1}^{n} a_i$$

Odległość Hamminga 2 ciągów kodowych jest równa wadze sumy tych ciągów:

$$d(\alpha,\beta) = \|\alpha+\beta\|$$

gdzie sumowanie jest sumowaniem mod 2 poszczególnych elementów tych ciągów:

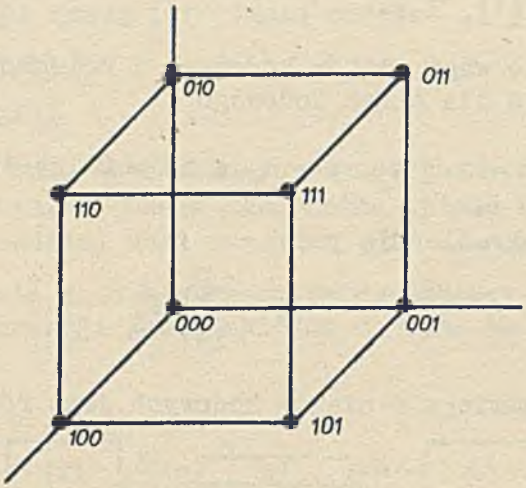
$$\alpha+\beta = (a_1+b_1,\ a_2+b_2,\ \ldots,\ a_n+b_n)$$

2. Określenie zależności między nadmiarem kodu a liczbą wykrywalnych i korygowalnych błędów.

3. Wyróżnienie pozycji informacyjnych i pozycji kontrolnych w kombinacjach kodowych oraz określenie ich współzależności.

4. Konstrukcja kodu umożliwiającego wykrywanie 2 błędów lub skorygowanie jednego błędu.

Opis matematyczny, stosowany przez Hamminga, jest typu geometrycznego. Zilustrowano to w nieco zmodyfikowanej formie na prostym przykładzie kodu 3-pozycyjnego. Mianowicie wszystkie słowa kodowe można rozważać jako wierzchołki kostki o boku równym 1 (rys. 5). Zbiór tych punktów można traktować jako przestrzeń wektorową 3-wymiarową, przy czym każdą pozycję ciągu kodowego przyjmuje się jako współrzędną tego ciągu, przy założeniu, że określona współrzędna jest równa 0 lub 1.



Rys. 5. Geometryczny obraz kodu binarnego 3-pozycyjnego

Przy niniejszych rozważaniach, dotyczących kodów binarnych, wystarczy następujące określenie wektora:

1. Przez wektor w przestrzeni n-wymiarowej rozumie się ciąg zero-jedynkowy o długości n.

2. Ciąg złożony z n zer nazywa się wektorem zerowym przestrzeni.

3. Sumę wektorów określa się następująco:

$$(a_1, a_2, \ldots a_n)+(b_1, b_2, \ldots b_n) = (a_1 + b_1, a_2+b_2, \ldots a_n+b_n)$$

gdzie: a_i, b_i = 0 lub 1, przy czym obowiązuje zasada dodawania (mod 2), tj. $0+0=0$, $0+1=1+0=1$, $1+1=0$

4. Jeśli u i v są w przestrzeni, to ich suma u + v jest w przestrzeni.

5. $u + v = v + u$

6. Jeśli wektor zerowy oznaczymy przez 0, to zachodzi $u + 0 = 0 + u = u$.

7. Dla każdego wektora u istnieje wektor przeciwny v, taki, że zachodzi $u + v = 0$.

Łatwo zauważyć, że każdy wektor binarny jest względem siebie przeciwny, tzn. $u + u = 0$.

(W rzeczywistości w przestrzeni wektorowej spełnione są jeszcze inne działania, a więc jest to struktura bogatsza niżliby to wynikało z ww. określenia).

Przestrzeń wektorową można opisać w zwartej postaci za pomocą macierzy generującej, która w omawianym przypadku może przyjąć postać:

$$G_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Cechą szczególną macierzy generującej jest to, że jej wektory wierszowe są od siebie liniowo niezależne, tzn., że żaden z tych wektorów nie jest liniową kombinacją pozostałych.

Jeżeli oznaczymy wektory macierzy generującej (zwane również wektorami bazy) przez v_1, v_2, v_3, to ich liniowe kombinacje dają pozostałe elementy przestrzeni wektorowej, czyli pozostałe wektory kodu:

$$\begin{aligned}
1\,0\,0 &= v_1 \\
0\,1\,0 &= v_2 \\
0\,0\,1 &= v_3 \\
1\,1\,0 &= v_1 + v_2 \\
1\,0\,1 &= v_1 + v_3 \\
0\,1\,1 &= v_2 + v_3 \\
1\,1\,1 &= v_1 + v_2 + v_3 \\
0\,0\,0 &= v_1 + v_1
\end{aligned}$$

Dalszą właściwością wyżej podanej macierzy generującej jest to, że wektory kolumnowe są też od siebie liniowo niezależne.

Jeżeli macierzą generującą będzie

$$G_2 = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix} \begin{matrix} = v_1 \\ = v_2 \end{matrix}$$

to ich liniowe kombinacje będą

$$1\ 0\ 1 = v_1 + v_2$$

$$0\ 0\ 0 = v_1 + v_1,$$

a więc łączna liczba wektorów kodowych będzie równa 4. Wektory kolumnowe tej nowej macierzy nie są już od siebie liniowo niezależne, każda z kolumn jest sumą dwóch pozostałych. Przyjęcie takiego kodu pozwala na wykrycie jednego błędu. Poszczególne błędy zapisane w postaci wektorów będą miały następującą postać

$$e_1 = 0\ 0\ 1$$
$$e_2 = 0\ 1\ 0$$
$$e_3 = 1\ 0\ 0$$

Na koniec możemy przyjąć, że kod składa się tylko z 2 ciągów a mianowicie:

$$0\ 0\ 0$$
$$1\ 1\ 1$$

co można zapisać za pomocą macierzy

$$G_3 = [111].$$

Kod taki pozwala wykryć 2 błędy lub skorygować 1 błąd; oto wektory błędów wykrywalnych

001
010
011
100
101
110

i wektory błędów korygowalnych

001
010
100

Te ostatnie wektory dodane do wektorów kodowych dają następujące ciągi

001		110
010	lub	101
100		011

Pierwsze 3 z tych ciągów są odległe o 1 od kombinacji 000 i o 2 od kombinacji 111, wobec czego, zakładając, że wystąpił tylko 1 błąd, tłumaczymy to na kombinację 000. Podobnie pozostałe 3 kombinacje tłumaczymy na wektor kodowy 111.

Taki sposób korekcji można by wykorzystać w metodzie kodowania, polegającej np. na 3-krotnym powtarzaniu każdego symbolu. Tak więc kombinację 01101 kodujemy w postaci 000111111000111. Nietrudno się przekonać, że kod taki w szczególnym przypadku może skorygować 5 błędów, jeśli błąd będzie typu 001010001100010, ale już wykrycie błędu typu 000110000000000 jest niemożliwe, a więc mimo dużej długości kod ten jest nieefektywny.

Minimalne wagi i minimalne odległości dla w.w. kodów są następujące:

kod	min waga	min odległość	liczba wykrywalnych błędów	liczba korygowalnych błędów
G_1	1	1	0	0
G_2	2	2	1	0
G_3	3	3	2	1

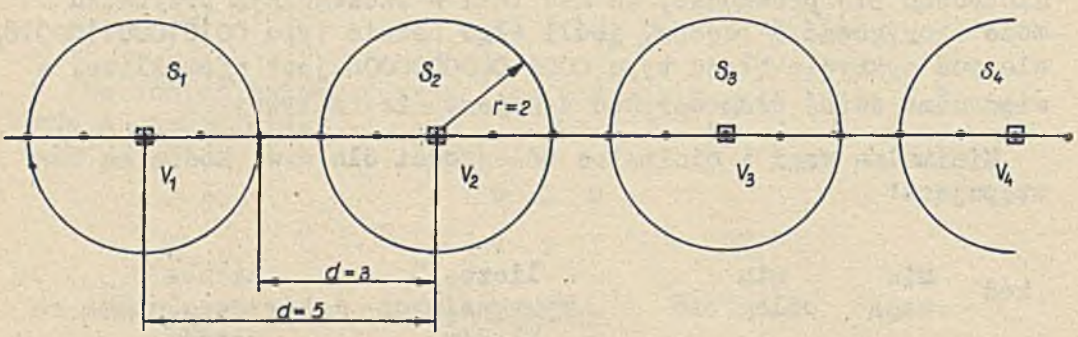
Kod podany przez Hamminga [1] można opisać za pomocą macierzy generującej G:

$$G = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}$$

Po rozwinięciu otrzymuje się 15 niezerowych wektorów kodowych o minimalnej wadze równej 3. Tym samym minimalna odległość

jest też równa 3, a zatem kod ten pozwala wykryć 2 błędy lub skorygować 1. Pierwsze 4 pozycje tego kodu są pozycjami informacyjnymi, a pozostałe - pozycjami kontrolnymi.

Hamming określił związek między liczbą korygowalnych błędów a minimalną odległością między wektorami kodowymi: dla korekcji t błędów minimalna odległość musi wynosić 2t + 1. W sposób przybliżony, dla t = 2 jest to zilustrowane na rys. 6
Jeśli wektor kodowy oznaczymy przez v_i, a przez e_i oznaczymy wektor błędu mający jedynki w tych miejscach, w których zaszły pomyłki przy przekazywaniu ciągu kodowego v_i, to odebrany wektor będzie równy $v_i + e_i$.



Rys. 6. Uproszczony obraz min. odległości między wektorami kodowymi i zdolności korekcyjnej kodu

Liczba błędów jest tu równa wadze wektora e_i: $w(e_i)$.
Dany kod koryguje t błędów, jeśli wektor $v_i + e_i$ należy do zbioru S_i, a więc jeśli $w(e) \leq t$. Zakłada się przy tym, że przekrój 2 dowolnych zbiorów S_i i S_j jest równy zeru.
Powstaje teraz zagadnienie konstrukcji kodów posiadających w/w właściwości, które to zagadnienie można ująć następująco: dana jest długość kodu n, i znana jest liczba korygowalnych błędów t, należy znaleźć kod, którego liczba znaków informacyjnych k, a więc i liczba wyrazów kodowych 2^k, będzie największa. W dążeniu do jak najlepszego rozwiązania tego zagadnienia zastosowano struktury algebraiczne takie jak grupy, pierścienie,

ciała, przestrzenie wektorowe i algebry [2], [3], [6], [8]. Opis algebraiczny okazał się bardzo owocny, bo:

1) podaje przepis na konstrukcję kodu,

2) umożliwia zapis kodu w sposób zwarty,

3) podaje informacje o właściwościach kodu.

Niniejsza praca poświęcona jest kodom cyklicznym, których podbudowę matematyczną stanowią pierścienie wielomianów i ciała Galois.

2. Podstawowe wiadomości o strukturach algebraicznych

Strukturą algebraiczną nazywa się zbiór elementów, spełniający pewne określone aksjomaty. W teorii kodów - w mniejszym lub większym stopniu - znajdują zastosowanie następujące struktury: grupy, pierścienie, ciała, przestrzenie wektorowe i algebry.

Poniżej zostaną pokrótce omówione grupy, pierścienie i ciała. Omówienie rozpoczyna się od przytoczenia 11 aksjomatów. Dalej podano, które z tych aksjomatów obowiązują w poszczególnych strukturach. Te krótkie rozważania są zilustrowane kilku przykładami.

Przez S oznaczono tu zbiór, a przez a, b, c elementy zbioru S. Aksjomaty oznaczono symbolami: A_1, A_2, ..., A_{11}.

a) Wykaz aksjomatów

A1. Jeśli a i b są w S, to a + b jest w S

A2. Jeśli a i b są w S, to a + b = b + a

A3. Jeśli a, b i c są w S, to a + (b + c) = (a + b) + c

A4. W S istnieje element neutralny zwany 0, taki, że 0 + a = a dla wszystkich a w S

A5. Dla każdego a w S istnieje element b w S taki, że a + b = 0.

A6. Jeśli a i b są w S, to a . b jest w S.

A7. Jeśli a i b są w S, to a . b = b . a.

A8. Jeśli a, b i c są w S, to a.(b . c) = (a . b) . c.

A9. W S istnieje element neutralny zwany 1 taki, że 1.a=a, dla wszystkich a z S.

A10. Dla każdego a = 0 w S istnieje w S element b taki, że a . b = 1.

A11. Jeśli a, b i c są w S, to a.(b + c) = ab + ac.

Jak widać z powyższego zestawienia, wprowadzono tu 2 operacje, z których pierwsza oznaczona symbolem "+" zwie się dodawaniem, a druga oznaczona symbolem "." mnożeniem. Działania te nie muszą się pokrywać ze zwykłym dodawaniem i mnożeniem znanym z arytmetyki.

b) Wykaz struktur

	A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8	A_9	A_{10}	A_{11}
1. Grupa addytywna abelowa	x	x	x	x	x						
2. Grupa multyplikatywna abelowa						x	x	x	x	x	
3. Pierścień	x	x	x	x	x	x					x
4. Pierścień przemienny z jedynką	x	x	x	x	x	x	x	x	x		x
5. Ciało	x	x	x	x	x	x	x	x	x	x	x

c) Przykłady

1. Zbiór złożony z 2 elementów: 0 i 1 z określonymi w nim operacjami zwykłego (arytmetycznego) mnożenia i dodawania spełnia aksjomaty A_4, A_6, A_7, A_8, A_9 i A_{10}. Tak określona struktura jest więc grupą multyplikatywną.

2. Zbiór złożony z 0 i 1 z dwoma operacjami: mnożeniem i dodawaniem boole'owskim spełnia aksjomaty A_1, A_2, A_3, A_4, A_6, A_7, A_8, A_9, A_{10}, A_{11}. Struktura ta jest również grupą multyplikatywną.

3. Zbiór złożony z 0 i 1 z operacją dodawania mod 2 spełnia aksjomaty $A_1 \div A_5$, a więc jest grupą addytywną.

4. Zbiór 2^n ciągów zero-jedynkowych o długości n, w których dowolny element można zapisać w postaci $a = (a_1, a_2, \ldots a_n)$, $a_i = 0$ lub 1, jest grupą, jeśli dodawanie określone jest następująco:

$$a + b = (a_1 + b_1, a_2 + b_2, \ldots, a_n + b_n).$$

Dodawanie poszczególnych elementów ciągu jest dodawaniem (mod 2). Jeśli element zerowy tego zbioru oznaczymy przez

$$\emptyset = (0, 0 \ldots 0),$$

to zachodzi

$$a + a = \emptyset$$

dla każdego elementu zbioru. Inaczej mówiąc, każdy element zbioru jest względem siebie elementem przeciwnym.

5. Zbiór złożony z 0 i 1 z operacją mnożenia i dodawania mod 2 jest ciałem, bo spełnione są w nim wszystkie 11 aksjomatów.

6. Zbiór liczb całkowitych (dodatnich, ujemnych i zera) z operacją arytmetycznego dodawania i mnożenia spełnia aksjomaty $A_1 \div A_9$ i A_{11}, jest więc pierścieniem.

7. Zbiór liczb całkowitych z dodawaniem i mnożeniem modulo dowolna liczba całkowita spełnia aksjomaty $A_1 \div A_9$ i A_{11}, a więc jest pierścieniem.

8. Zbiór liczb całkowitych z dodawaniem i mnożeniem modulo dowolna liczba pierwsza spełnia wszystkie aksjomaty $A_1 \div A_{11}$, a więc zbiór ten jest ciałem.

9. Zbiór wszystkich wielomianów jednej zmiennej z dodawaniem i mnożeniem znanym z algebry elementarnej jest pierścieniem.

3. Pierścień liczb całkowitych

Rozważania dotyczące pierścieni wielomianów rozpoczniemy od omówienia zbioru liczb całkowitych. Mianowicie zbiór ten ułożymy w postaci następującej tabeli zawierającej 6 warstw

0	6	-6	12	-12	18	-18	24 ....
1	7	-5	13	-11	19	-17	25 ....
2	8	-4	14	-10	20	-16	26 ....
3	9	-3	15	-9	21	-15	27 ....
4	10	-2	16	-8	22	-14	28 ....
5	11	-1	17	-7	23	-13	29 ....

W pierwszej warstwie występują wielokrotności liczby 6, drugą warstwę otrzymujemy przez dodanie do elementów pierwszej warstwy liczby 1, trzecią przez dodanie liczby 2 itd. Dany element określonej warstwy po podzieleniu przez 6 daje resztę równą elementowi pierwszej kolumny tejże warstwy, wobec czego poszczególne warstwy, zwane w teorii pierścieni klasami reszt, możemy opisać według elementów pierwszej kolumny. Dla odróżnienia ujmiemy te wyrazy w klamry. Tak więc mamy tu następujące klasy: $\{0\}$, $\{1\}$, $\{2\}$, $\{3\}$, $\{4\}$ i $\{5\}$. Elementy zbioru tych klas możemy dodawać i mnożyć według następującej reguły:

1) tworzymy sumę lub iloczyn w taki sposób, jak to się robi w arytmetyce,

2) tak otrzymany wynik dzielimy przez 6 i resztę z dzielenia traktujemy jako wynik naszej operacji.

Pokażemy to na przykładach, przy czym dla uproszczenia pomijamy tu klamry

1) $1 + 3 = 4$

2) $3 + 5 = 8$; $8 : 6 = 1$ r. 2, czyli
$3 + 5 = 2 \pmod 6$

3) $3 \circ 4 = 12$; $12 : 6 = 2$ r 0, czyli
$3 \circ 4 = 0 \pmod 6$

4) $3 \circ 5 = 15$; $15 : 6 = 2$ r 3, czyli
$3 \circ 5 = 3 \pmod 6$

Mówimy, że jest to dodawanie i mnożenie według modułu 6, a więc piszemy np. $3 \circ 5 = 3 \pmod 6$.

Postępując w podobny sposób można utworzyć tabelkę dodawania i mnożenia (mod. 6).

+	0	1	2	3	4	5
0	0	1	2	3	4	5
1	1	2	3	4	5	0
2	2	3	4	5	0	1
3	3	4	5	0	1	2
4	4	5	0	1	2	3
5	5	0	1	2	3	4

∘	0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	1	2	3	4	5
2	0	2	4	0	2	4
3	0	3	0	3	0	3
4	0	4	2	0	4	2
5	0	5	4	3	2	1

Jeżeli oznaczymy omawiany zbiór przez R, a elementy zbioru przez a, b, c, to możemy stwierdzić, że spełnia on aksjomaty: A_1, A_2, A_3, A_4, A_5, A_6, A_{11}, a więc zbiór ten jest pierścieniem.

Pierwszą warstwę tabeli, jak na str. 160 nazywamy ideałem pierścienia, a pierwszy niezerowy element ideału nazywamy generatorem ideału.

Pierścień złożony z elementów 0, 1, 2, 3, 4, 5 z dodawaniem i mnożeniem mod 6 można rozłożyć na klasy reszt w dwojaki sposób, a mianowicie według ideału, którego generatorem jest 2:

0, 2, 4

1, 3, 5,

albo według ideału, którego generatorem jest 3:

0, 3
1, 4
2, 5

Pomijamy tu przypadek, kiedy ideałem jest cały pierścień.

4. Pierścienie wielomianów i kody cykliczne [3], [4], [5], [6]

Powyższe rozważania możemy przenieść na wielomiany jednej zmiennej o współczynnikach przy zmiennych równych 0 lub 1. Wielomiany można rozłożyć na warstwy, przy czym w pierwszej warstwie występują wielokrotności dowolnego wielomianu, na przykład $x^n + 1$. Otrzymane warstwy są klasami reszt wielomianów (mod $x^n + 1$). Możemy je oznaczyć według pierwszego wyrazu w każdej warstwie. Otrzymamy w ten sposób zbiór klas, który jest pierścieniem wielomianów (mod $x^n + 1$). Ten nowy pierścień możemy rozłożyć na klasy reszt według ideału generowanego przez wielomian będący dzielnikiem wielomianu $x^n + 1$.

Przykład 1

Niech $x^n + 1 = x^5 + 1$. Zbiór wszystkich wielomianów o współczynnikach 0 lub 1 można rozłożyć na klasy według ideału generowanego przez $x^5 + 1$.

Początek tabeli będzie wyglądał następująco:

0	x^5+1	x^6+x	x^6+x^5+x+1	.....
1	x^5	x^6+x+1	x^6+x^5+x	.....
x	x^5+x+1	x^6	x^6+x^5+1	.....
$x+1$	x^5+x	x^6+1	.	.
x^2	x^5+x^2+1	x^6+x^2+x	.	.
x^2+1	x^5+x^2	.	.	.
x^2+x	.	.	.	
.	.	.		
.	.	.		
$x^4+x^3+x^2+x+1$	$x^5+x^4+x^3+x^2+x$	.		

Należy pamiętać, że przy operowaniu wielomianami obowiązuje dodawanie (mod 2):

$$1 \cdot x^k + 1 \cdot x^k = 0 \cdot x^k$$
$$1 \cdot x^k + 0 \cdot x^k = 1 \cdot x^k$$
$$0 \cdot x^k + 1 \cdot x^k = 1 \cdot x^k$$
$$0 \cdot x^k + 0 \cdot x^k = 0 \cdot x^k$$

Będzie więc np.

$$(x^5 + x) + (x + 1) = x^5 + 1$$

$$(x^5 + 1)^2 = x^{10} + 1$$

Wielomian w k-tym wierszu pierwszej kolumny jest resztą otrzymaną po podzieleniu dowolnego wielomianu w k-tym wierszu przez x^5+1. Tak np. wielomian $x^7+x^5+x^2+x$ znajdujący się w czwartym wierszu tabeli daje po podzieleniu przez $x^5 + 1$ resztę $x + 1$, tj. wielomian w czwartym wierszu pierwszej kolumny. Istotnie mamy:

$$(x^7 + x^5 + x^2 + x) : (x^5 + 1) = x^2 + 1$$

$$\begin{array}{r|l} x^7 + x^2 & \\ \hline x^5 + x & \\ \hline x^5 + 1 & \\ \hline x + 1 & \end{array}$$

Klasy reszt otrzymane w powyższej tabeli możemy oznaczyć odpowiednio: $\{0\}$, $\{1\}$, $\{x\}$, $\{x + 1\}$, $\{x^2\}$, ... $\{x^4+x^3+x^2+x+1\}$. Tworzą one pierścień wielomianów (mod $1 + x^5$). Pierścień ten można

rozłożyć na klasy reszt według ideału, którego generator $g(x)$ musi być dzielnikiem wielomianu $x^5 + 1$. Otóż $x^5 + 1 = (x + 1)(x^4+x^3+x^2+x+1)$. Jeżeli przyjąć, że $g(x) = 1 + x$, to otrzymamy dwie klasy, które oznaczamy przez 0 i 1.

$\{0\}$	0	$x+1$	x^2+x	x^2+1	x^3+x^2	...	x^4+1
$\{1\}$	1	x	x^2+x+1	x^2	x^3+x^2+1	...	x^4

Zauważmy, że elementy ideału nie są liniowo niezależne. Już na przykład trzeci element jest sumą pierwszego i drugiego: $(1+x) + (x+x^2) = 1 + x^2$. W istocie do określenia tego ideału wystarczą cztery wyrazy:

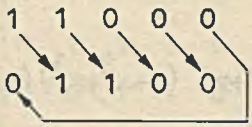
$$1 + x = g(x)$$
$$x + x^2 = x \cdot g(x)$$
$$x^2+x^3 = x^2 \cdot g(x)$$
$$x^3 + x^4 = x^3 \cdot g(x),$$

bo wszystkie pozostałe są ich liniowymi kombinacjami. Zamiast wypisywać wielomiany, możemy zapisać tylko współczynniki przy zmiennej x, przy czym zapisu dokonujemy w postaci macierzy generującej zawierającej w/w 4 wyrazy:

$$G = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

Wiersze tej macierzy i wszystkie liniowe kombinacje tych wierszy tworzą podprzestrzeń wektorową, zwaną podprzestrzenią

cykliczną albo kodem cyklicznym. Nazwa kodu wywodzi się z cykliczności przesuwania elementów kodu, np.:



Tak więc wiersze macierzy generującej dany kod wynikają z cyklicznego przestawienia elementów w wierszu poprzednim. Zatem znając pierwszy wiersz znamy całą macierz generującą, a więc i wszystkie wyrazy kodowe. Pierwszy wiersz jest zaś tylko pewnym sposobem zapisu generatora ideału pierścienia wielomianów $(\mathrm{mod}\ x^n + 1)$.

Z tego płynie wniosek, że znajomość wielomianu generującego $g(x)$ i znajomość wielomianu $x^n + 1$, którego $g(x)$ jest dzielnikiem całkowicie opisuje kod.

W naszym przykładzie poszczególne wyrazy kodu cyklicznego będą równe:

$$
\begin{aligned}
1\ 1\ 0\ 0\ 0 &= v_1 \\
0\ 1\ 1\ 0\ 0 &= v_2 \\
0\ 0\ 1\ 1\ 0 &= v_3 \\
0\ 0\ 0\ 1\ 1 &= v_4 \\
1\ 0\ 1\ 0\ 0 &= v_5 = v_1 + v_2 \\
1\ 1\ 1\ 1\ 0 &= v_6 = v_1 + v_3 \\
1\ 1\ 0\ 1\ 1 &= v_7 = v_1 + v_4 \\
0\ 1\ 0\ 1\ 0 &= v_8 = v_2 + v_3 \\
0\ 1\ 1\ 1\ 1 &= v_9 = v_2 + v_4 \\
0\ 0\ 1\ 0\ 1 &= v_{10} = v_3 + v_4 \\
1\ 0\ 0\ 1\ 0 &= v_{11} = v_1 + v_2 + v_3 \\
1\ 0\ 1\ 1\ 1 &= v_{12} = v_1 + v_2 + v_4 \\
1\ 1\ 1\ 0\ 1 &= v_{13} = v_1 + v_3 + v_4 \\
0\ 1\ 0\ 1\ 1 &= v_{14} = v_2 + v_3 + v_4 \\
1\ 0\ 0\ 0\ 1 &= v_{15} = v_1 + v_2 + v_3 + v_4
\end{aligned}
$$

Przykład 2

Generatorami ideałów pierścienia wielomianów (mod $1 + x^7$), w oparciu o zależność

$$1 + x^7 = (1+x)\ (1+x^2+x^3)(1+x+x^3),$$

mogą być następujące wielomiany:

$$g_1(x) = 1+x$$
$$g_2(x) = 1+x^2+x^3$$
$$g_3(x) = 1+x+x^3$$
$$g_4(x) = g_1(x) \cdot g_2(x) = 1+x+x^2+x^4$$
$$g_5(x) = g_1(x) \cdot g_3(x) = 1+x^2+x^3+x^4$$
$$g_6(x) = g_2(x) \cdot g_3(x) = 1+x+x^2+x^3+x^4+x^5+x^6$$

W przypadku $g_1(x) = 1+x$ mamy następujące wielomiany liniowo niezależne:

$$g_1(x) = 1 + x$$
$$x \cdot g_1(x) = x + x^2$$
$$x^2 \cdot g_1(x) = x^2 + x^3$$
$$x^3 \cdot g_1(x) = x^3 + x^4$$
$$x^4 \cdot g_1(x) = x^4 + x^5$$
$$x^5 \cdot g_1(x) = x^5 + x^6,$$

odpowiednia macierz generująca będzie więc

$$G_1 = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

Wierszy liniowo niezależnych jest 6, wobec czego liczba wszystkich wyrazów kodowych (bez zerowego) jest równa $2^6 - 1 = 63$.

Dokonując tzw. operacji elementarnych (p. np. [8]) możemy macierz G_1 przekształcić w macierz równoważną G_{11}:

$$G_{11} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Pierwszy wiersz macierzy G_{11} pokrywa się z pierwszym wierszem macierzy G_1.

Drugi wiersz G_{11} jest sumą 2 pierwszych wierszy G_1

Trzeci wiersz G_{11} jest sumą 3 pierwszych wierszy G_1

Czwarty wiersz G_{11} jest sumą 4 pierwszych wierszy G_1

Piąty wiersz G_{11} jest sumą 5 pierwszych wierszy G_1

Szósty wiersz G_{11} jest sumą wszystkich wierszy G_1.

Łatwo można się przekonać, że wiersze macierzy G_{11} są też liniowo niezależne. Opisany kod jest przykładem najprostszego

kodu nadmiarowego; charakteryzuje się on tym, że liczba jedynek w każdym wyrazie tego kodu jest liczbą parzystą. Liczba kombinacji wynosi

$$\binom{7}{2} + \binom{7}{4} + \binom{7}{6} = 21 + 35 + 7 = 63,$$

co pokrywa się z poprzednim wyliczeniem.

Tę samą liczbę wyrazów kodowych można otrzymać przy 6 kolumnach informacyjnych. Wnosimy z tego, że siódma pozycja w danym ciągu kodowym nie ma znaczenia jako pozycja informacyjna, może to więc być tylko pozycja kontrolna.

Macierz generującą w kodzie cyklicznym można zapisać w postaci

$$G = \left[R_{n-k}\ I_k\right]$$

gdzie: I_k jest macierzą jednostkową o wymiarze k x k, będącą równocześnie macierzą informacyjną, a R_{n-k} jest macierzą kontrolną o wymiarze k x (n-k). Taki sposób zapisu, to jest pozycje kontrolne z lewej strony, a pozycje informacyjne z prawej, przyjął się w kodach cyklicznych, odpowiada to przepływowi informacji od lewej strony ku prawej. W omawianym przypadku macierz G_{11} zapiszemy w postaci

$$G_{11} = \left[R_1\ I_6\right]$$

Jeśli jako generator ideału pierścienia obrać wielomian $g_3(x) = 1 + x^2 + x^3$, to macierz generująca będzie

$$\begin{matrix} & & & & & & & p_1 & p_2 & p_3 & i_1 & i_2 & i_3 & i_4 \end{matrix}$$

$$G = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 \end{bmatrix} = \left[R_3\ I_4\right]$$

W tym przypadku mamy tylko 4 pozycje informacyjne: i_1, i_2, i_3, i_4; pozostałe 3: p_1, p_2, p_3 są pozycjami kontrolnymi. Zależności zachodzące między nimi są następujące

$$p_1 = i_1 + i_2 + i_3$$
$$p_2 = i_2 + i_3 + i_4$$
$$p_3 = i_1 + i_2 + i_4$$

Warto zauważyć, że indeksy przy "i" pokrywają się z numerami tych wierszy macierzy $[R\ I]$, w których w danej kolumnie "p" występują jedynki. Tak np. w kolumnie p_2 występują jedynki w 2, 3 i 4 wierszu, wobec czego możemy od razu napisać

$$p_2 = i_2 + i_3 + i_4$$

Minimalna waga kodu generowanego przez G_3, a więc tym samym minimalna odległość między wektorami kodowymi, jest równa 3. Oznacza to, że kod ten pozwala skorygować 1 błąd. W istocie jest to przytoczony uprzednio kod Hamminga, zapisany w nieco innej formie.

Przykład 3

Kod generowany przez $g(x) = x^{10}+x^8+x^5+x^4+x^2+x+1$, będący dzielnikiem wielomianu $x^{15} + 1$, można przedstawić za pomocą macierzy

$$G = \begin{array}{c} \begin{matrix} p_1 & p_2 & p_3 & p_4 & p_5 & p_6 & p_7 & p_8 & p_9 & p_{10} & i_1 & i_2 & i_3 & i_4 & i_5 \end{matrix} \\ \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \end{array} = [R_{10}\ I_5]$$

Liczba pozycji informacji jest równa 5, pozostałe 10 to pozycje kontrolne.

Kod cykliczny można - przy znajomości wielomianu generującego g(x) - krótko zapisać w postaci (n,k), co interpretujemy następująco:

liczba elementów kodu = n,

liczba elementów informacyjnych = k,

liczba elementów kontrolnych = n - k,

stopień wielomianu generującego = n - k.

Kod w przykładzie 3 można więc krótko zapisać w formie (15,5); liczba elementów kodu wynosi tu 15, liczba elementów informacyjnych 5, liczba elementów kontrolnych 15-5 = 10, stopień generatora ideału g(x) jest równy 15-5 = 10.

W oparciu o powyższy przykład omówimy sposób tworzenia kodu cyklicznego z kodu informacyjnego, przy założeniu, że znany jest wielomian generujący g(x). W cytowanym przykładzie pozycje informacyjne można zapisać w postaci macierzy generującej:

$$I_5 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad \begin{aligned} k_1(x) &= 1 \\ k_2(x) &= x \\ k_3(x) &= x^2 \\ k_4(x) &= x^3 \\ k_5(x) &= x^4 \end{aligned}$$

a wielomian generujący $g(x) = 1+x+x^2+x^4+x^5+x^8+x^{10}$.

Z zapisu macierzy kodu cyklicznego w postaci macierzy generującej G, widać, że macierz I_5 przesunięta jest o 10 miejsc w prawo, co w odniesieniu do każdej pozycji wiersza macierzy I_5 oznacza pomnożenie przez x^{10}, a więc

element 1 kolumny I_5 znajdzie się w 10 kolumnie G
" 2 " I_5 " " w 11 " G
" 3 " I_5 " " w 12 " G
" 4 " I_5 " " w 13 " G
" 5 " I_5 " " w 14 " G

Do tak przesuniętej informacji musimy dodać taki wielomian R(x), żeby w sumie otrzymać wielomian będący wielokrotnością wielomianu generującego

$$x^{10} \cdot k(x) + R(x) = Q(x) \cdot g(x),$$

co oznacza, że R(x) jest resztą z podzielenia wielomianu $x^{10} \cdot k(x)$ przez wielomian genergujący g(x). Niech np.

$k(x) = x^2 \equiv$ ciąg informacyjny 00100

$x^{10} \cdot k(x) = x^{12}$

$x^{12} : (1+x+x^2+x^4+x^5+x^8+x^{10}) = 1+x^2$, $R(x) = 1+x+x^3+x^5+x^6+x^7+x^8$

$f(x) = 1+x+x^3+x^5+x^6+x^7+x^8+x^{12}$

odpowiada to ciągowi 110101111000100.
Ciąg ten pokrywa się z 3 wierszem macierzy G. Podobnie dla ciągu informacyjnego 01100, czyli dla

$k(x) = x^1 + x^2$, mamy

$x^{10} \cdot k(x) = x^{11} + x^{12}$

$(x^{11}+x^{12}) : (1+x+x^2+x^4+x^5+x^8+x^{10}) = 1+x+x^2$, $R(x) = 1+x^2+x^7+x^8+x^9$

$f(x) = R(x) + x^{10} \cdot k(x) = 1+x^2+x^7+x^8+x^9+x^{11}+x^{12}$,

co odpowiada ciągowi 101000011101100, który pokrywa się z ciągiem otrzymanym z G przez zsumowanie 2 i 3 wiersza.

Ogólnie, wzór na wielomian kodu cyklicznego f(x), jest następujący

$$f(x) = x^{n-k} \cdot k(x) + R(x)$$

gdzie:

k - liczba pozycji w kodzie binarnym nierozszerzonym,
n - liczba pozycji w kodzie rozszerzonym,
n-k - liczba pozycji kontrolnych,

$k(x)$ - wielomian informacyjny,
$g(x)$ - wielomian generujący stopnia $n-k$,
$R(x)$ - reszta z podzielenia wielomianu $x^{n-k} \cdot k(x)$ przez $g(x)$,
$f(x)$ - wielomian kodu rozszerzonego.

5. Wykrywanie błędów za pomocą kodów cyklicznych

Z tego co powiedziano wyżej wynika, że wielomianami kodowymi są tylko wielokrotności generatora ideału pierścienia wielomianów $(\mathrm{mod}\ x^n+1)$. Wszystkie inne wielomiany tego pierścienia nie są podzielne przez generator, a więc nie są wielomianami kodowymi. Z tego wyciągamy 2 praktyczne wnioski, dotyczące zdolności kodów cyklicznych do wykrywania błędów.

1. Niepodzielność odebranego wielomianu $h(x)$ przez $g(x)$ oznacza, że $h(x)$ nie jest ciągiem kodowym, a więc, że wystąpił w nim błąd.

2. Podzielność wielomianu $h(x)$ przez $g(x)$ oznacza, że $h(x)$ jest wielomianem kodowym, albo że wystąpił w nim niewykrywalny błąd.

Wprowadzimy oznaczenia:

$f(x)$ - wielomian kodowy,

$e(x)$ - wielomian błędu,

$h(x) = f(x) + e(x)$ = wielomian odebrany.

Ponieważ ilorazy

$$\frac{h(x)}{g(x)} \text{ i } \frac{e(x)}{g(x)}$$

dają te same reszty, więc wystarczy analizować tylko iloraz

$$\frac{e(x)}{g(x)}$$

Pojedynczy błąd $e_1(x)$ możemy zapisać w postaci x^i. Ponieważ $g(x)$ jest wielomianem co najmniej 2-składnikowym, więc x^i nie jest podzielne przez $g(x)$, z czego wniosek, że każdy pojedynczy błąd jest w każdym kodzie cyklicznym wykrywalny.

Jeżeli $g(x)$ zawiera czynnik $x+1$, to oznacza, że $x = 1$ jest pierwiastkiem każdego wielomianu kodowego, a więc, że liczba jedynek w ciągu kodowym jest liczbą parzystą. Oznacza to, że każda nieparzysta liczba błędów zostaje przez taki kod wykryta.

Podwójny błąd można wyrazić w postaci $x^i + x^j = x^i(1 + x^{j-i})$, gdzie $i < j$. Wystarczy stwierdzić, że $1+x^{j-i}$ nie jest podzielne przez $g(x)$.

Mówimy, że wielomian $g(x)$ należy do wykładnika e, jeśli e jest najmniejszą dodatnią liczbą taką, że $x^e + 1$ jest podzielne przez $g(x)$ bez reszty. Jeśli więc dla długości kodu n zachodzi

$$e \geqslant n > (j - i),$$

to na pewno $1 + x^{j-i}$ nie jest podzielne przez $g(x)$, a więc każdy podwójny błąd jest wykrywalny.

Przykłady

1) $g(x) = 1 + x$ należy do wykładnika 1, bo $1 + x$ jest dzielnikiem $1 + x$. Przykładem kodu generowanego przez $g(x) = 1+x$ jest kod przytoczony na str. 167. Ponieważ długość kodu $n = 7$, a $e = 1$, więc kod ten nie wykrywa błędów podwójnych, lecz tylko nieparzystą liczbę błędów.

2) $g(x) = 1 + x^2 + x^3$ należy do wykładnika 7, a więc jeśli długość kodu $n \leqslant 7$, to istnieje możliwość wykrywania pojedynczych i podwójnych błędów. (Por. przykład na str. 168).

3) $g(x) = 1 + x + x^4$ jest dzielnikiem wielomianu $1 + x^{15}$. Kod opisany przez macierz

$$G = \begin{bmatrix} 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 \end{bmatrix}$$

będzie wykrywał pojedyncze i podwójne błędy, bo długość kodu $n < e = 15$.

4) $g(x) = 1 + x + x^2 + x^4 + x^5 + x^8 + x^{10}$ jest dzielnikiem wielomianu $1 + x^{15}$, a więc żeby kod ten (por. str. 169) mógł wykrywać pojedyncze i podwójne błędy, to jego długość nie może być większa niż 15.

Aby móc wykrywać pojedyncze, podwójne i potrójne błędy, trzeba żeby generator był typu $(1 + x) \cdot g_1(x)$ i żeby długość kodu n była nie większa niż wykładnik, do którego należy wielomian $g_1(x)$. Błędy pojedyncze i potrójne są wykrywalne przez obecność czynnika $(1 + x)$ w $g(x)$, a wykrywalność podwójnych błędów wynika z tego, że $g_1(x)$ należy do $e < n$.

Ważną właściwością kodu cyklicznego (n,k) generowanego przez $g(x)$ stopnia $n-k$, jest zdolność wykrywania paczek błędów. I tak każdy kod cykliczny może wykrywać paczkę błędów o długości $n-k$ lub o mniejszej długości. Oznaczmy wektor błędu grupowego przez $r(x) = x^j r_o(x)$, gdzie współczynnik przy x^j jest pierwszym niezerowym współczynnikiem wielomianu $r(x)$ i wobec tego stopień wielomianu $r_o(x)$ jest na pewno mniejszy niż stopień $g(x)$. Gdyby $r(x)$ był wektorem kodowym, to albo x^j, albo $r_o(x)$ byłby podzielny przez $g(x)$. Ale x^j nie może być podzielne przez $g(x)$, bo $g(x)$ jest dzielnikiem $x^n + 1$, a dalej $r_o(x)$ jest stopnia mniejszego niż $n - k$, więc nie może być podzielne przez wielomian $g(x)$ stopnia $n - k$, a zatem $r(x)$ nie może być wektorem kodowym. Tak więc np. w kodzie $(n,k) = (7,4)$, generowanym przez $g(x) = 1+x+x^3$, nałożenie się błędu $r(x) = x^4 + x^6$ na wektor kodowy $1+x+x^2+x^5 = (x^2 + 1) \cdot g(x)$ daje wektor: $1 + x + x^2 + x^4 + x^5 + x^6$, niepodzielny przez $g(x)$.

Liczbę możliwych błędów grupowych o szerokości $b = n - k$ elementów, obliczamy na podstawie następującego rozumowania: jeżeli szerokość paczki błędów wynosi $n - k$, to pierwszy i ostatni element wektora błędu musi być równy 1, a pozostałe $n - k - 2$ pozycji mogą utworzyć 2^{n-k-2} kombinacji. Przy założeniu, że pierwszy i ostatni element wektora błędu znajdują się na tych samych pozycjach, istnieje możliwość wystąpienia 2^{n-k-2} różnych kombinacji paczki błędów o szerokości $n - k$. Taka paczka w wektorze kodowym n-elementowym może zająć $k + 1$ pozycji, wobec czego liczba możliwych paczek błędów o szerokości $n - k$ w kodzie o długości n wynosi $(k + 1) \cdot 2^{n-k-2}$.

Paczka błędów o szerokości n - k - 1 może znaleźć się na k+2 pozycjach, przy czym w każdej pozycji tej paczki możliwych jest 2^{n-k-3} kombinacji wektorów błędów. W sumie możliwych wektorów błędów o długości n - k - 1 jest $(k + 2).2^{n-k-3}$. Zakładając, że najwęższa paczka błędów mogącą znaleźć się na n - 1 pozycjach, jest równa 2, można obliczyć całkowitą możliwą wykrywalną liczbę wektorów błędów o długości n - k lub mniejszej. Wynosi ona

$$(k+1).2^{n-k-2} + (k+2).2^{n-k-3} + \ldots + (n-1).2^{0}$$

Przykład

Dla n = 7, k = 3, n - k = 4 liczba wykrywalnych paczek błędów o długościach 4 lub 3 lub 2 wynosi:

$$4.2^{2} + 5.2 + 6.2^{0} = 32$$

Wektory tych błędów mają postać następującą:

0 0 0 1 0 0 1	0 0 0 0 1 0 1
0 0 0 1 0 1 1	0 0 0 0 1 1 1
0 0 0 1 1 0 1	0 0 0 1 0 1 0
0 0 0 1 1 1 1	0 0 0 1 1 1 0
0 0 1 0 0 1 0	0 0 1 0 1 0 0
0 0 1 0 1 1 0	0 0 1 1 1 0 0
0 0 1 1 0 1 0	0 1 0 1 0 0 0
0 0 1 1 1 1 0	0 1 1 1 0 0 0
0 1 0 0 1 0 0	1 0 1 0 0 0 0
0 1 0 1 1 0 0	1 1 1 0 0 0 0
0 1 1 0 1 0 0	0 0 0 0 0 1 1
0 1 1 1 1 0 0	0 0 0 0 1 1 0
1 0 0 1 0 0 0	0 0 0 1 1 0 0
1 0 1 1 0 0 0	0 0 1 1 0 0 0
1 1 0 1 0 0 0	0 1 1 0 0 0 0
1 1 1 1 0 0 0	1 1 0 0 0 0 0

Część paczek błędów o długości b = n - k może być również wykryta. Załóżmy mianowicie, że paczka o długości b zaczyna się na i-tym elemencie, a kończy się na (i + b - 1) elemencie i ma postać $r(x) = x^{i} . r_{i}(x)$, gdzie $r_{i}(x)$ jest stopnia b - 1.

Wielomian $r_i(x)$ może przyjąć 2^{b-2} kombinacji. Błąd jest niewykrywalny, jeśli $r_i(x)$ jest podzielny przez $g(x)$.

$$r_i(x) = g(x) \cdot Q(x).$$

Ponieważ $g(x)$ jest stopnia $n - k$, a $r_i(x)$ stopnia $b - 1$, więc $Q(x)$ jest stopnia $b - 1 - n + k$. Jeśli $b - 1 = n - k$, czyli $b = n - k + 1$, to $Q(x) = 1$. Tak więc na 2^{b-2} możliwych błędów tylko jeden jest niewykrywalny. Inaczej mówiąc stosunek błędów niewykrywalnych do wszystkich błędów jest dla $b=n-k+1$ równy

$$\frac{1}{2^{n-k-1}}$$

Tak więc np. w kodzie (7,3), generowanym przez $g(x) = 1+x+x^2+x^4$, wektor błędu r_1 o długości $b = n - k + 1 = 5$ może przyjąć następujące postaci:

$$r_{11}(x) = 1\ 0\ 0\ 0\ 1$$
$$r_{12}(x) = 1\ 0\ 0\ 1\ 1$$
$$r_{13}(x) = 1\ 0\ 1\ 0\ 1$$
$$r_{14}(x) = 1\ 0\ 1\ 1\ 1$$
$$r_{15}(x) = 1\ 1\ 0\ 0\ 1$$
$$r_{16}(x) = 1\ 1\ 0\ 1\ 1$$
$$*r_{17}(x) = 1\ 1\ 1\ 0\ 1$$
$$r_{18}(x) = 1\ 1\ 1\ 1\ 1$$

z których tylko błąd postaci $r_{17} = 1 + x + x^2 + x^4$ jest niewykrywalny, bo jest równy $g(x)$. Wszystkie inne wektory są niepodzielne przez $g(x)$.

Kod cykliczny, generowany przez $g(x) = (1 + x) \cdot g_1(x)$, może wykryć również 4 błędy występujące w 2 paczkach po 2. Błąd taki można wyrazić w postaci

$$x^i + x^{i+1} + x^j + x^{j+1} = (x+1)(x^i+x^j)$$

Po podzieleniu tego wyrażenia przez $(1 + x) \cdot g_1(x)$ otrzymamy

$$\frac{x^i + x^j}{g_1(x)} = \frac{x^i(1 + x^{j-i})}{g_1(x)}.$$

Ostatnie wyrażenie daje jakąś resztę, jeśli $n < e$, gdzie e jest wykładnikiem, do którego należy $g_1(x)$, a więc omawiany rodzaj błędu jest wykrywalny.

6. Ciała Galois i tworzenie kodów cyklicznych korygujących błędy

Rozważania dotyczące ciał Galois rozpoczniemy od krótkiego omówienia 2 zbiorów liczb naturalnych, a mianowicie zbioru (mod 4) i zbioru (mod 5). Utwórzmy tabelki dodawania i mnożenia dla obu zbiorów:

a) (mod 4)

+	0	1	2	3
0	0	1	2	3
1	1	2	3	0
2	2	3	0	1
3	3	0	1	2

·	0	1	2	3
0	0	0	0	0
1	0	1	2	3
2	0	2	0	2
3	0	3	2	1

b) (mod 5)

+	0	1	2	3	4
0	0	1	2	3	4
1	1	2	3	4	0
2	2	3	4	0	1
3	3	4	0	1	2
4	4	0	1	2	3

·	0	1	2	3	4
0	0	0	0	0	0
1	0	1	2	3	4
2	0	2	4	1	3
3	0	3	1	4	2
4	0	4	3	2	1

Pierwszy zbiór jest zbiorem reszt w wyniku dzielenia dowolnej liczby całkowitej przez 4. Na podstawie tabelki dodawania i mnożenia i w oparciu o tabelę struktur (str.158) wnosimy, że zbiór ten jest pierścieniem. Drugi zbiór jest zbiorem reszt dzielenia liczb całkowitych przez 5. Zbiór ten jest ciałem, jeśli zastosować dodawanie i mnożenie według modułu 5.

Jeśli rząd ciała jest równy liczbie pierwszej lub potędze liczby pierwszej, to jest to tzw. ciało Galois, oznaczane symbolem GF. Z powyższych przykładów widać, że w tym drugim przypadku, kiedy rząd ciała = p^m, elementami nie mogą być liczby całkowite (mod p^m), czyli reszty dzielenia przez p^m.

Rozpatrzmy teraz zbiór reszt z dzielenia dowolnego wielomianu o współczynnikach przy zmiennej równych 0 lub 1

a) przez wielomian rozkładalny: $1+x^3=(1+x)(1+x+x^2)$,

b) przez wielomian nierozkładalny: $1+x+x^3$.

Otrzymujemy następujące zbiory wielomianów reszt

$(\mathrm{mod}\ 1+x^3)$	$(\mathrm{mod}\ 1+x+x^3)$
$0 = 0$	$0 = 0$
$1 = 1$	$1 = 1$
$A = x$	$a = x$
$B = 1+x$	$b = 1+x$
$C = x^2$	$c = x^2$
$D = 1+x^2$	$d = 1+x^2$
$E = x+x^2$	$e = x+x^2$
$F = 1+x+x^2$	$f = 1+x+x^2$

Np. $x^5:(x^3+1)=x^2$, $R = x^2$ $\qquad$ $x^5:(x^3+x+1)=x^2+1$, $R=x^2+x+1$

$$\frac{x^5+x^2}{x^2} \qquad \frac{x^5+x^3+x^2}{x^3+x^2}$$

$$\frac{x^3+x+1}{x^2+x+1}$$

Reszty te - dla uproszczenia zapisu - oznaczamy literami i tworzymy tabelkę dodawania i mnożenia:

a) $(\mathrm{mod}\ 1 + x^3)$

+	0	1	A	B	C	D	E	F
0	0	1	A	B	C	D	E	F
1	1	0	B	A	D	C	F	E
A	A	B	0	1	E	F	C	D
B	B	A	1	0	F	E	D	C
C	C	D	E	F	0	1	A	B
D	D	C	F	E	1	0	B	A
E	E	F	C	D	A	B	0	1
F	F	E	D	C	B	A	1	0

∘	0	1	A	B	C	D	E	F
0	0	0	0	0	0	0	0	0
1	0	1	A	B	C	D	E	F
A	0	A	C	E	1	B	D	F
B	0	B	E	D	D	E	B	0
C	0	C	1	D	A	E	B	F
D	0	D	B	E	E	B	D	0
E	0	E	D	B	B	D	E	0
F	0	F	F	0	F	0	0	F

b) $(\mathrm{mod}\ 1 + x + x^3)$

+	0	1	a	b	c	d	e	f
0	0	1	a	b	c	d	e	f
1	1	0	b	a	d	c	f	e
a	a	b	0	1	e	f	c	d
b	b	a	1	0	f	e	d	c
c	c	d	e	f	0	1	a	b
d	d	c	f	e	1	0	b	a
e	e	f	c	d	a	b	0	1
f	f	e	d	c	b	a	1	0

∘	0	1	a	b	c	d	e	f
0	0	0	0	0	0	0	0	0
1	0	1	a	b	c	d	e	f
a	0	a	c	e	b	1	f	d
b	0	b	e	d	f	c	1	a
c	0	c	b	f	e	a	d	1
d	0	d	1	c	a	f	b	e
e	0	e	f	1	d	b	a	c
f	0	f	d	a	1	e	c	b

Jest więc np.

1) $D + D = (1 + x^2) + (1 + x^2) = 0$

2) $D + F = (1 + x^2) + (1 + x + x^2) = x = A$

3) $D \circ E = (1 + x^2) \circ (x + x^2) = 1 + x^2$.

bo

$$\frac{(1 + x^2) \circ (x + x^2)}{x \qquad + x^3}$$

$$\frac{x^2 \qquad\qquad + x^4}{x + x^2 \quad + x^3 \quad + x^4}$$

$$(x^4+x^3+x^2+x) : (x^3+1) = x + 1$$

$$\underline{x^4 \qquad +x}$$
$$x^3+x^2$$

$$\underline{x^3 \qquad +1}$$
$$x^2 \quad +1$$

4) $d \cdot e = (1+x^2)\cdot(x+x^2) = x+x^2+x^3+x^4 = 1+x \pmod{1+x+x^3}$,

bo $(x^4+x^3+x^2+x) : (x^3+x+1) = x+1$

$$\underline{x^4 \quad +x^2+x}$$
$$x^3$$

$$\underline{x^3 \quad +x+1}$$
$$x+1$$

Z tych tabelek widać, że zbiór wielomianów (mod wielomian rozkładalny $1 + x^3$) jest pierścieniem, a zbiór wielomianów (mod wielomian nierozkładany $1 + x + x^3$) jest ciałem. Rząd tego ciała jest równy $2^3 = 8$; jest to przykład ciała Galois.

Przykład wielomianu $p(x) = 1 + x + x^3$ posłuży nam do dalszych rozważań. I tak pierwiastkiem tego wielomianu, tj. pierwiastkiem równania $p(x) = 1 + x + x^3 = 0$, jest element α taki, że $\alpha^3 = 1 + \alpha$. Element ten nazywa się elementem pierwotnym, bo każdy niezerowy element ciała wielomianów (mod $1+x+x^3$) da się wyrazić jako potęga tego elementu. Mamy mianowicie

$$\alpha^0 = 1$$
$$\alpha^1 = \alpha$$
$$\alpha^2 = \alpha^2$$
$$\alpha^3 = 1+\alpha$$
$$\alpha^4 = \alpha+\alpha^2$$
$$\alpha^5 = 1+\alpha+\alpha^2$$
$$\underline{\alpha^6 = 1 \quad +\alpha^2}$$
$$\alpha^7 = 1$$

Rząd elementu α jest równy 7. Ponieważ równanie $1+x+x^3 = 0$ jest 3 stopnia, więc musi mieć 3 pierwiastki. Są nimi: α, α^2 i α^4. Łatwo się przekonać o tym, że np. α^2 jest pierwiastkiem równania; podnosząc do kwadratu równanie $1+\alpha+\alpha^3 = 0$:

$$(1 + \alpha + \alpha^3)^2 = 0$$
$$1 + \alpha^2 + \alpha^6 = 0$$
$$1 + (\alpha^2) + (\alpha^2)^3 = 0$$

Wykorzystuje się tu właściwość, że dla elementów a,b z $GF(2^m)$ zachodzi

$$(a+b)^2 = a^2 + b^2$$

Wielomian $p(x) = 1 + x + x^3$ jest dla elementu α tzw. minimalną funkcją $m(x)$, tzn. że ten wielomian jest wielomianem najniższego stopnia, takim, że $m(\alpha) = 0$. Tym samym jest to minimalna funkcja dla α^2 i α^4. Dla elementów α^3, α^6, $\alpha^{12} = \alpha^5$ minimalną funkcją jest $1+x^2+x^3$, a dla elementu 1 taką funkcją jest $1 + x$. Można to ująć w tabelkę:

elementy $GF(2^3)$	$m(x)$
1	$m_0(x) = 1 + x$
α	$m_1(x) = 1 + x + x^3$
α^2	$m_1(x) = 1 + x + x^3$
α^4	$m_1(x) = 1 + x + x^3$
α^3	$m_3(x) = 1 + x^2 + x^3$
α^6	$m_3(x) = 1 + x^2 + x^3$
$\alpha^{12} = \alpha^{12-7} = \alpha^5$	$m_3(x) = 1 + x^2 + x^3$

elementy $GF(2^3)$	m(x)
1 i α	$m_0(x) \cdot m_1(x) = (1+x)(1+x+x^3)$
1 i α^3	$m_0(x) \cdot m_3(x) = (1+x)(1+x^2+x^3)$
α i α^3	$m_1(x) \cdot m_3(x) = (1+x+x^3)(1+x^2+x^3)$
1, α i α^3	$m_0(x) \cdot m_1(x)\, m_3(x) =$ $(1+x)(1+x+x^3)(1+x^2+x^3) = 1 + x^7$

Ostatni wiersz tej tabelki mówi, że zbiór wszystkich elementów ciała jest zbiorem pierwiastków wielomianu $x^7 + 1$, a więc

$$x^7 + 1 = (x+1)(x+\alpha)(x+\alpha^2)(x+\alpha^3)(x+\alpha^4)(x+\alpha^5)(x+\alpha^6).$$

Rozważania te można uogólnić w następujący sposób:

Dla każdej liczby naturalnej m istnieje wielomian pierwotny p(x), stopnia m, jest to taki wielomian nierozkładalny, którego pierwiastkiem jest element pierwotny ciała wielomianów mod p(x), rzędu 2^m, przy założeniu że rozpatrujemy tylko wielomiany o współczynnikach 0 i 1, tj. - inaczej mówiąc - wielomiany nad GF 2. Rząd elementu pierwotnego jest równy $q-1=2^m-1$ i każdy niezerowy element może być wyrażony jako potęga tego elementu. Dalej wszystkie elementy ciała $GF(2^m)$ są pierwiastkami równania

$$x^{2^m} + x = 0,$$

czyli że wszystkie niezerowe elementy ciała $GF(2^m)$ są pierwiastkami równania

$$x^{2^m-1} + 1 = 0$$

Dalej wielomian p(x) jest minimalną funkcją dla elementu $\alpha, \alpha^2, \alpha^4, \alpha^8, \alpha^{16} \ldots$. Dla elementu ciała $\alpha^3, \alpha^6, \alpha^{12} \ldots$

oraz $\alpha^5, \alpha^{10}, \alpha^{20}, \ldots$ itd. istnieją minimalne funkcje stopnia nie przewyższającego liczby m.

Po tych rozważaniach można podać "przepis" na tworzenie kodów cyklicznych korygujących t błędów, a więc kodów o minimalnej wadze $w = 2t + 1$. Będą to tzw. kody Bose-Chaudhuri-Hocquenghema [3], [4], [6]. Wielomian $f(x)$ jest wielomianem kodowym, jeśli elementy $\alpha, \alpha^2, \alpha^3, \alpha^4, \ldots, \alpha^{2t}$ są pierwiastkami tego wielomianu. Wykorzystując zaś fakt, że każda parzysta potęga ma tę samą minimalną funkcję co któraś nieparzysta potęga

(np. $\alpha, \alpha^2, \alpha^4, \alpha^8, \ldots$ mają tę samą min. funkcję $m_1(x)$
$\alpha^3, \alpha^6, \alpha^{12}, \ldots$ " " " " " $m_3(x)$
$\alpha^5, \alpha^{10}, \alpha^{20}, \ldots$ " " " " " $m_5(x)$
$\alpha^7, \alpha^{14}, \ldots$ " " " " " $m_7(x)$

możemy powiedzieć, że $f(x)$ jest wielomianem kodowym, jeśli elementy $\alpha, \alpha^3, \alpha^5 \ldots \alpha^{2t-1}$ są pierwiastkami tego wielomianu. A więc generatorem kodu cyklicznego jest

$$g(x) = \left[\text{NWW}\ m_1(x), m_3(x), m_5(x), \ldots m_{2t-1}(x)\right].$$

Stopień wielomianu $m_i(x)$ jest równy lub mniejszy niż m, gdzie m jest stopniem wielomianu pierwotnego $p(x)$, którego pierwiastkiem jest element α. To zaś wynika z tego, że każdy czynnik wielomianu ma stopień m lub mniejszy niż m.

Tak np.

$$1+x^{15} = (1+x)(1+x+x^4)(1+x+x^2+x^3+x^4)(1+x+x^2)(1+x^3+x^4)$$

wobec tego będzie:

elementy	minimalna funkcja	
1	$1+x$	$= m_0(x)$
$\alpha, \alpha^2, \alpha^4, \alpha^8,$	$1+x+x^4$	$= m_1(x)$
$\alpha^3, \alpha^6, \alpha^{12}, \alpha^{24} = \alpha^9$	$1+x+x^2+x^3+x^4$	$= m_3(x)$
α^5, α^{10}	$1+x+x^2$	$= m_5(x)$
$\alpha^7, \alpha^{14}, \alpha^{28} = \alpha^{13}, \alpha^{26} = \alpha^{11}$	$1+x^3+x^4$	$= m_7(x)$

Kod znajdujemy z tabel (p. np. [3]), gdzie podana jest

a) długość kodu $= n$,

b) liczba pozycji informacyjnych $= k$,

c) liczba korygowalnych błędów $= t$.

Taką tabelę buduje się w następujący sposób:

Dla danego m wyszukujemy wielomian pierwotny $p(x)$ stopnia m. Warto zwrócić uwagę na to, że np. wielomiany: $1+x+x^2+x^3+x^4$ lub $1+x^3+x^6$ są wprawdzie nierozkładalne, ale nie są pierwotne. Znając m znajdujemy długość kodu n

$$n = 2^m - 1.$$

Dalej określamy liczbę błędów t, pierwiastki $\alpha, \alpha^3, \ldots \alpha^{2t-1}$ i ich minimalne funkcje, tj. $m_1(x), m_3(x), \ldots m_{2t-1}(x)$. Iloczyn tych funkcji daje wielomian generujący $g(x)$.

$$g(x) = \text{NWW}\left[m_1(x), m_3(x), \ldots m_{2t-1}(x)\right],$$

którego stopień n-k określa liczbę pozycji informacyjnych k. Wobec tego liczba wszystkich wektorów kodowych jest równa 2^k. Przykład takiej tabeli podany jest na str. 185-187.

Funkcję m_i znajdujemy wg wzoru

$$m_i = (x-\alpha^i)(x-\alpha^{2i})(x-\alpha^{4i}) \ldots$$

Np. dla m = 5, m_9 byłoby równe

$$m_9 = (x-\alpha^9)(x-\alpha^{18})(x-\alpha^5)(x-\alpha^{10})(x-\alpha^{20})$$

Łatwo zauważyć (tabela na str.186), że m_9 jest równe m_5 i wobec tego czynnikami g(x) przy t = 6 są $m_1(x)$, $m_3(x)$, $m_5(x)$, $m_7(x)$ i $m_{11}(x)$.

Przykład kodu (15,5), umożliwiającego korekcję 3 błędów i wykrywającego paczki błędów o długości 10, podany jest na str. 169.

m	n	p(x)	t	pierwiastki f(x)	czynniki g(x)	stopień g(x)	(n, k)
3	7	$1+x+x^3$	1	$\alpha, \alpha^2, \alpha^4$	$m_1(x)$	3	(7,4)
4	15	$1+x+x^4$	1	$\alpha, \alpha^2, \alpha^4, \alpha^8$	$m_1(x)$	4	(15,11)
			2	$\alpha, \alpha^2, \alpha^4, \alpha^8$ $\alpha^3, \alpha^6, \alpha^{12}, \alpha^9$	$m_1(x)$ $m_3(x)$	8	(15,7)
			3	$\alpha, \alpha^2, \alpha^4, \alpha^8$ $\alpha^3, \alpha^6, \alpha^{12}, \alpha^9$ α^5, α^{10}	$m_1(x)$ $m_3(x)$ $m_5(x)$	10	(15,5)

m	n	p(x)	t	pierwiastki f(x)	czynniki g(x)	stopień g(x)	(n, k)
5	31	$1+x^2+x^5$	1	$\alpha, \alpha^2, \alpha^4, \alpha^8, \alpha^{16}$	$m_1(x)$	5	(31,26)
			2	$\alpha, \alpha^2, \alpha^4, \alpha^8, \alpha^{16}$ $\alpha^3, \alpha^6, \alpha^{12}, \alpha^{24}, \alpha^{17}$	$m_1(x)$ $m_3(x)$	10	(31,21)
			3	$\alpha, \alpha^2, \alpha^4, \alpha^8, \alpha^{16}$ $\alpha^3, \alpha^6, \alpha^{12}, \alpha^{24}, \alpha^{17}$ $\alpha^5, \alpha^{10}, \alpha^{20}, \alpha^9, \alpha^{18}$	$m_1(x)$ $m_3(x)$ $m_5(x)$	15	(31,16)
			4	$\alpha\ \alpha^2\ \alpha^4\ \alpha^8\ \alpha^{16}$ $\alpha^3\ \alpha^6\ \alpha^{12}\ \alpha^{24}\ \alpha^{17}$ $\alpha^5\ \alpha^{10}\ \alpha^{20}\ \alpha^9\ \alpha^{18}$ $\alpha^7\ \alpha^{14}\ \alpha^{28}\ \alpha^{15}\ \alpha^{19}$	$m_1(x)$ $m_3(x)$ $m_5(x)$ $m_7(x)$	20	(31,11)
			5	$\alpha, \alpha^2, \alpha^4, \alpha^8, \alpha^{16}$ $\alpha^3, \alpha^6, \alpha^{12}, \alpha^{24}, \alpha^{17}$ $\alpha^5, \alpha^{10}, \alpha^{20}, \alpha^9, \alpha^{18}$ $\alpha^7, \alpha^{14}, \alpha^{28}, \alpha^{25}, \alpha^{19}$ $(\alpha^9, \alpha^{18}, \alpha^5, \alpha^{10}, \alpha^{20})$	$m_1(x)$ $m_3(x)$ $m_5(x)$ $m_7(x)$	20	(31,11)
			6	$\alpha, \alpha^2, \alpha^4, \alpha^8, \alpha^{16}$ $\alpha^3, \alpha^6, \alpha^{12}, \alpha^{24}, \alpha^{17}$ $\alpha^5, \alpha^{10}, \alpha^{20}, \alpha^9, \alpha^{18}$	$m_1(x)$ $m_3(x)$ $m_5(x)$		

m	n	p(x)	t	pierwiastki f(x)	czynniki g(x)	stopień g(x)	(n,k)
				$\alpha^7, \alpha^{14}, \alpha^{28}, \alpha^{25}, \alpha^{19}$ $\alpha^{11}, \alpha^{22}, \alpha^{13}, \alpha^{26}, \alpha^{21}$ $(\alpha^9, \alpha^{18}, \alpha^5, \alpha^{10}, \alpha^{20})$	$m_7(x)$ $m_{11}(x)$	25	(31,6)

LITERATURA

[1] Hamming R.W.: Error Detecting and Correcting Codes, BSTJ, April 1950.

[2] Slepian D.A.: A Class of Binary Signaling Alphabets. BSTJ, Jan 1956.

[3] Peterson W.W.: Error Correcting Codes. MIT Press, 1961.

[4] Udałow A., Suprun B.: Izbytocznoje Kodirowanije. Swiaz 1964.

[5] Peterson W.W., Brown D.T.: Cyclic Codes for Error Detection. PIRE, Jan 1961.

[6] Bose R.C., Ray Chaudhuri D.K.: On a Class of Error Correcting Binary Group Codes. Inf. and Control, 3, 1960.

[7] Abramson N.: Information and Coding. McGraw Hill, 1963.

[8] Birkhoff G., Mac Lane S.: Przegląd algebry współczesnej. PWN 1960.

Rękopis złożono w Redakcji w dniu 7.XII.65 r.

КОЛЬЦА МНОГОЧЛЕНОВ И ПОЛЯ ГАЛУА В ПРИМЕНЕНИИ ДЛЯ СИНТЕЗА ЦИКЛИЧЕСКИХ КОДОВ

Р е з ю м е

Вопрос синтеза линейных бинарных корректирующих кодов можно представить следующим спообом: дана длина кода а также известно число ошибок t, которые следует исправить; нужно найти код, в котором число информационных знаков k будет наибольшее, при чем вес w каждой кодовой последовательности должен выполнять условие $w \geq 2t + 1$. Для решения этого вопроса применяется алгебраические структуры такие, как группы, кольца, поля и векторные пространства. В настоящей работе содержится систематический обзор этих структур, в отдельности, кольцов многочленов, а также полей Галуа, служащих математическим обоснованием для теории циклических кодов. Работа проиллюстрирована примерами.

THE USE OF POLYNOMIAL RINGS AND GALOIS FIELDS FOR CYCLIC CODES CONSTRUCTION

S u m m a r y

The problem of construction linear binary correcting codes is as follows: given the length n of a code word and the number t of errors to be corrected, a code is to be found in which the number of information symbols k is maximum; the Hamming weight w for each n-tuple of this code must fulfil the relation $w \geq 2t + 1$. As a tool for solving this problem algebraic structures such as groups, rings, fields and vector spaces are used. This paper gives a systematic review of all these structures with an emphasis on polynomial rings and Galois fields which form mathematical basis for cyclic codes. Many examples are given for illustration.