

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ЗАХИСТУ ІНФОРМАЦІЇ
КАФЕДРА ІНФОРМАЦІЙНОЇ ТА КІБЕРНЕТИЧНОЇ БЕЗПЕКИ**

Г.І Гайдур., З.З. Бондаренко, В.В. Марченко, Н.С. Чумак

**МЕТОДИЧНИЙ ПОСІБНИК
ЛАБОРАТОРНИЙ ПРАКТИКУМ
з навчальної дисципліни
“ТЕОРІЯ ІНФОРМАЦІЇ ТА КОДУВАННЯ”**

**для підготовки бакалаврів
галузі знань - 12 Інформаційні технології
за спеціальністю - 125 Кібербезпека**

**Затверджено Вченою радою
Навчально–наукового інституту Захисту інформації
Державного університету телекомунікацій,
протокол № __ від _____ 2021 р.**

Київ - 2021

Рецензенти: - професор кафедри Мобільних та відеоінформаційних технологій
ННІТ ДУТ, доктор технічних наук, професор А.О. Макаренко
- доцент кафедри Телекомунікаційних систем та мереж ННІТ ДУТ,
кандидат технічних наук, доцент О.Г. Варфоломєєва

Методичний посібник призначений для підготовки бакалаврів вищих навчальних закладів з навчальної дисципліни “Теорія інформації та кодування”- циклу дисциплін загальної підготовки галузі знань - 12 Інформаційні технології, спеціальності - 125 Кібербезпека.

Навчальна дисципліна “Теорія інформації та кодування” вивчається протягом одного семестру. Матеріал відповідає програмі дисципліни.

Г.І Гайдур., З.З. Бондаренко, В.В. Марченко, Н.С. Чумак. Лабораторний практикум з навчальної дисципліни “Теорія інформації та кодування”
Навчальний посібник для студентів вищих навчальних закладів – Київ: ДУТ, ННІЗІ, 2021. – 50с.

Методичний посібник містить вказівки до виконання лабораторних робіт з навчальної дисципліни «Теорія інформації та кодування».

До кожної лабораторної роботи надано основні теоретичні положення, завдання та алгоритм виконання за допомогою програмного середовища Matlab. Питання для самоперевірки рівня знань студентів сприяють закріпленню теоретичного матеріалу.

В додатках наведено приклад оформлення звіту лабораторної роботи, терміни та визначення, які повинні розуміти та знати студенти при виконанні лабораторних робіт. Також представлено матеріали для ознайомлення з програмним пакетом Matlab, як універсальним забезпеченням для дослідження, розрахунків та моделювання різноманітних процесів, що дозволяють більш ефективно та практично використовувати навчальний час, дають змогу твердому закріпленню теоретичного матеріалу за рахунок гнучкості та доступності, економічно використовувати обладнання.

ЗМІСТ

| | |
|---|----|
| ВСТУП | 4 |
| Організація та проведення лабораторних робіт | 5 |
| ЛАБОРАТОРНА РОБОТА 1. Модуляція, методи модуляції. Двосмугова амплітудна модуляція. Ознайомлення з середовищем моделювання MATLAB. Спектри АМ сигналів при однотонольній модуляції | 6 |
| ЛАБОРАТОРНА РОБОТА 2. Дослідження частотної модуляції неперервного сигналу..... | 14 |
| ЛАБОРАТОРНА РОБОТА 3. Цифрові методи передачі безперервних повідомлень. Імпульсно-кодова модуляція..... | 19 |
| ЛАБОРАТОРНА РОБОТА 4. Код Хемінга..... | 24 |
| ЛАБОРАТОРНА РОБОТА 5. Циклічний код..... | 29 |
| ЛІТЕРАТУРА | 39 |
| Додаток А. Приклад оформлення титульної сторінки звіту лабораторної роботи..... | 40 |
| Додаток Б. Особливості роботи в середовищі моделювання MATLAB..... | 41 |
| Додаток В. Терміни та визначення, які повинні розуміти та знати студенти при виконанні лабораторних робіт..... | 47 |

ВСТУП

Метою проведення лабораторних робіт з навчальної дисципліни «Теорія інформації та кодування» є формування у студентів комплексу базових знань, умінь, навичок, необхідних для розуміння основних положень теорії інформації, принципів дії систем кодування інформації, використання математичних моделей процесів у своїй практичній діяльності для розв'язання виробничих, проектних та наукових задач.

При підготовці до виконання роботи студент повинен вивчити відповідний теоретичний матеріал, передбачений навчальною програмою, ознайомитись з описом лабораторної роботи.

Завданнями проведення лабораторних робіт є формування у студентів наступних умінь:

- уміти характеризувати типові сигнали та завади, характеризувати та оцінювати основні характеристики сигналів, аналізувати переваги і недоліки конкретних видів модуляції, проводити розрахунки характеристик сигналів, чітко характеризувати процес дискретизації сигналів за теоремою Котельникова;

- уміти розуміти сутність та оцінювати отримані результати, диференціювати, інтегрувати, уніфікувати, оцінювати зміст теорії передачі інформації, проводити аналітичні експерименти із використанням математичних моделей;

- уміти аналізувати процес завадостійкості систем зв'язку з різними видами модуляції, проводити аналіз окремих технічних рішень і прогнозувати очікувані результати.

Лабораторні заняття передбачають набуття навичок роботи з програмним середовищем MATLAB, як універсальним забезпеченням при дослідженні модуляції, спектрального представлення сигналів та дослідженні аналого-цифрових перетворень.

Під час захисту лабораторних робіт студент повинен показати опанування основними термінами, базовими знаннями із відповідних розділів курсу.

Методичний посібник має на меті забезпечити якісне виконання лабораторних робіт студентами, а також допомогти їм закріпити теоретичні знання основних розділів навчальної дисципліни «Теорія інформації та кодування» та виробити навички свідомого застосування правил, методів, принципів, законів у конкретних ситуаціях, практичного виконання розрахунків та експериментальних досліджень, використовуючи сучасні обчислювальні засоби, можливість своєчасно адаптуватися до зростаючого потоку інформації, розвитку галузі інформаційних технологій, спеціальності кібербезпека.

Організація та проведення лабораторних занять

Лабораторні роботи студенти виконують у два етапи. Перший етап – підготовчий, вимагає з'ясування завдання лабораторної роботи, повторення теоретичних знань з її тематики. На першому етапі потрібно ознайомитись з необхідним програмним забезпеченням для виконання конкретної лабораторної роботи.

Другий етап – виконання лабораторної роботи потребує здійснення експериментальних досліджень, перевірки та оцінення отриманих результатів. Обов'язково потрібно порівняти результати якісно чи кількісно з теоретичними положеннями, зробити висновки. Після цього складають звіт про виконану роботу.

Звіт має обов'язково містити: назву лабораторної роботи; дату виконання роботи; прізвище та ініціали студента; шифр групи; мету лабораторної роботи; структурну схему дослідження; програму для MATLAB, графіки, висновки за результатами роботи.

ЛАБОРАТОРНА РОБОТА №1

МОДУЛЯЦІЯ, МЕТОДИ МОДУЛЯЦІЇ . ДВОСМУГОВА АМПЛІТУДНА МОДУЛЯЦІЯ. СПЕКТРИ АМ СИГНАЛІВ ПРИ ОДНОТОНАЛЬНІЙ МОДУЛЯЦІЇ. ОЗНАЙОМЛЕННЯ З СЕРЕДОВИЩЕМ МОДЕЛЮВАННЯ MATLAB (4 години)

Мета роботи

1. Ознайомитися з середовищем моделювання MATLAB, особливостями створення та відображення сигналів.
2. Дослідити фізичні процеси при амплітудній модуляції.
3. Дослідити спектр АМ сигналу.

Теоретичні відомості

Визначення модуляції. **Модуляція** (лат. *modulatio* - розміреність, ритмічність) - процес зміни одного або декількох параметрів високочастотного несучого коливання за законом низькочастотного інформаційного сигналу (повідомлення).

Інформація, що передається, закладена в керуючому (модулюючому) сигналі, а роль носія інформації виконує високочастотне коливання, що має назву несучого. Модуляція, таким чином, являє собою процес «насадки» інформаційного коливання на заздалегідь відому несучу.

В результаті модуляції спектр низькочастотного керуючого сигналу переноситься в область високих частот. Це дозволяє при організації мовлення налаштувати функціонування всіх приймально-передавальних пристроїв на різних частотах для того, щоб вони «не заважали» один одному.

У ролі несучого коливання можуть бути використані коливання різної форми (гармонійні, прямокутні, трикутні і т. д.), однак найчастіше застосовуються гармонійні коливання. В залежності від того, який з параметрів несучого коливання змінюється, розрізняють види модуляції, а саме: амплітудна, частотна, фазова та ін. Модуляція дискретним сигналом називається *цифровою модуляцією* або *маніпуляцією*.

Розрізняють наступні види модуляції:

- Амплітудна модуляція (АМ);
- Амплітудна модуляція з однією бічною смугою (SSB - односмугова АМ);
- Балансна амплітудна модуляція (БАМ) - АМ з придушенням несучої;
- Частотна модуляція (ЧМ);

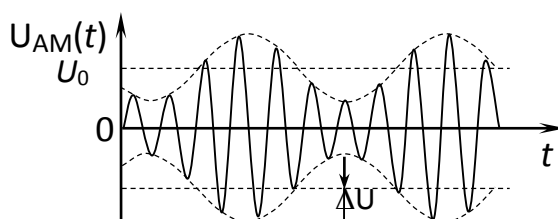
- Фазова модуляція (ФМ).

У радіотехніці широкого поширення набули системи модуляції, що використовують в якості несучого просте гармонійне коливання $S(t) = U_0 \cos(\omega_0 t + \varphi_0)$ тоді

$$U_{AM}(t) = U_0 [1 + MU(t)] \cos(\omega_0 t + \varphi_0),$$

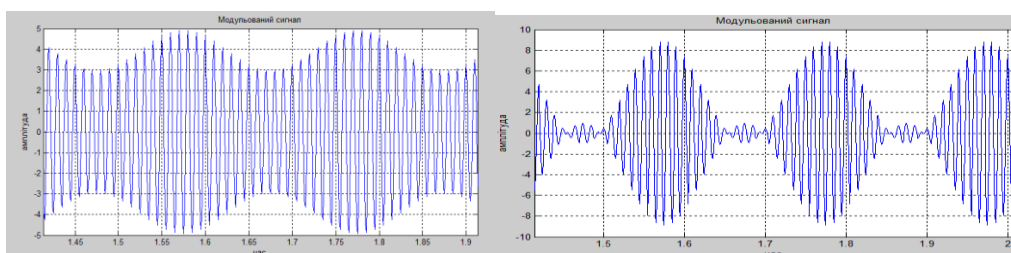
де $U_0 [1 + MU(t)] = U(t)$ – огибающая несучої

U_0 – постійний коефіцієнт, рівний амплітуді несучого коливання без модуляції ΔU – амплітуда зміни огибающей, M – коефіцієнт амплітудної модуляції $M = \Delta U / U_0$ – модулюючий корисний сигнал.



При малій глибині модуляції, тобто $M < 1$, відносна зміна обвідної невелика. Якщо ж є наближена рівність $M \approx 1$, то говорять про глибоку модуляцію.

Максимальне збільшення ΔU не повинно перевищувати значення U_0 . Інакше з'являються спотворення сигналу (перемодуляція), при якій форма обвідної перестає повторювати форму модулюючого сигналу. Для того, щоб спотворень не було, необхідно виконання умови $M \leq 1$.



$M < 1$

$M > 1$

Найпростіший АМ - сигнал може бути отриманий у разі, коли модулюючим низькочастотним сигналом є гармонійне коливання з частотою Ω .

$$U(t) = U \cos(\Omega t + \gamma).$$

АМ - сигнал, модульований одним гармонійним коливанням, називається однотональним і може бути представлений як сума простих гармонійних коливань з різними частотами.

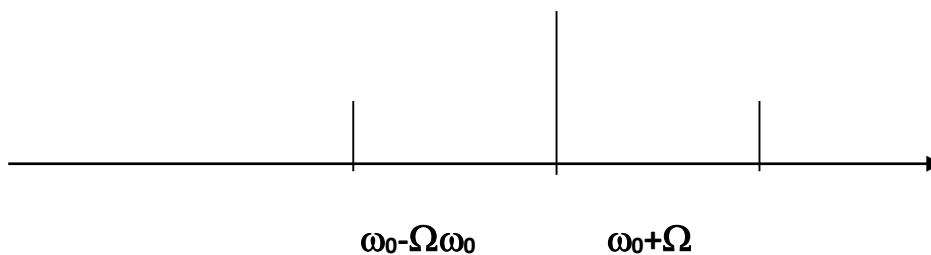
$$U_{AM}(t) = U_0 [1 + M \cos(\Omega t + \gamma)] \cos(\omega_0 t + \varphi_0)$$

$$U_{AM}(t) = U_0 \cos(\omega_0 t + \varphi_0) + U_0 M \cos(\Omega t + \psi) \cos(\omega_0 t + \varphi_0)$$

Запишемо добуток косинусів у вигляді суми

$$U_{AM}(t) = U_0 \cos(\omega_0 t + \varphi_0) + \frac{U_0 M}{2} \cos[(\omega_0 + \Omega)t + \varphi_0 + \gamma] + \frac{U_0 M}{2} \cos[(\omega_0 - \Omega)t + \varphi_0 - \gamma].$$

Ця формула встановлює спектральний склад однотонального АМ коливання, де ω_0 -несуча частота, $\omega_0 + \Omega$ -верхня бокова частота, $\omega_0 - \Omega$ -нижня бокова частота, 2 і 3 доданки відповідають новим коливанням, що з'явилися в процесі модуляції амплітуди



Тобто, в склад спектру АМ сигналів при однотональній модуляції входять три складові:

1. Немодульована несуча $U_0(t) = U_0 \cos(\omega_0 t + \varphi_0)$.
2. Верхня бокова частота $\frac{U_0 M}{2} \cos((\omega_0 + \Omega)t + \varphi_0 + \gamma)$.
3. Нижня бокова частота $\frac{U_0 M}{2} \cos((\omega_0 - \Omega)t + \varphi_0 - \gamma)$.

Будуючи за формулою спектральну діаграму однотонального АМ-сигналу, слід перш за все звернути увагу на рівність амплітуд верхнього і нижнього бокових коливань, а також на симетрію розташування цих спектральних складових щодо несучого коливання.

Середня потужність АМ-сигналу дорівнює сумі потужностей несучого і бічних коливань:

$$P_{AM} = P_0 + P_{НБ} + P_{ВБ}$$

При коефіцієнті амплітудної модуляції, що дорівнює одиниці, верхня і нижня бічні смуги мають амплітуди, рівні половині несучої амплітуди. Таким чином, третя частина потужності передавача витрачається на передачу корисної

інформації, а дві третини - на випромінювання несучої, що не несе корисної інформації.

Переваги амплітудної модуляції:

- смуги частот, які займає амплітудо-модульований сигнал, досить вузькі, тобто займають менший частотний діапазон;
- реалізація пристроїв нескладна.

Недоліки амплітудної модуляції:

- середня потужність сигналу набагато менша за пікову, тобто апаратура використовується не на повну потужність;
- перешкодозахищеність нижча ніж при ЧМ та ФМ.

Завдання

Дано:

1. Низькочастотне гармонійне коливання $U_{\text{инф}}(t) = U_{\text{инф}}\cos(\Omega t + \varphi)$ (відповідно варіанту, табл.1)
2. Високочастотне несуче коливання $U_0(t) = U_0\cos(\omega_0 t + \varphi_0)$ (відповідно варіанту, табл.1)

Отримати амплітудно-модульоване коливання та графіки модулюючого, несучого, модульованого сигналу та спектру АМ сигналу.

Порядок виконання роботи

1. Ознайомитися з основними принципами модуляції сигналів. Ознайомитись з програмою Matlab.
2. Запустити програму Matlab за допомогою ярлика на робочому столі.
3. Відкрити NewScript.
4. Запрограмувати значення модулюючого, несучого та модульованого сигналу.
5. За допомогою Matlab отримати графіки модулюючого, несучогої амплітудно-модульованого сигналу.
6. За допомогою Matlab отримати графік спектру АМ сигналу.
7. Зробити висновки.

Зміст звіту

1. Лістинг програми;
2. Всі необхідні графічні побудови.
3. Висновки.

% Виведення графіків на екран

figure (1) % створення першого вікна для виведення графіку

plot (____) %виведення графіку сигналу

gridon % включення відображення сітки

title (_____) % назва графіку

ylabel (__) % підпис осі y

xlabel (____) % підпис осі x

holdon % ввімкнення накладення графіків

Контрольні запитання

1. Дайте визначення модуляції, для чого вона застосовується?
2. Які види модуляції вам відомі?
3. Дайте визначення амплітудної модуляції.
4. Як визначити коефіцієнт амплітудної модуляції, який він повинен бути?
5. Що таке перемодуляція?
6. Який спектр АМ сигналу?
7. Назвіть переваги та недоліки амплітудної модуляції .

Зразок виконання роботи

```
% Вихідні дані для однотоноальної амплітудної модуляції
```

```
clear all
```

```
%Початкові дані
```

```
Dt=0.001;
```

```
t0 = 0;
```

```
tkinc = 5;
```

```
U0 = 4; % [Вольт]- амплітуда опорного сигналу
```

```
Uinf = 0.5; % [Вольт]- амплітуда інформаційного сигналу
```

```

M1 = 0.3; % [безрозмірна] - коефіцієнт амплітудної модуляції
Flinf = 5;% частота інформаційного сигналу
fil = 45; % [grad] - фаза інформаційного сигналу
F2op = 10^2;% частота опорного сигналу
fi0=10; % [grad] - фаза опорного сигналу

%Розрахунок
t=t0:dt:tkinc; %[с] - час передачі модульованого сигналу (частота
опорного сигналу)
Omega1=Flinf*2*pi; % [Герц] - частота інформаційного сигналу
omega0=F2op*2*pi; % [Герц] - частота опорного сигналу
rad=pi/180;
St=U0*cos(omega0*t+fi0*rad); % опорний сигнал
U1=Uinf*cos(Omega1*t+fil*rad); % інформаційний сигнал
A=1+M1*U1;

Uam=A.*St; % модульований сигнал

Fs = 1/dt; %10000; % Sampling frequency
T = 1/Fs; % Sampling period
L = Fs*tkinc+1; %100001; % Length of signal
t = (0:L-1)*T; % Time vector
Y = fft(Uam); % швидке перетворення Фур'є сигналу
P2 = abs(Y/L);
P1 = P2(1:(L/2)+1);
P1(2:end-1) = 2*P1(2:end-1);
f = Fs*(0:(L/2))/L;
figure(1)
plot(f,P1)
title('Single-Sided Amplitude Spectrum of X(t)')
xlabel('f (Hz)')
ylabel('|P1(f)|')

```

```
figure(2)
plot(t,St)
title('Опорний сигнал')
xlabel('час')
ylabel('амплітуда')
grid % сітка
holdon % закріплення графіка
plot(t,U1,'r')
```

```
figure(3)
plot(t,U1)
title('Інформаційний сигнал')
xlabel('час')
ylabel('амплітуда')
grid
```

```
figure(4)
plot(t,Uam)
title('Модульований сигнал')
xlabel('час')
ylabel('амплітуда')
grid
```

Таблиця 1 Параметри сигналів згідно з варіантом

| Варіант №п/п | Амплітуда несучого сигналу U_0 , В | Амплітуда модулюючого сигналу U , В | Частота несучого сигналу, Гц | Частота модулюючого сигналу f_c , Гц |
|--------------|--------------------------------------|---------------------------------------|------------------------------|--|
| 1 | 5 | 2 | 100 | 5 |
| 2 | 3 | 0,5 | 150 | 7 |
| 3 | 4 | 1,3 | 200 | 30 |
| 4 | 2 | 0,5 | 400 | 40 |
| 5 | 1,2 | 0,4 | 350 | 34 |
| 6 | 3 | 1,5 | 160 | 20 |
| 7 | 5 | 1,5 | 200 | 50 |
| 8 | 4 | 1,2 | 30 | 6 |
| 9 | 1 | 0,5 | 20 | 1 |
| 10 | 2 | 0,3 | 30 | 2 |
| 11 | 3 | 0,5 | 20 | 7 |
| 12 | 5 | 3 | 240 | 12 |
| 13 | 4 | 1,5 | 16 | 4 |
| 14 | 1 | 0,3 | 200 | 12 |
| 15 | 2 | 0,2 | 300 | 33 |
| 16 | 3 | 2 | 20 | 2 |
| 17 | 5 | 1,5 | 30 | 5 |
| 18 | 2 | 0,7 | 40 | 6 |
| 19 | 3 | 1,1 | 120 | 4 |
| 20 | 5 | 3 | 16 | 3 |
| 21 | 4 | 2,5 | 20 | 2 |
| 22 | 1 | 0,1 | 30 | 5 |
| 23 | 2 | 1 | 160 | 8 |
| 24 | 3 | 1,2 | 80 | 4 |
| 25 | 5 | 2 | 60 | 4 |
| 26 | 3 | 0,9 | 48 | 2 |
| 27 | 5 | 4 | 36 | 8 |
| 28 | 4 | 2 | 16 | 2 |
| 29 | 3 | 0,5 | 200 | 34 |
| 30 | 1 | 0,4 | 240 | 24 |

ЛАБОРАТОРНА РОБОТА №2

ДОСЛІДЖЕННЯ ЧАСТОТНОЇ МОДУЛЯЦІЇ НЕПЕРЕРВНОГО СИГНАЛУ

Мета роботи

- 1 Дослідити фізичні процеси при частотній модуляції.
- 2 Дослідити спектр ЧМ сигналу

Теоретичні відомості

Частотна модуляція була запропонована Едвіном Армстронгом і запатентована ним 26 грудня 1933 року.

Частотна модуляція - це зміна частоти високочастотного несучого колювання відносно його центрального значення ω_0 за законом модулюючого сигналу.

$$\omega = \omega_0 + \Delta\omega \cos\Omega t,$$

де $\cos\Omega t$ - модулюючий сигнал, $\Delta\omega$ - **девіація частоти**. **Девіація частоти** - це найбільше відхилення несучої частоти ω від її центрального значення ω_0 .

$$\omega = \omega_0 + \Delta\omega(t)$$

Амплітуда колювань при цьому залишається незмінною.

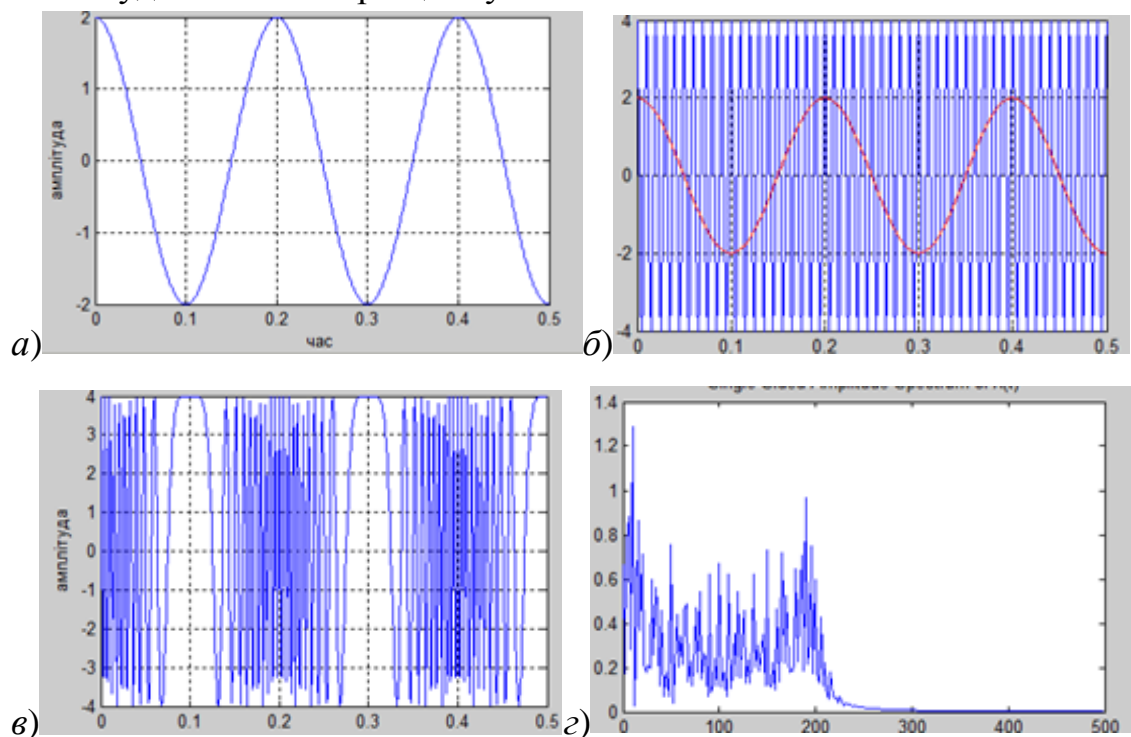


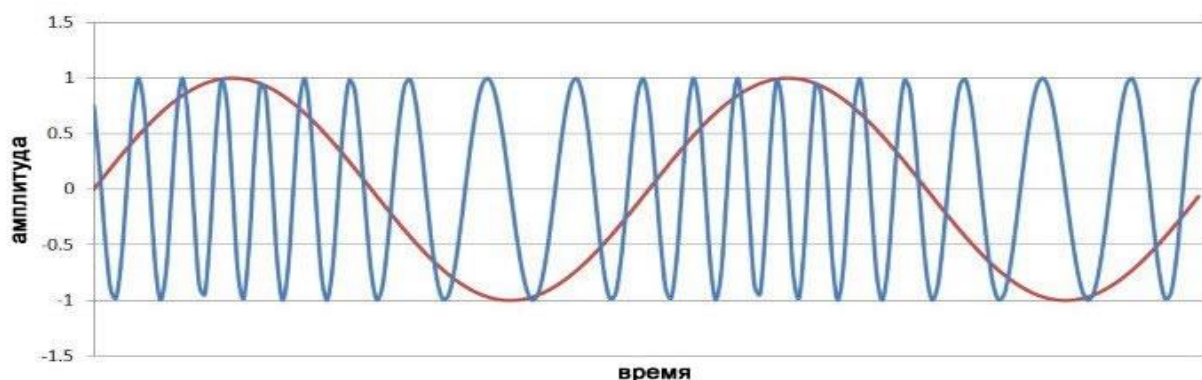
Рис. 1: *a* – модулюючий сигнал; *b* – несучий сигнал; *в* – частотно-модульований сигнал; *г* – спектр частотно-модульованого сигналу

При частотній модуляції смуга частот модульованого коливання залежить від **індексу частотної модуляції**

$$\beta = \Delta\omega/\Omega,$$

де $\Delta\omega$ - девіація частоти, Ω – частота модулюючого сигналу.

$$U_{\text{ЧМ}}(t) = U_0 \cos(\omega_0 t + \beta \sin \Omega t + \varphi_0)$$



При $\beta \ll 1$ справедливо наближене співвідношення:

$$U(t) \approx U_0 (\sin \omega t + \beta \sin \Omega t \cos \omega t).$$

В цьому випадку частотно-модульоване коливання, так само як і амплітудно-модульоване, складається з несучої частоти ω і нижньої та верхньої бічних частот $\omega + \Omega$ і $\omega - \Omega$. Тому при малих β смуги частот, займані амплітудно-модульованим і частотно-модульованим сигналами, однакові. При більших індексах β спектр бічних частот значно збільшується. Крім коливань з частотами $\omega \pm \Omega$, з'являються коливання, частоти яких дорівнюють $\omega \pm 2\Omega$, $\omega \pm 3\Omega$ і т. д., амплітуда яких зменшується при віддаленні від ω_0 . Повна ширина смуги частот, яку займає частотно-модульоване коливання з девіацією $\Delta\omega$ і модулюючою частотою Ω (з точністю, достатньою для практичних цілей), може вважатися рівною $2\Delta\omega + 2\Omega$. Ця смуга завжди ширша, ніж при амплітудній модуляції.

На практиці ширину спектра обмежують частотами складових, амплітуди яких не менші за $0,1U_0$

Оскільки при передачі сигналу майже всі паразитні перешкоди впливають на амплітуду сигналу, але не змінюють його частоту, суттєвою перевагою ЧС являється висока перешкодозахищеність.

ЧМ застосовується для високоякісної передачі звукового сигналу в радіомовленні, для звукового супроводу телевізійних програм, в музичних синтезаторах.

Перевага частотної модуляції – висока перешкодозахищеність,

Недолік – широкий частотний спектр.

Завдання

Дано:

1. Низькочастотне гармонійне коливання $U_{\text{інф}}(t) = U_{\text{інф}}\cos(\Omega t + \varphi)$.

2. Високочастотне несуче коливання $U_0(t) = U_0\cos(\omega_0 t + \varphi_0)$.

Отримати частотно-модульоване коливання та графіки модулюючого, несучого, частотно-модульованого сигналу та спектру ЧМ сигналу.

Порядок виконання роботи

1. Ознайомитися з основними принципами частотної модуляції сигналів.
2. Запустити програму Matlab.
3. Відкрити NewScript.
4. Запрограмувати значення модулюючого та несучого сигналів.
5. За допомогою Matlab отримати графіки модулюючого, несучого і частотно-модульованого сигналів.
6. За допомогою Matlab отримати графік спектру ЧМ сигналу.
7. Зробити висновки.

Запитання для самоперевірки

1. Дайте визначення частотної модуляції
2. Як визначити коефіцієнт частотної модуляції, який він повинен бути?
3. Що таке девіація частоти?
4. Який спектр ЧМ сигналу?
5. Назвіть переваги та недоліки частотної модуляції .

Зразок виконання роботи

```
% Вихідні дані для однотональної частотної модуляції
clear all
%Початкові дані
dt=0.001;
t0=0;
tkinc=0.5;
U0=4; % [Вольт]- амплітуда опорного сигналу
```



```

Uinf=2; % [Вольт]- амплітуда інформаційного сигналу
beta=1; % [безрозмірна] - коефіцієнт частотної модуляції
F1inf=5;% частота інформаційного сигналу
fi1=0; %45; % [grad] - фаза інформаційного сигналу
F2op=10^2;% частота опорного сигналу
fi0=0; %10; % [grad] - фаза опорного сигналу
%Розрахунок
t=t0:dt:tkinc; %[с] - час передачі модульованого сигналу (частота
опорного сигналу)
Omega1=F1inf*2*pi; % [Герц] - частота інформаційного сигналу
omega0=F2op*2*pi; % [Герц] - частота опорного сигналу
rad=pi/180;
St=U0*cos(omega0*t+fi0*rad); % опорний сигнал
U1=Uinf*cos(Omega1*t+fi1*rad); % інформаційний сигнал
ModSignal=U0*cos(omega0*t+beta*omega0/Omega1*sin(Omega1*t+fi1*rad));
%модульований сигнал
Fs = 1/dt; %10000; % Sampling frequency
T = 1/Fs; % Sampling period
L = Fs*tkinc+1; %100001; % Length of signal
t = (0:L-1)*T; % Time vector
Y = fft(ModSignal); % швидкеперетворенняФур"есигналу
P2 = abs(Y/L);
P1 = P2(1:(L/2)+1);
P1(2:end-1) = 2*P1(2:end-1);
f = Fs*(0:(L/2))/L;

figure(1)
plot(f,P1)
title('Single-Sided Amplitude Spectrum of X(t)')
xlabel('f (Hz)')
ylabel('|P1(f)|')
hold on % закріпленняграфіка
plot(t,U1,'r')

figure(2)
plot(t,St)
title('Опорнийсигнал')
xlabel('час')
ylabel('амплітуда')
grid % сітка
hold on % закріпленняграфіка
plot(t,U1,'r')

figure(3)
plot(t,U1)
title('Інформаційнийсигнал')
xlabel('час')

```

```
ylabel('амплітуда')  
grid
```

```
figure(4)  
plot(t,ModSignal)  
title('Модульований сигнал')  
xlabel('час')  
ylabel('амплітуда')  
grid
```

ЛАБОРАТОРНА РОБОТА №3

ЦИФРОВІ МЕТОДИ ПЕРЕДАЧІ БЕЗПЕРЕРВНИХ ПОВІДОМЛЕНЬ. ІМПУЛЬСНО-КODOVA МОДУЛЯЦІЯ

Мета роботи

Вивчення загальних принципів цифрової обробки сигналів на прикладі імпульсно-кової модуляції за допомогою програмного середовища Matlab.

Теоретичні відомості

Дискретні повідомлення вже в первісному вигляді носять символний характер, в той час як аналогові повідомлення мають вигляд безперервних функцій часу, які необхідно перетворити в дискретний (цифровий) сигнал таким чином, щоб зберегти істотну частину інформації, яка міститься в повідомленні. Представлення аналогового сигналу в цифровій формі здійснюється за допомогою імпульсно-кової модуляції ІКМ.

В основі ІКМ лежить аналогово-цифрове перетворення сигналу АЦП, яке полягає в дискретизації в часі, квантуванні за рівнем і кодуванні отриманих відліків.

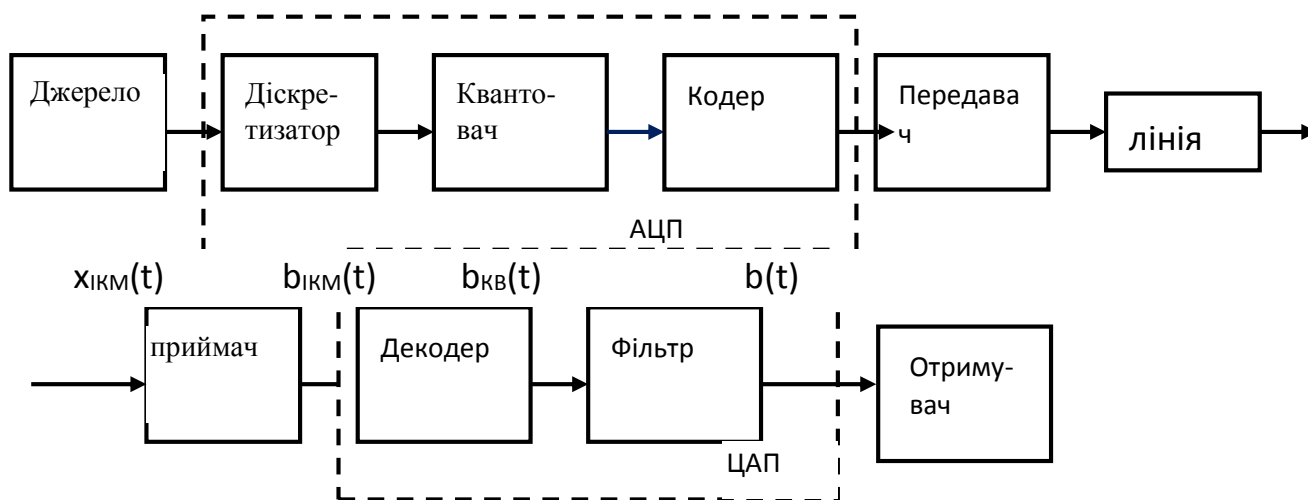


Рис.1. Структурна схема цифрової системи передачі.

Перший етап ІКМперетворення - представлення сигналу з обмеженим спектром часовими відліками, на основі теореми Котельникова, згідно з якою сигнал з обмеженим спектром частот можна повністю представити сукупністю відліків, взятих через інтервали часу з тактовою частотою, яка щонайменше вдвоє перевищує найвищу частоту спектра сигналу F

$$b(t) = \sum_{-\infty}^{\infty} b(k\Delta t) \frac{\sin 2\pi F(t - k\Delta t)}{2\pi F(t - k\Delta t)}$$

Це співвідношення показує, що для відновлення вихідного сигналу за сукупністю відліків $b(k/2F)$, взятих з частотою $2F$, їх слід пропустити на прийомі через ідеальний фільтр нижніх частот з смугою F , який має імпульсну реакцію

$$g(t) = \frac{\sin 2\pi Ft}{2\pi Ft}$$

Відомо, що умова точного представлення функції за допомогою ряду Котельникова не може бути реалізована (ідеальна дельта-послідовність, нескінченний часовий інтервал, обмеженість спектра). Це обумовлює похибки дискретизації.

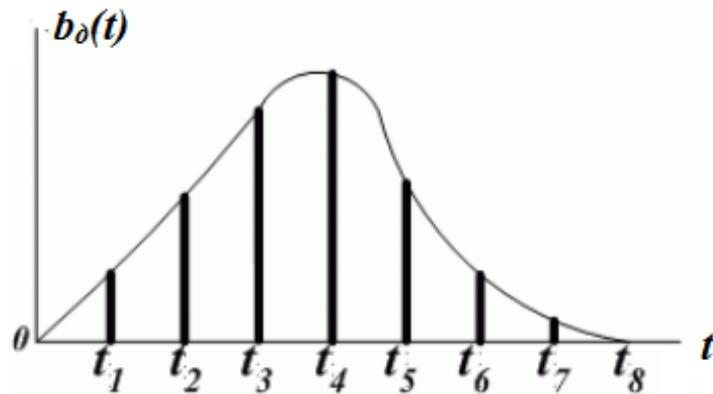


Рис.2. Дискретизація сигналу за часом

Аналоговий сигнал перед дискретизацією, щоб уникнути ефекту накладення спектрів обмежують діапазоном частот, в якому зосереджена основна енергія спектра (до F_B).

Другий етап ІКМ перетворення- квантування сигналу. При цьому безперервний інтервал значень сигналу, що передається, замінюється кінцевою множиною дозволених для передачі значень. Відстань між сусідніми дискретними рівнями називають інтервалом або **кроком квантування** Δ .

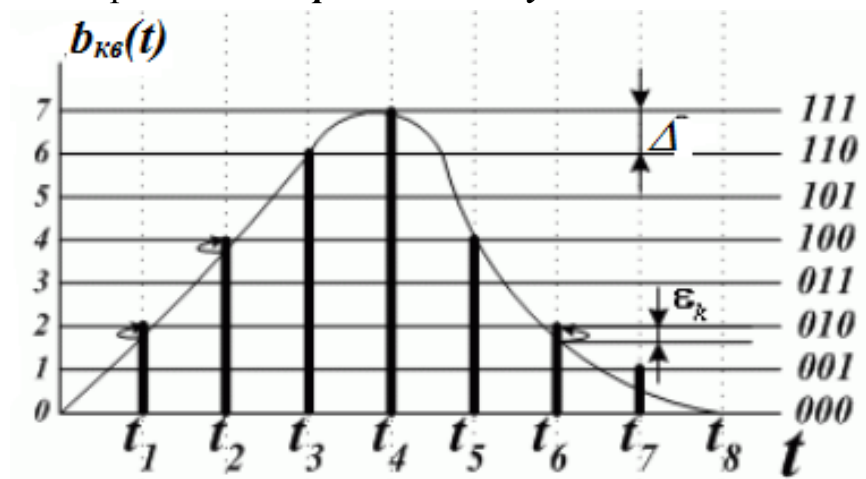


Рис.3. Квантування сигналу за рівнем

Сигнал при передачі кожного разу округлюють до найближчого дозволеного рівня квантування. При цьому неминуче виникає помилка

квантування (шум квантування). Потужність шумів квантування визначається числом рівнів квантування.

Третій етап ІКМ-перетворення - кодування квантованих значень двійковим кодом (рис.3). При ІКМ безперервне повідомлення спочатку піддається дискретизації по часу і по рівню, а потім отримана послідовність L рівнів кодується (двійковим кодом). При цьому кожному рівню присвоюється кодова комбінація з n символів "1" і "0". Число символів в комбінації має задовольняти умові $2^n \geq L$ або $n \geq \log L$.

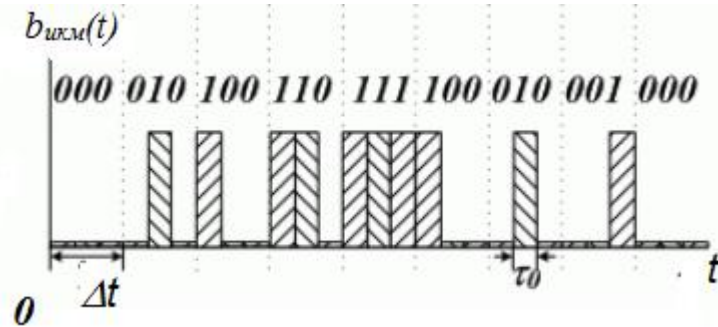


Рис.3. Кодування сигналу

На приймальній стороні сигнал після демодуляції надходить на **цифро-аналоговий перетворювач** (ЦАП), призначення якого - відновити безперервне повідомлення за прийнятою послідовністю кодових комбінацій.

У приймальній частині кодові групи декодуються і утворюють послідовність імпульсів, величина яких пропорційна переданим рівням квантування. В результаті проходження цих імпульсів через ФНЧ відновлюється передане безперервне повідомлення.

Завдання

Дано:

1. Аналоговий сигнал $U(t) = U_0 \sin(t)$

Виконати:

1. Дискретизувати аналоговий сигнал часовими відліками, на основі теореми Котельникова в програмному середовищі Matlab.
2. Квантувати сигнал по рівню.
3. Отримати в Matlab графік дискретизованого та квантованого по рівню сигналу та графік шумів квантування.

Порядок виконання роботи

1. Ознайомитися з основними принципами імпульсно-кодової модуляції сигналів.
2. Запустити програму Matlab.
3. Відкрити NewScript.
4. За допомогою Matlab отримати графік дискретизованого та квантованого по рівню сигналу та графік шумів квантування.
5. Зробити висновки.

Запитання для самоперевірки

1. Дайте визначення імпульсно-кодової модуляції.
2. Сформулюйте теорему Котельникова.
3. В скільки етапів проводяться ІКМ - перетворення?
4. Назвіть і поясніть етапи ІКМ - перетворення?
5. Що таке шуми квантування і чому вони виникають?

Зразок виконання роботи

```
dt=0.05;
t = [0:dt:2*pi]; % Times at which to sample the sine function
mp=0.5;
L=16; % = 2^4 - 4 bits
dp=2*mp/L;
sig = mp*sin(t); % Original signal, a sine wave
partition = [-mp:dp:mp]; % Length 15, to represent 16 intervals
codebook = [-(mp):dp:(mp+dp)]; % Length 16, one entry for each interval
[index,quants] = quantiz(sig,partition,codebook); % Quantize.

figure(1)
plot(t,sig,'x')
hold on
stem(t,quants,'r','.')

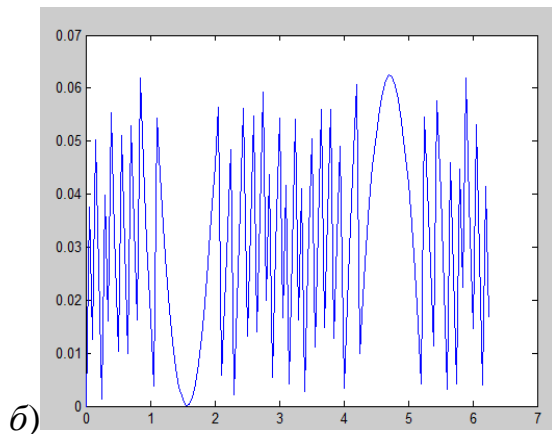
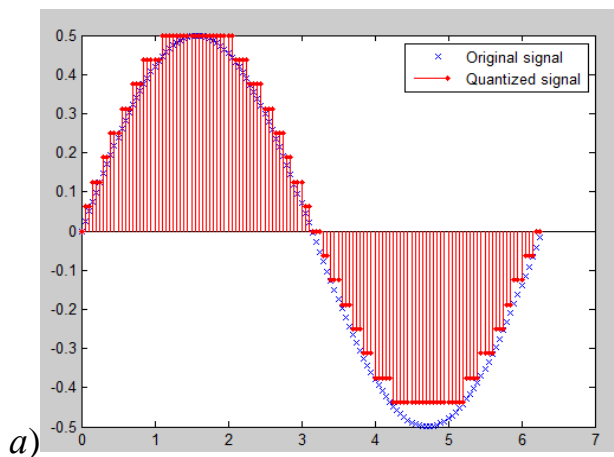
```

```

legend('Originalsignal','Quantizedsignal');
%axis([-0.2 7 -1.2 1.2])
q=quants-sig; % quantizationerror

figure(2)
plot(t,q)
Nq=dp.^2/12; % quantizationnoise
So=mp.^2/2; % signalpower
SqNr=So/Nq; %Signal-to-QuantizationNoiseRatio
mm=sig/mp;
mu=15;
y=(1/log(1+mu))*log(1+mu*sig/mp);

```



a – сигнал дискретизований за часом і квантований по рівню, *б* – шуми квантування

ЛАБОРАТОРНА РОБОТА №4

КОД ХЕМІНГА

Мета роботи: змоделювати процеси оцифрування дискретизованого сигналу, його кодування з використанням коду Хемінга, передачу по дискретному каналу зв'язку та декодування.

Теоретичні відомості

Код Хемінга належить до систематичних кодів. *Систематичними* називають такі коди, в яких інформаційні та коригувальні символи розташовані по суворо визначеній системі і завжди займають строго певні місця в кодових комбінаціях. Код Хеммінга складається з комбінації з m інформаційних і k перевірочних символів. Перевірочні символи стоять на місцях 2^i де $i = 0, 1, 2, 3, \dots$ тобто 1, 2, 4, 16, 32 ... Надлишкова частина коду будується таким чином, щоб під час декодування можна було встановити не тільки наявність помилки, але і її розташування всередині кодової комбінації. Досягається це шляхом багаторазової перевірки прийнятої кодової комбінації на парність. Сума одиниць на перевірочних позиціях повинна бути парною. Якщо сума парна - значення контрольного коефіцієнта дорівнює нулю, в іншому випадку – одиниці

При цьому число перевірок завжди дорівнює числу контрольних розрядів k . При кожній перевірці охоплюється частина інформаційних символів і один з контрольних розрядів, в ході перевірки отримують один перевірочний символ. В результаті перевірок виходить k -розрядне двійкове число, яке вказує номер позиції спотвореного символу. Якщо в результаті перевірки отримана комбінація з k нулів, значить помилок в кодовій комбінації не виявлено. Код Хеммінга виправляє одиночні помилки.

Для того, щоб кодувати сигнал кодом Хемінга, його потрібно спочатку дискретизувати за часом, потім квантувати.

Квантування за амплітудою – це процес заміни реальних (вимірних) значень амплітуди сигналу значеннями, наближеними з деякою точністю. Кожен з

можливих рівнів називається рівнем квантування, а відстань між двома найближчими рівнями квантування називається кроком квантування. У випадку лінійного розбиття амплітудної шкали на рівні, квантування називають лінійним (однорідним). Пристрій, який здійснює квантування та оцифрування квантів називається аналогово-цифровим перетворювачем (АЦП). Кількість рівнів квантування визначається розрядністю АЦП, тобто кількістю двійкових символів, якими буде представлятися один дискретний відлік. Для трьохрозрядного АЦП таких рівнів буде $2^3 = 8$, для вісьмирозрядного - $2^8 = 256$.

Робота з Matlab

Моделюючи роботу АЦП використовуйте функцію округлення до найближчого цілого *fix()*.

Для кодування/декодування інформаційної послідовності методом Хеммінга використовуються вбудовані функції системи *encode()* та *decode()*.

Завдання

1. Змоделювати сигнал повідомлення заданий аналітичною функцією $y = 5 \sin(2\pi ft)$, $f = 20 \text{Гц}$, $t \in (0, 0.2]$.
2. Провести оцифрування сигналу
3. Закодувати інформаційне повідомлення використовуючи код Хеммінга
4. Змоделювати дискретний канал зв'язку шляхом внесення в інформаційне повідомлення завади.
5. Декодувати повідомлення та визначити скільки помилок може виправити код Хеммінга.

Порядок виконання роботи

1. Ознайомитися з кодом Хеммінга
2. Запустити програму Matlab.
3. Відкрити NewScript.
4. Змоделювати сигнал та провести його оцифрування.
5. Закодувати інформаційне повідомлення використовуючи код Хеммінга

Моделюючи роботу АЦП використовуйте функцію округлення до найближчого цілого *fix()*.

Для кодування/декодування інформаційної послідовності методом Хемінга використовуються вбудовані функції системи *encode()* та *decode()*.

6 За допомогою Matlab отримати графік дискретизованого і квантованого по рівню сигналу та графік шумів квантування.

7. Зробити висновки.

Запитання для самоперевірки

1. До яких кодів належить код Хемінга?
2. Побудова коду Хеммінга.
3. Яку кількість перевірок потрібно зробити для визначення наявності, або відсутності помилки?
4. Скільки помилок може виправити код Хемінга?

Зразок виконання роботи

```
Time Start = 0; %початковий відлік часу
Time End = 0.2;% кінцевий відлік часу
Time Step = 0.2/1023; %крок побудови сигналу
Abd Bitrate = 8; % кількість біт - розрядність АЦП
numberOfQuantizationLevels = 2.^abdBitrate; % кількість рівнів квантування
(2^8)=256
timeArray = [timeStart:timeStep:timeEnd]; % масив значень часу
signalArray = 5*sin(2*pi*20*timeArray);% +7*sin(2*pi*3*timeArray); % сигнал
%plot(t, sig, 'x')
sigMaxValue = max(signalArray); % максимум сигналу
sigMinValue = min(signalArray);% мінімум сигналу
quantumStep = (sigMaxValue - sigMinValue)/(numberOfQuantizationLevels-1);
%крок квантування
quantLevels = [sigMinValue:quantumStep:sigMaxValue]; % масив значень сигналу
за квантованими рівнями
codebook = [sigMinValue:quantumStep:sigMaxValue + quantumStep]; % масив
значень сигналу за квантованими рівнями+одне значення
[index,quants] = quantiz(signalArray,quantLevels,codebook); % квантування
сигналу (вектор,лінія значень)
figure(1)
plot(timeArray, quants);
%plot(timeArray, index, '+');
%encodeusinghammingcode
binaryIndex = de2bi( index ); % перетворення десяткових значень в бінарні
```

```

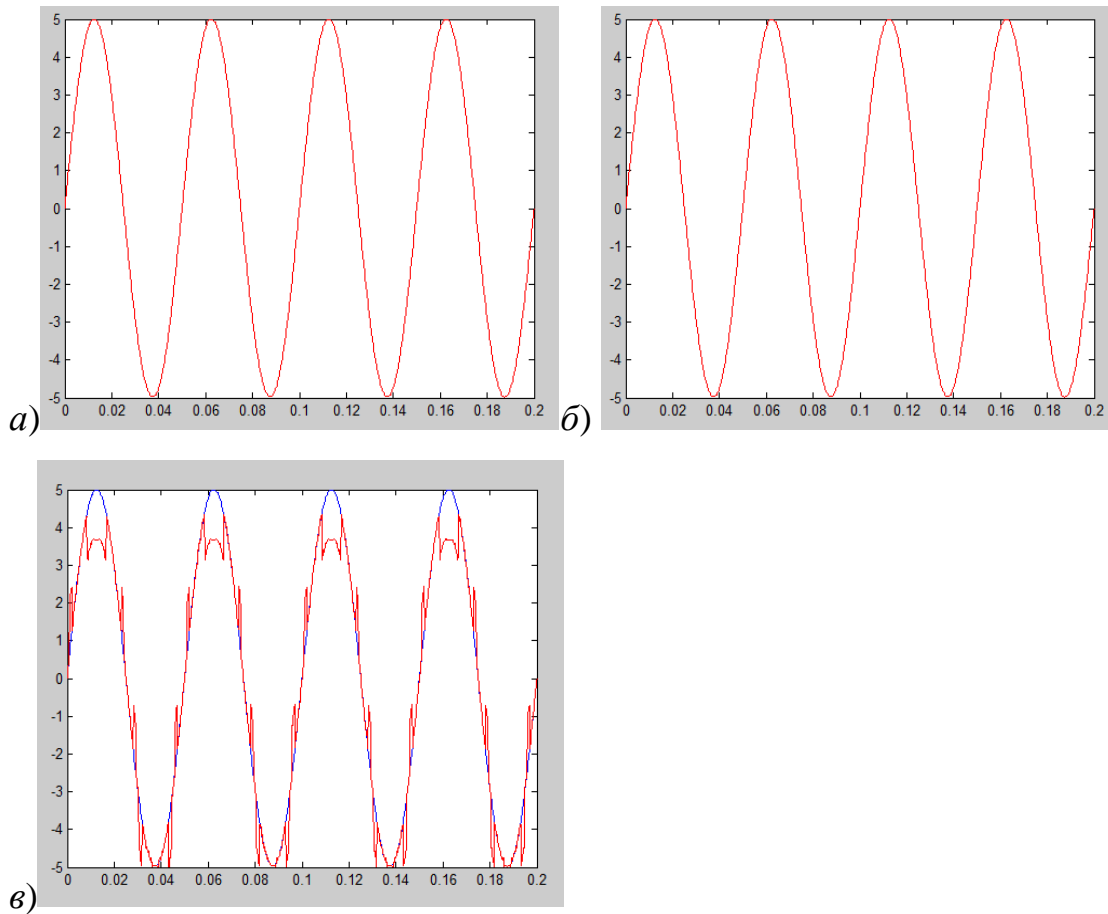
%spliteachrowofbinaryIndexmatrixinto 2 parts (partlength = 4)
binaryIndexSplitted = reshape(binaryIndex.',4,2048).'; % розбивка бінарних
елементів на матрицю значень по 4 біти
n = 7; % Codelength
k = 4; % Messagelength
%encodepart
encodedMatrix = [];
forthis_row = binaryIndexSplitted.'
encodedMatrix = [encodedMatrix,encode(this_row,n,k,'hamming/binary')];
end
encodedMatrix = encodedMatrix.';

encodedMatrixOrigin = encodedMatrix; % оригінальна матриця закодованих
повідомлень
%addnoise % encodedMatrix(:,1) = 0;
% encodedMatrix(:,2) = 0;
%encodedMatrix(:,3) = 0;
%encodedMatrix(:,4) = 0;
%encodedMatrix(:,5) = 1;
%encodedMatrix(:,6) = 1;

[emh,emw] = size(encodedMatrix);
changedValuesAfterNoise =
biterr(encodedMatrixOrigin,encodedMatrix)%кількість змінених елементів
noiseRate = changedValuesAfterNoise/(emh*emw) %ПОМИЛКИ по відношенню до
сигналу(відсоткова кількість)

%decodepart
decodedMatrix = [];
forthis_row = encodedMatrix.'
decodedMatrix = [decodedMatrix,decode(this_row,n,k,'hamming/binary')];
end
decodedMatrix = decodedMatrix.';
[dmh,dmw] = size(decodedMatrix);
errorsDecode = biterr(binaryIndexSplitted, decodedMatrix)
errorsDecodeRate = errorsDecode/(dmh*dmw)
binaryIndexOutput = reshape(decodedMatrix.',8,1024).';% об'єднують
повідомлення до 8 біт
indexOutput = bi2de(binaryIndexOutput).'+1;% бінарна в десяткову матриця
перетворюється
quantsOutput = quantLevels(indexOutput);
holdon
plot(timeArray, quantsOutput, 'r');
holdoff

```



а) Сигнал прийнятий без помилки; б) Сигнал прийнятий з однією помилкою;

в) Сигнал прийнятий з двома помилками

ЛАБОРАТОРНА РОБОТА №5

ЦИКЛІЧНИЙ КОД

(4години)

Мета роботи: змодельовати процеси оцифрування дискретизованого сигналу, його кодування з використанням циклічного коду, передачу по дискретному каналу зв'язку та декодування.

Теоретичні відомості

Циклічні коди відрізняються високою ефективністю виявлення помилок і порівняно простою реалізацією кодуючих і декодуючих пристроїв у вигляді регістрів з зсувом і зворотними зв'язками.

Циклічний код це код, у якого з приналежної йому комбінації

$$a_0, a_1, a_2 \dots a_{n-1}, \dots a_n$$

шляхом циклічної перестановки отримують також дозволена комбінацію

$$a_n, a_0, a_1 \dots a_{n-2}, a_{n-1}.$$

Дозволеними комбінаціями циклічного коду є такі комбінації, які діляться на деякий вихідний (утворюючий) багаточлен $P(x)$. Комбінація з помилкою (з діапазону помилок, які може виявити даний код) після ділення на $P(x)$ дасть залишок відмінний від нуля, що свідчить про наявність спотворення у прийнятій комбінації.

При кодуванні багаточлен $G(x)$, що відображає двійковий код повідомлення, яке підлягає передачі, помножують на x^r . Це збільшує довжину кодової комбінації на r розрядів, які потрібні для запису у них перевірочних розрядів.

Добуток $G(x) \times x^r$ ділять на утворюючий поліном $P(x)$, а залишок $R(x)$ від цього ділення складають з $G(x) \times x^r$, тобто записують залишок у r молодших розрядів (де після зсуву – перемноження на x^r записані нулі). Отриманий поліном $F(x) = G(x) \times x^r + R(x)$ є комбінацією, що ділиться без залишку на $P(x)$, тобто він є дозволеною комбінацією даного циклічного коду.

Дійсно, якщо $f(x)$ є частка від ділення $G(x) x^r$ на $P(x)$, то

$$G(x) x^r = f(x) \times P(x) \oplus R(x). \quad (43)$$

Додавши за модулем 2 до правої та лівої частини $R(x)$ отримуємо

$$G(x) x^r \oplus R(x) = f(x) \times P(x), \quad (44)$$

Тобто, комбінація $F(x) = G(x) \times x^r \oplus R(x)$ дійсно ділиться на $P(x)$ без залишку.

Робота з Matlab

Для кодування/декодування інформаційної послідовності використовуються вбудовані функції системи *encode()* та *decode*.

Завдання 1

1. Змоделювати сигнал повідомлення заданий аналітичною функцією $y = 5 \sin(2\pi ft)$, $f = 20 \text{Гц}$, $t \in (0, 0.2]$
2. Провести оцифрування сигналу
3. Закодувати інформаційне повідомлення використовуючи циклічний код
4. Змоделювати дискретний канал зв'язку шляхом внесення в інформаційне повідомлення завади.
5. Декодувати повідомлення та визначити скільки помилок може виправити циклічний код.

Порядок виконання роботи

1. Ознайомитися з циклічним кодом
2. Запустити програму Matlab.
3. Відкрити NewScript.
4. Змоделювати сигнал та провести його оцифрування.
5. Закодувати інформаційне повідомлення використовуючи циклічний код
6. Перевірити здатність коду виявити та виправити помилки певної кратності, для чого ввести помилки. Визначити, чи може код забезпечити виправлення помилок кратності 2 та 3.
7. Зробити висновки.

Питання для самоперевірки

1. До яких кодів належить циклічний код?
2. Побудова циклічного коду.
3. Як проводять кодування і декодування циклічним кодом?
4. Скільки помилок може виправити циклічний код?

Зразок виконання роботи

Файл `crc.m`

```
function [crc_bits] = crc(in_bits)
    crc_poly = [1,0,1,1];
    crc_len = 3;

    % Define local variables
    crc_rem = zeros(1,crc_len+1);
    tmp_array = [in_bits, zeros (1,crc_len)];

    % Loop to calculate CRC bits
    for(n=1:length(in_bits)+crc_len)
        for(m=1:3)
            crc_rem(m) = crc_rem(m+1);
        end
        crc_rem(crc_len+1) = tmp_array(n);

        if (crc_rem(1) ~= 0)
            for(m=1:3)
                crc_rem(m) = mod(crc_rem(m)+crc_poly(m), 2);
            end
        end
    end

    for(n=1:3)
        crc_bits(n) = crc_rem(n+1);
    end
end
```

Файл основної програми

```
time Start = 0;
time End = 0.2;
time Step = 0.2/1023;
abd Bitrate = 8;
number Of Quantization Levels = 2.^abdBitrate;

time Array =[time Start: timeStep: timeEnd];
signal Array = 5*sin(2*pi*20*time Array);
%plot(t,sig,'x')
sigMax Value = max(signal Array);
sigMin Value = min(signal Array);
quantum Step = (sig Max Value - sig Min Value)/(numberOfQuantizationLevels-1);

quant Levels = [sigMin Value: quantum Step: sig Max Value];
codebook = [sig Min Value: quantum Step: sig Max Value + quantum Step];

[index, quants] = quantiz(signal Array, quant Levels, codebook);

plot(time Array, quants);
%plot(time Array, index, '+');
```

```

%encode using hamming code

binary Index = de2bi( index );
%split each row of binary Index matrix into 2 parts (part length = 4)
binary Index Splitted = reshape(binaryIndex.',4,2048).';

n = 7;           % Code length
k = 4;           % Message length

%encode part
encoded Matrix = [];
for this_row = binary Index Splitted.'
    encoded Message = crc(this_row');
    encoded Matrix = [encoded Matrix, [this_row',encoded Message]'];
end
encoded Matrix = encoded Matrix.';

encoded MatrixOrigin = encoded Matrix;
%add noise
encoded Matrix(:,1) = 1;
encoded Matrix(:,2) = 0;
%encoded Matrix(:,3) = 1;
%encoded Matrix(:,4) = 0;
%encoded Matrix(:,5) = 1;
%encoded Matrix(:,6) = 1;

[emh,emw] = size(encode dMatrix);
changed ValuesAfter Noise = biterr(encoded Matrix Origin, encodedMatrix)
noiseRate = changed Values After Noise/(emh*emw)

%decode part
decode Matrix = [];
for this_row = encoded Matrix.'
if(~any(crc(this_row') ~= 0))
    decoded Matrix = [decoded Matrix,this_row(1:k)];
else
for(i=1:(n+1))
    temp = this_row';
if (i == (n+1))
    decoded Matrix = [decoded Matrix, temp(1:k)'];
break;
end
    temp(i)=~temp(i);
if(~any(crc(temp) ~= 0))
    decoded Matrix = [decoded Matrix, temp(1:k)'];
break;
end
end
end
end
end
decodedMatrix = decoded Matrix.';

[dmh,dmw] = size(decoded Matrix);
errors Decode = biterr(binary Index Splitted, decoded Matrix)
errors Decode Rate = errors Decode/dmh*dmw

binary Index Output = reshape(decodedMatrix.',8,1024).';
index Output = bi2de(binary Index Output).'+1;

```



```

quants Output = quant Levels(index Output);

hold on
plot(time Array, quants Output);

hold off

```

Завдання 2

Представити свої ініціали у двійковому коді, після чого:

- а) визначити відстані Хемінга між кодовими комбінаціями першої та другої, другої та третьої, першої та третьої літер;
- б) закодувати кожен із літер завадостійким циклічним кодом;
- в) отримати залишки шляхом ділення інформаційних складових на утворюючий поліном;
- г) сформувані кодові слова, якими повинні бути представлені літери ініціалів при передачі їх в каналі зв'язку;
- д) перевірити правильність сформованих кодових слів шляхом ділення їх на утворюючий поліном (залишки повинні дорівнювати нулю);
- е) перевірити здатність коду виявити помилки, для чого ввести помилку у вигляді багаточлена помилки $E(x)$,
- є) виправити помилку;
- з) визначити надлишковість та відносну швидкість коду

Приклад виконання

Іваненко Петро Семенович

Ініціали ППС можна закодувати з використанням того чи іншого двійкового коду відповідних літер. Наприклад, за номером літери у алфавіті (без урахування г та пробілу). Літера І - 10, П - 19, С - 21. У двійковому п'ятизначному коді коди літер відповідно будуть такими: І - 01010, П - 10011, С - 10101.

а) Відстані Хемінга:

$$d_{II} = \oplus \frac{\begin{array}{r} 01010 \\ 10011 \\ \hline 1+1+0+0+1=3 \end{array}}$$

$$d_{IC} = \oplus \frac{\begin{array}{r} 01010 \\ 10101 \\ \hline 1+1+1+1+1=5 \end{array}}$$

$$d_{ПС} = \oplus \frac{10011}{0+0+1+1+0=2}$$

Математична модель кодування циклічним кодом.

б) Кодування циклічним кодом з твірним поліномом $P(x) = x^3 + x + 1$.

Кодуємо літеру I

$$G_I(x) = 0 \cdot x^4 + 1 \cdot x^3 + 0 \cdot x^2 + 1 \cdot x^1 + 0 \cdot x^0 = x^3 + x$$

$$G_I(x) \cdot x^3 = x^6 + x^4$$

отримуємо залишок діленням $G_I(x)x^3$ на $P(x)$.

$$\begin{array}{r} \oplus x^6 + x^4 \! \! \! \underline{x^3 + x + 1} \\ \underline{x^6 + x^4 + x^3} \quad x^3 + 1 \\ 0 \quad 0 \quad x^3 \\ \oplus \underline{\quad \quad \quad x^3 + x + 1} \\ \quad \quad \quad x + 1 = R_I(x) \end{array}$$

Кодове слово завадостійкого циклічного коду буде мати вигляд

$$F_I(x) = G_I(x) \cdot x^3 \oplus R_I(x).$$

$F_I(x) = x^6 + x^4 + x + 1$. Враховуючи, що його розрядність n повинна дорівнювати 8 (п'ять інформаційних розрядів і три перевіірочні), то у двійковому коді це буде $\bar{F}_I = 01010011$.

Таким же чином отримують $F_{II}(x), \bar{F}_{II}, F_C(x), \bar{F}_C$.

Математична модель декодування циклічного коду.

Перевірка вірності формування $F_I(x)$ декодуванням. Ділимо $F_I(x)$ на $P(x)$

$$\begin{array}{r} x^6 + x^4 + x + 1 \! \! \! \underline{x^3 + x + 1} \\ \oplus \underline{x^6 + x^4 + x^3} \quad x^3 + 1 \\ 0 \quad 0 \quad x^3 + x + 1 \\ \oplus \underline{\quad \quad \quad x^3 + x + 1} \\ \quad \quad \quad 0 \quad 0 \quad 0 \end{array}$$

Залишок дорівнює нулю. Перевірочні розряди отримані вірно.

Аналогічно перевіряють $F_D(x), F_C(x)$

в) Визначення $d_{F_1F_D}, d_{F_1F_C}, d_{F_DF_C}$ аналогічно пункту а).

г) Перевіряємо можливість виявлення помилок. Нехай багаточлен помилки $E(x)=x^5$

Спотворене кодове слово буде

$$\tilde{F}_I(x) = F_I(x) + E(x)$$

$$\tilde{F}_I(x) = \frac{\oplus x^6 + x^4 + x + 1}{x^5} \bigg/ x^6 + x^5 + x^4 + x + 1$$

ділимо $\tilde{F}_I(x)$ на $P(x)$

$$\begin{array}{r} x^6 + x^5 + x^4 + x + 1 \div x^3 + x + 1 \\ \oplus x^6 + \quad x^4 + x^3 \quad x^3 + x^2 \\ \hline 0 + x^5 + x^3 + x + 1 \\ \oplus \quad x^5 + x^3 + x^2 \\ \hline 0 \quad 0 \quad x^2 + x + 1 \end{array}$$

- залишок не дорівнює нулю. Помилка виявлена.

д) Надлишковість коду 3/8; відносна швидкість 5/8.

Зразок виконання роботи в MATLAB

Файл cyclic_encoder_decoder.m

```
function [crc_bits] = cyclic_encoder_decoder (in_bits, crc_poly)
    crc_len = length(crc_poly) - 1;

% Define local variables
    crc_rem = zeros(1,crc_len+1);
    tmp_array = [in_bits, zeros(1,crc_len)];

% Loop to calculate CRC bits
    for (n=1:length(in_bits)+crc_len)
        for (m=1:1:crc_len)
```

```

        crc_rem(m) = crc_rem(m+1);
end
        crc_rem(crc_len+1) = tmp_array(n);

if(crc_rem(1) ~= 0)
for(m=1:crc_len+1)
        crc_rem(m) = mod(crc_rem(m)+crc_poly(m), 2);
end
end
end

for(n=1:crc_len)
        crc_bits(n) = crc_rem(n+1);
end
end

```

Файлsum_vector_mod2.m

```

function [result] = sum_vector_mod2(vector1, vector2)
if (length(vector1) == length(vector2))
        result = mod(vector1 + vector2, 2);
elseif (length(vector1) > length(vector2))
        result = mod(vector1 + [[zeros(1,length(vector1) -
length(vector2))],vector2], 2);
else
        result = mod(vector2 + [[zeros(1,length(vector2) -
length(vector1))],vector1], 2);
end
end

```

Файл з основною програмою

```

%Ініціали у двійковому представленні
I = [0,1,0,1,0];
P = [1,0,0,1,1];
S = [1,0,1,0,1];
%Відстані Хемінга
D_ip = sum(mod(I+P, 2));
D_is = sum(mod(I+S, 2));
D_ps = sum(mod(P+S, 2));
%Кодування циклічним кодом з поліномом
I_encoded = [I, cyclic_encoder_decoder(I, crc_poly)];
P_encoded = [P, cyclic_encoder_decoder(P, crc_poly)];
S_encoded = [S, cyclic_encoder_decoder(S, crc_poly)];
%Перевірка вірності формування
%Залишок від ділення
I_decoded_remainder = cyclic_encoder_decoder(I_encoded, crc_poly);
%Перевірка чи всі елементи в залишку дорівнюють 0, якщо хоча б один елемент
не 0,
%то I_decoded_has_error буде дорівнювати 1 (true/правда), тобто є помилка
I_decoded_has_error = any(I_decoded_remainder ~= 0);

P_decoded_remainder = cyclic_encoder_decoder(P_encoded, crc_poly);
P_decoded_has_error = any(P_decoded_remainder ~= 0);

S_decoded_remainder = cyclic_encoder_decoder(S_encoded, crc_poly);

```

```

S_decoded_has_error = any(S_decoded_remainder ~= 0);

%Перевірка можливість виявлення помилок
%Багаточлен помилки
E_x = [1,0,0,0,0,0]; %x^5
%E_x = [1,0,0,0,0,1]; %x^5+1
%E_x = [1,0,0,0,0,1,0,0,0,1]; %x^9+x^4+1
%Спотворене кодове слово
I_encoded_with_error = sum_vector_mod2(I_encoded,E_x);
%Визначення помилки, якщо вона є, то I_encoded_with_error_has_error буде
дорівнювати 1
I_encoded_with_error_has_error =
any(cyclic_encoder_decoder(I_encoded_with_error, crc_poly) ~= 0);
P_encoded_with_error = sum_vector_mod2(P_encoded,E_x);
P_encoded_with_error_has_error =
any(cyclic_encoder_decoder(P_encoded_with_error, crc_poly) ~= 0);

S_encoded_with_error = sum_vector_mod2(S_encoded,E_x);
S_encoded_with_error_has_error =
any(cyclic_encoder_decoder(S_encoded_with_error, crc_poly) ~= 0);

%Виправлення помилки
I_error_fix = I_encoded_with_error(1:length(I));
if(I_encoded_with_error_has_error == 1)
    for(i=1:length(I_encoded_with_error))
        temp = I_encoded_with_error;
        temp(i)=~temp(i);
        if(~any(cyclic_encoder_decoder(temp, crc_poly) ~= 0))
            I_error_fix = temp(1:length(I));
            break;
        end
    end
end
end

%Перевірка чи закодована літера на початку і після виправлення співпадають,
якщо isEqual дорівнює 1,
%то помилка виправлена вірно
isEqual = isequal(I,I_error_fix);

```

Результат виконання програми

| Workspace | |
|--|-------------------|
| Name ▲ | Value |
| <input type="checkbox"/> crc_poly | [1 0 1 1] |
| <input type="checkbox"/> D_ip | 3 |
| <input type="checkbox"/> D_is | 5 |
| <input type="checkbox"/> D_ps | 2 |
| <input type="checkbox"/> E_x | [1 0 0 0 0 0] |
| <input type="checkbox"/> i | 3 |
| <input type="checkbox"/> l | [0 1 0 1 0] |
| <input checked="" type="checkbox"/> l_decoded_has_error | 0 |
| <input type="checkbox"/> l_decoded_remainder | [0 0 0] |
| <input type="checkbox"/> l_encoded | [0 1 0 1 0 0 1 1] |
| <input type="checkbox"/> l_encoded_with_error | [0 1 1 1 0 0 1 1] |
| <input checked="" type="checkbox"/> l_encoded_with_error_has_error | 1 |
| <input type="checkbox"/> l_error_fix | [0 1 0 1 0] |
| <input checked="" type="checkbox"/> isEqual | 1 |
| <input type="checkbox"/> P | [1 0 0 1 1] |
| <input checked="" type="checkbox"/> P_decoded_has_error | 0 |
| <input type="checkbox"/> P_decoded_remainder | [0 0 0] |
| <input type="checkbox"/> P_encoded | [1 0 0 1 1 1 0 0] |
| <input type="checkbox"/> P_encoded_with_error | [1 0 1 1 1 1 0 0] |
| <input checked="" type="checkbox"/> P_encoded_with_error_has_error | 1 |
| <input type="checkbox"/> S | [1 0 1 0 1] |
| <input checked="" type="checkbox"/> S_decoded_has_error | 0 |
| <input type="checkbox"/> S_decoded_remainder | [0 0 0] |
| <input type="checkbox"/> S_encoded | [1 0 1 0 1 1 0 1] |
| <input type="checkbox"/> S_encoded_with_error | [1 0 0 0 1 1 0 1] |
| <input checked="" type="checkbox"/> S_encoded_with_error_has_error | 1 |
| <input type="checkbox"/> temp | [0 1 0 1 0 0 1 1] |

ЛІТЕРАТУРА

Основна:

1. Стеклов В. К., Беркман Л. Н. Теорія електричного зв'язку. Підручник. – К.: Техніка, 2006. – 552 с.
2. Жураковський Ю.П., Полторак В.П. Теорія інформації і кодування. - К.: Вища школа, 2001.–255 с.
3. Цымбал В. П. Теория информации и кодирования. – К.: Вища школа, 1992. – 263 с.
4. Беркман Л.Н., Отрох С.І., Тарбаєв С.І., Чумак Н.С. Загальні поняття про сигнали та канали зв'язку. Навчальний посібник підготовлено для самостійної роботи студентів вищих навчальних закладів – Київ: ДУТ ННІТІ, 2017. – 132 с.
5. Беркман Л.Н., Варфоломеева О.Г., Коршун Н.В., Макаренко А.О. Сигнали в системах телекомунікацій та методи їх обробки. Навч. посібник підготовлено для самостійної роботи студентів вищих навчальних закладів . – Київ: ДУТ ННІТІ, 2017. – 92с.
6. Беркман Л.Н., Бондарчук А.П., Гайдур Г.І., Чумак Н.С. Кодування джерел інформації та каналів зв'язку. Навч. посібник підготовлено для самостійної роботи студентів вищих навчальних закладів за кредитно-модульною організацією навчального процесу. – Київ: ННІТІ ДУТ, 2018. – 91с.
7. Варфоломеева О.Г., Отрох С.І., Твердохліб М.Г., Чумак О.І. Оптимальне приймання сигналів. Основи теорії завадостійкості. Навч. посібник підготовлено для самостійної роботи студентів вищих навчальних закладів . – Київ: ННІТІ ДУТ, 2018. – 120 с.

Додаткова:

1. Радиотехнические системы передачи информации. Учебное пособие для вузов / В.А.Борисов, В.В.Калмыков, Я.М.Ковальчук и др. / Под ред. В.В.Калмыкова - М.: Радио и связь, 1990. - 304 с.
2. Игнатов В.А. Теория информации и передача сигналов. - М.: Радио и связь, 1991. - 280 с.
3. Кузьмин И.В. Кедров В.А. Основы теории информации и кодирования. - К.: Вища школа, 1986. - 238 с.
4. Цымбал В.П. Теория информации и кодирования. - К.: Вища школа, 1990. - 263 с.
5. Колесник В.Д., Полтырев Г.Ш. Курс теории информации. - М.: Наука, 1982. - 320 с.
6. Гойхман Э.Ш., Лосев Ю.И. Передача информации в АСУ. - М.: Сов. радио, 1974. - 280 с.
7. Теория передачи сигналов. / Под ред. А.Г.Зюко. - М.: Радио и связь, 1984.

ДОДАТОК А

Приклад оформлення титульної сторінки звіту лабораторної роботи

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ЗАХИСТУ ІНФОРМАЦІЇ
КАФЕДРА ІНФОРМАЦІЙНОЇ ТА КІБЕРНЕТИЧНОЇ БЕЗПЕКИ**

Теорія інформації та кодування

З В І Т

лабораторної роботи № 1

Модуляція, методи модуляції .

Двосмугова амплітудна модуляція. Ознайомлення з MATLAB

Виконав(ла): студент(ка) _____ групи

Прізвище І.Б. _____

Дата здачі/захисту _____

Перевірив _____

ДОДАТОК Б. ОСОБЛИВОСТІ РОБОТИ В MATLAB

MatLAB (Matrix Laboratory) — високопродуктивна мова для технічних розрахунків. Вона включає в себе розрахунки, візуалізацію та програмування у зручному середовищі, де задачі та рішення виражаються у формі, близькій до математичної (рис. А.1).

MatLAB — це інтерактивна система, в якій основним елементом даних є масив. Це дозволяє вирішувати різноманітні задачі, пов'язані з технічними розрахунками, особливо з тими, де використовуються матриці та вектори, в декілька разів швидше, ніж при використанні скалярних мов програмування.

Типове використання MatLAB — математичні розрахунки, створення алгоритмів, моделювання, аналіз та візуалізація даних, наукова та інженерна графіка, розробка прикладних програм.

Програмний пакет Matlab повністю задовольняє тим потребам і являє собою універсальне забезпечення для дослідження, розрахунків та моделювання різноманітних процесів у технічній сфері, що дозволяє використовувати його також у процесі навчання студентів у галузі зв'язку.

Основний доступ до довідкової інформації забезпечує операція меню Help. Вона відкриває доступ до довідкової системи MATLAB

Головними вікнами MATLAB є :

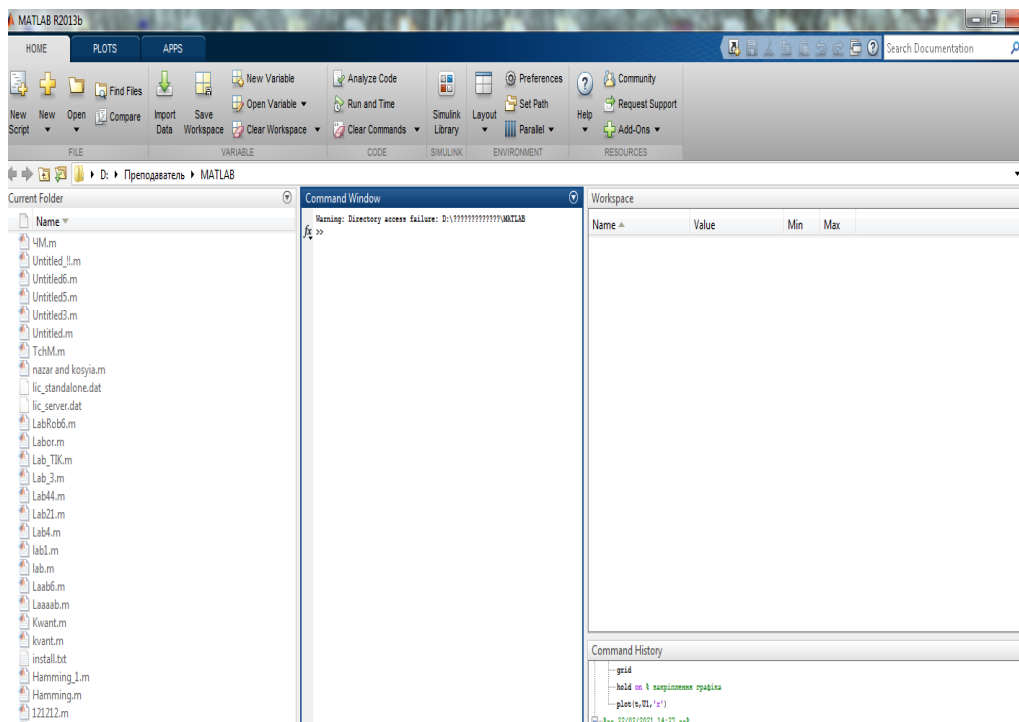


Рис 1

Command Window - командне вікно. Command Window є головним вікном MATLAB. У командному вікні з'являються символи команд, які набираються

користувачем з клавіатури, відображаються результати цих команд, тексти команд, що виконуються та інформація про помилки, які виявлено системою. Якщо програма MATLAB готова до прийняття наступної команди, то у командному вікні з'являється знак запрошення >>;

Work Space – робоча область. У вікні робочої області з'являється інформація про всі змінні, які створені та доступні у даному сеансі роботи;

Command History – історія команд. У вікні історії команд представлено всі команди MATLAB, які відображалися під час роботи у командному вікні;

Current Directory – каталог, в якому знаходиться останній відкритий файл;

У верхній частині вікна MATLAB знаходиться рядок меню, який використовується за традиційними правилами системи WINDOWS.

Основні операції MATLAB розглянемо у командному вікні.

Для ознайомлення з основними правилами використання команд MATLAB треба використати такі дії в Command Window.

Все що написано після знаку «%» - це просто коментарі до команд.

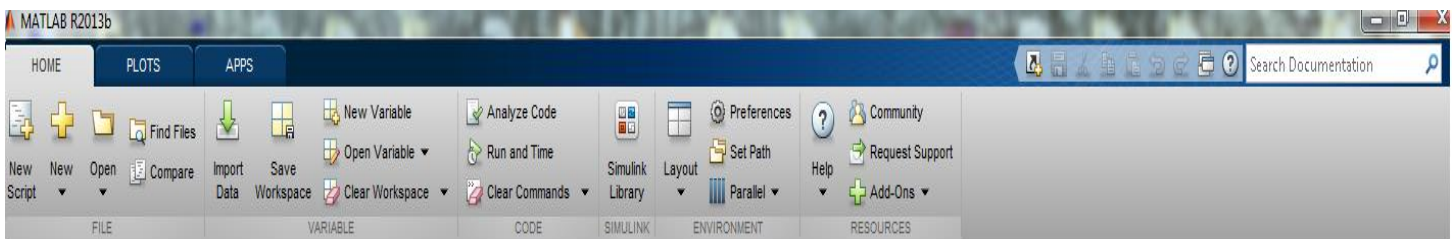


Рисунок 3 - Частина вікна системи MATLAB з меню і панеллю інструментів

Призначення усіх кнопок панелі інструментів:

New script (Новий m – файл) – виводить порожнє вікно редактора скрипт- файлів;

Openfile (Відкрити файл) – відкриває вікно для завантаження m-файлу;

Cut(Вирізувати) – вирізає виділений фрагмент і поміщає його у буфер;

Copy (Копіювати)– копіює виділений фрагмент у буфер;

Paste (Вставити) – переносить фрагмент з буфера в поточний рядок ВВЕДЕННЯ

Undo(Відмінити) – відмінює попередню операцію;

Redo (Повторити) – відновлює останню скасовану операцію; Simulink – відкриває

вікно браузеру бібліотек Simulink;

Help (Допомога) – відкриває вікно довідки.

Набір кнопок панелі інструментів забезпечує виконання найпоширеніших

команд і цілком достатній для повсякденною роботи с системою. Виклик вікна

відкриття нового файлу відкриває вікно редактора/відладчика m- файлів. Це вікно показано на рис.4.

Оформлення графіків

Для побудови графіків функції використовується команда **plot**. Однак слід зазначити, що при побудові графіків достатньо вагомим є оформлення та представлення інформації, як наприклад, *координатна сітка, підписи осей, заголовки елементів*. Такі можливості реалізуються або за допомогою додаткових параметрів або за допомогою команд та функцій. Перечислимо основні із них. Сітка наноситься командою **grid on**, функції **xlabel, ylabel** служать для розміщення підписів осей, а **title** – для заголовку.

Команда **hold on** дозволяє додавати криві на уже існуючий графік. Функція **subplot** дозволяє виводити декілька графіків в одному вікні.

Запис тексту програми (M-файлу) мовою Matlab має підпорядковуватися наступним правилам.

1. Зазвичай кожний оператор записується в окремому рядку тексту програми. Ознакою кінця оператора є символ (він не виникає у вікні) повернення каретки й переходу на наступний рядок, який вводиться в програму при натисканні клавіші Enter, тобто при переході при записі тексту програми на наступний рядок.

2. Можна розміщувати кілька операторів в одному рядку. Тоді попередній оператор у тому ж рядку має закінчуватися символом ';' або ','.

3. Можна довгий оператор записувати в декілька рядків. При цьому попередній рядок оператора має завершуватися трьома крапками ('...').

4. Якщо черговий оператор не закінчується символом ';' , результат його дії при виконанні програми буде виведений у командне вікно. Щоб запобігти виведенню на екран результатів дії оператора програми, запис цього оператора в тексті програми має закінчуватися символом '!'.

5. Рядок програми, що починається із символу '%', не виконується. Цей рядок сприймається системою Matlab як коментар. Тому для введення коментарю в будьяке місце тексту програми достатньо почати відповідний рядок із символу '%'

6. Рядки коментарю, які передують першому виконуваному (тобто такому, що не є коментарем) оператору програми, сприймаються системою Matlab як опис програми. Саме ці рядки виводяться в командне вікно, якщо в ньому набрано команду help .

7. У програмах мовою Matlab відсутній символ закінчення тексту програми.

8. У мові Matlab змінні не описуються і не оголошуються. Будь-яке нове ім'я, що зустрічається в тексті програми при її виконанні, сприймається системою Matlab як ім'я матриці. Розмір цієї матриці встановлюється при введенні значень її елементів

або визначається діями по встановленню значень її елементів, описаними у попередньому операторі або процедурі. Ця особливість робить мову Matlab дуже простою у вжитку і привабливою. У мові Matlab неможливо використання вхідної матриці або змінної, у якій попередньо не введені або обчислені значення її елементів (а значить - і визначені розміри цієї матриці). У протилежному випадку при виконанні програми Matlab з'явиться повідомлення про помилку - "Змінна не визначена".

9. Імена змінних можуть містити лише букви латинського алфавіту або цифри і мають починатися з букви. Загальна кількість символів в імені може сягати 30. В іменах змінних можуть використовуватися як великі, так і малі букви. Особливістю мови Matlab є те, що великі й малі букви в іменах розрізняються системою. Наприклад, символи "a" і "A" можуть використовуватися в одній програмі для позначення різних величин. Script-файл (файл-сценарій, або керуюча програма) та файл-функція (процедура) матимуть розширення .m при записі тексту файлу на диск, тобто їх не можна розрізнити по типу. За допомогою Script-файлів оформлюють основні програми, що керують із початку до кінця організацією усього обчислювального процесу, і окремі частини основних програм (вони можуть бути записані у виді окремих Script-файлів). Як файл-функції оформляють окремі процедури й функції (тобто такі частини програми, які розраховані на неодноразове використання Script-файлами або другими процедурами при змінених значеннях вхідних параметрів і не можуть бути виконані без попереднього завдання значень деяких змінних, які називають вхідними).

MATLAB має ефективні засоби для графічного зображення функції. Для побудови графіків функції використовується команда plot.

```
>> help plot
```

Приклад використання: послідовність команд для побудови графіка функції $y=\sin(x)$ на відрізку $[0,1]$:

```
>> x=0:0.01:1      – визначаємо значення x;
```

```
>> y=sin(x)        – визначаємо значення y=sin(x):
```

```
>> plot(x,y,'ro')  – будуємо графік функції у вказаних точках літерами «o»  
червоного кольору:
```

```
>> legend('sin(x)') назва графіка:
```

```
>> text(0.5,0.7,'y=sin(x)') в точці з координатами (0.5,0.7) на площині робимо  
надпис « y=sin(x)»
```

```
>> xlabel('x')     – надпис на осі Ox
```

```
>> ylabel('y')     – надпис на осі Oy
```

```
>> grid on        – показує на графіку координатну сітку.
```

```
>> hold on       – після цієї команди наступні графіки відображаються у тому ж  
графічному вікні.
```

Дія останніх команд відміняється командами `grid off`, `hold off`.

Наступні команди здійснюють очистку вікон

>> `clear all` – стирає всі змінні у `Workspace`;

>> `clear a` – стирає значення змінної `a`, яка після цього стає невизначеною;

>> `clear b,c,d` – стирає значення змінних `b,c,d`;

>> `clc` – очищає `Command Window`

>> `clf` – очищає графічне вікно

$U(t)$ - обвідна, UU максимальне і мінімальне значення обвідної

ТЕРМІНИ ТА ВИЗНАЧЕННЯ, ЯКІ ПОВИННІ РОЗУМІТИ ТА ЗНАТИ СТУДЕНТИ ПРИ ВИКОНАННІ ЛАБОРАТОРНИХ РОБІТ

Архітектура – характер будови мережі.

Байт (англ. byte) основна одиниця кількості інформації, що сприймається і обробляється в комп'ютері. Вона відповідає восьми розрядам двійкового коду: 1 байт = 8 біт.

Біт(англ. binary двійковий + digit знак, цифра) – одиниця вимірювання кількості інформації в двійковій системі; відповідає інформації, яку одержують при здійсненні однієї з двох рівноймовірнісних подій. Біт може мати тільки два значення "1" або "0".

Віртуальна приватна мережа (VPN) - виділена мережа, яка використовує ресурси мережі загального користування, об'єднує абонентські лінії, що підключені до різних комутаторів, забезпечує приватний план нумерації і надає послуги мережі загального користування.

Взаємодія мереж зв'язку (communication networks interworking) - спільне функціонування технологічно сполучених інформаційних мереж з метою виконання загальних завдань.

Віртуальний канал (virtual connection) - сукупність устаткування системи передачі синхронної цифрової ієрархії, що забезпечує передачу віртуальних контейнерів.

Дані - інформація у формі, придатній для автоматизованої обробки її засобами обчислювальної техніки.

Дисперсія (лат. dispersus розсіювання) – міра розсіювання випадкової величини навколо її середнього значення.

Додаток– спеціалізоване програмне забезпечення, орієнтоване на виконання будь-якої послуги.

Джерела повідомлень – людина, тварина, птах, різні предмети і явища об'єктивної дійсності.

Ентропія (грецьк. en в, всередині + trope зміна, перетворення) – в теорії інформації – міра невизначеності ситуації; в математиці – міра невизначеності випадкової функції.

Завади, перешкоди — у системах автоматичного керування — сигнали або дії, що спотворюють корисний сигнал, який несе основну інформацію у пристроях вимірювання, телевимірювання, зв'язку. Вплив завади може призвести до значних помилок систем вимірювання. За своєю природою завади можуть бути детермінованими та випадковими. **Детерміновані завади**. Приклад детермінованої завади — фон від джерела живлення змінного струму. За допомогою спеціальних конструктивних заходів, вплив детермінованих завад можна усунути. Вплив детермінованих завад на результати вимірювання, вираховують як систематичну похибку. **Випадкові завади** - джерелами випадкових завади є теплові шуми, похибки, які виникають під час перетворення сигналів. Випадкові завади описуються деякою випадковою функцією. Дуже поширене представлення випадкової завади як «білого шуму». Спеціальні схеми вимірювання враховують вплив

адитивної та мультиплікативної та інші завади. Будь-які зміни коефіцієнта передачі і часу поширення сигналів спричинюють флуктуації сигналів на виході каналів і можуть розглядатися як завади. Зміни коефіцієнта передачі_μ проявляються у вигляді флуктуацій амплітуди сигналу і називається **мультиплікативною завадою**.

Завадостійкість — здатність технічного засобу працювати за призначенням без погіршення робочих характеристик при дії наявних електромагнітних завад.

З'єднання(connection) - сукупність каналів, комутаційних і інших функціональних пристроїв, тимчасово зв'язаних між собою для забезпечення передачі сигналів між двома або більше пунктами мережіелектрозв'язку.

Ієрархія— у широкому розумінні розташування частин або елементів цілого в певному порядку від вищого до нижчого.

Інтернет - всесвітня інформаційна система загального користування, яка логічно зв'язана глобальним адресним простором та базується на Інтернет-протоколі, визначеному міжнародними стандартами.

Інтелектуальна мережа – мережа, що здатна надавати і формувати будь-які послуги на базі різноманітних транспортних мереж.

Інтерфейс(англ. interface з'єднання) – точка з'єднання двох елементів або загальна межа між двома взаємопов'язаними системами (чи рівнями однієї системи), яка забезпечує сумісність пристроїв незалежно від їх типу та виробника.

Інформація – це відомості про предмети, події, явища об'єктивної дійсності.

Інформаційні повідомлення, або просто повідомлення – це символи, знаки, процеси, що відображають у тій чи іншій формі інформацію, яка не є відомою одержувачу.

Інфокомунікаційна послуга (послуга інформаційного суспільства) - послуга зв'язку, що передбачає автоматизовану обробку, зберігання або надання по запиту інформації з використанням засобів обчислювальної техніки, як на вхідному, так і на вихідному кінці з'єднання.

Канал(canal) — сукупність технічних засобів і середовище передачі, за допомогою яких інформація передається від джерела (передавача) до отримувача (приймача) або навпаки.

Квазі - ніби, майже, немовби) у складних словах означає “ніби”, несправжній, фальшивий.

Код – це алгоритм (правило), згідно з яким кожному повідомленню присвоюється відповідна кодова комбінація. **Алфавіт коду** – це весь можливий набір цифр (символів), який використовуються для формування кодових комбінацій. **Блокові коди** –це коди, в яких довжина кодових комбінацій є однаковою. **Лінійні коди** –це коди, в яких, знаючи дві дозволені кодові комбінації, можна отримати за допомогою лінійної операції дозволена кодова комбінація. **Систематичні коди** – це коди, в яких позиції розміщення інформаційних і перевірних розрядів є незмінними.

Кодова комбінація – це набір цифр (символів), яким ставиться у відповідність певне повідомлення (буква, число, відлік неперевного сигналу), тобто, це кінцева послідовність кодових знаків.

Кінцеве устаткування користувача(customer premises equipment) - устаткування формування сигналів електрозв'язку, щопідключає до кінцевого вузла транспортної мережі й призначене для передачі або прийому заданої

Комутатор - пристрій, призначений для взаємопідключення вхідних і вихідних точок, зовнішніх ліній, каналів і т. ін.

Комутація(communication) -установлення на вимогу окремого з'єднання між потрібними входом і виходом з безлічі входів і виходів на час, необхідний для передачі сигналів.

Користувач(client) – суб'єкт, якому надаються мережні послуги інфокомунікації.

Лінія сполучна(junction transmission) -лінія передачі, що з'єднує між собою мережну станцію й мережний вузол або дві мережні станції між собою. Примітка. Сполучної лінії привласнюють назви залежно від первинної мережі, до якої вона належить — магістральна, внутрізонова, місцева.

Маршрутизатор - вузол мережі, що спрямовує пакети на основі спеціального критерію, який забезпечує оптимальне використання мереж між вихідним пунктом та пунктом призначення пакетів.

Модуляція – процес вкладення інформаційного сигналу у сигнал-переносник. Якщо немає уточнень, то під інформаційним сигналом ми розуміємо, що це неперервний (аналоговий) сигнал. **Амплітудна модуляція (АМ)** — вид модуляції, при якій параметром сигналу-носія, що змінюється, є амплітуда його коливань. **Частотна модуляція (ЧМ, ФМ, англ. *Frequency modulation*)** — тип аналогової модуляції, за якого частота вихідного сигналу змінюється у часі залежно від миттєвого значення інформаційного сигналу, тобто інформаційний сигнал керує частотою опорного сигналу. У порівнянні з амплітудною модуляцією, амплітуда сигналу залишається постійною. **Фазова модуляція** — один з видів модуляції, при якій фаза несучого коливання керується інформаційним сигналом. Фазова модуляція, що не пов'язана з початковою фазою сигналу-носія, називається відносною фазовою модуляцією (ВФМ).

Мультиплексор - електронний пристрій, призначений для об'єднання цифрових сигналів у єдиний цифровий потік

Мультиплексування – процес об'єднання

Оптимальний – найкращий з можливих варіантів чогось, найбільш відповідних даному завданню, умовам. Умова **оптимального приймання**: якщо всі можливі сигнали рівноймовірні і мають однакову енергію, оптимальний приймач відтворює повідомлення, що відповідає тому переданому сигналу,

Порт(port) (вводу-виводу) – точка підключення, один або більше адресів пам'яті, що використовуються для приймання і передавання інформації.

Провайдери Інтернет – організації, які надають користувачам послуги з доступу в Інтернет.

Послуга - структурний набір можливостей, які реалізуються на замовлення користувача.

- Постачальник інформації (Content Provider – CP)** - індивідуальний підприємець або юридична особа, що надає інформацію постачальникові послуги для її розповсюдження або надання користувачам мережею оператора зв'язку.
- Прикладна програма** - структурований набір програмних функцій, який дає користувачу можливість доступу до мультимедійних служб.
- Протокол** – набір правил і форматів (стандартів), які регламентують процедуру взаємодії однойменних рівнів двох різних рівноправних об'єктів на основі обміну інформацією і дозволяють спростити складний процес між об'єктами різних моделей і типів.
- Протокол доступу (Access protocol)** - набір процедур, прийнятий для інтерфейсу в заданій опорній точці між користувачем і мережею для забезпечення здатності користувача користуватися послугами й/або техніко-експлуатаційними можливостями мережі.
- Сервер** (англ. server подавати) – пристрій, за допомогою якого абонент може отримати послуги будь-якої мережі (елемент мережі, який виконує приймання запитів з метою їх обслуговування та передачу відповідей на отримані запити).
- Сервіс** (англ. service служити, обслуговувати) – надання послуг.
- Сигналізація** – процес повідомлення про всі керовані процеси, які відбуваються в мережі (нормальне функціонування, помилки або переривання).
- Синхронізація** – приведення двох або кількох періодично змінних явищ до точної взаємної відповідності періодів їх перебігу.
- Сигнал (signal)** - фізичне явище, одна або кілька характеристик якого можуть змінюватися з метою подання інформації. **Цифровий сигнал** – послідовність двійкових імпульсів, які представляють певне число (послідовність цифр).
- Сигналом-переносником** може бути: гармонічний сигнал, послідовність, прямокутних імпульсів, послідовність двійкових імпульсів та, напевно, інші сигнали. Для передавання сигналів через довкілля (ефір) використовують гармонійний сигнал. **Інформаційний сигнал** – сигнал на виході джерела інформації. Аналіз спектральних характеристик сигналів, які отримуємо від джерел інформації і які є матеріальними носіями інформації, показує, що їхні спектральні складові зосереджені у низькочастотному діапазоні. Труднощі ефективного випромінювання низькочастотних сигналів та значне поглинання їхньої енергії у довкіллі у разі поширення на значні відстані не дають змоги безпосередньо їх передавати одержувачу. Вирішення цієї проблеми забезпечується їх переміщенням (перенесенням) спектра низькочастотних сигналів у діапазон високих частот, для яких умови випромінювання та поширення є сприятливими. Для цього використовують високочастотний гармонічний сигнал, який виконує функцію переносника низькочастотного сигналу. Тому такий сигнал називають **переносником або несучим сигналом**.
- Стек протоколів** - багаторівневий набір протоколів, спільне використання яких забезпечує множину функцій мережі.

Стиснення - схема представлення даних для зменшення їх обсягу при збереженні прийнятної якості.

Телекомунікаційна мережа - комплекс технічних засобів, телекомунікацій та споруд, призначених для маршрутизації, комутації, передавання та/або приймання знаків, сигналів письмового тексту, зображень та звуків або повідомлень будь-якого роду, по радіо, провідних, оптичних та інших електромагнітних системах між кінцевим обладнанням.

Транспортна мережа електрозв'язку(transport network) – частина мережі електрозв'язку, що забезпечує доставку інформаційних і службових сигналів по заданих адресах і складається з ряду підмереж з можливо різними принципами їхньої організації й приналежності до різних операторів.

Транзакція– набір операцій, які виконуються одночасно як одна операція.

Трафік– навантаження (суцільність потоку даних) лінії зв'язку чи каналу.

Управління безпекою - захист мережі від несанкціонованого доступу.

Управління трафіком - ряд дій, що вживаються в усіх відповідних елементах мережі для запобігання стану перевантаження.

Чарунка (рос. ячейка) – мала робоча зона; частина інформації.

Якість обслуговування - сукупний ефект надання послуги, який визначає ступінь задоволення користувача послугою.