

Міністерство освіти і науки України
Національний університет «Запорізька політехніка»
Кафедра радіотехніки та телекомунікацій

МЕТОДИЧНІ ВКАЗІВКИ

до лабораторних робіт з дисципліни
«Широкопasmові технології телекомунікацій»
для студентів спеціальності
172 «Електронні комунікації та радіотехніка»
всіх форм навчання

Частина III

2024

Методичні вказівки до лабораторних робіт з дисципліни «Широкосмугові технології телекомунікацій» для студентів спеціальності 172 «Електронні комунікації та радіотехніка» всіх форм навчання. Частина III / Укл.: В.С. Кабак, Г.В. Мороз, Г.Ф. Вишник. – Запоріжжя: НУ «Запорізька політехніка», 2024. – 67 с.

Укладачі: В.С.Кабак, доцент, к.т.н.,
Г.В. Мороз, ст. викладач,
Г.Ф. Вишник, зав. лаб.

Рецензент: С.В. Морщавка, доцент, к.т.н.

Відповідальний.
за випуск: Г.Ф. Вишник, зав. лаб.

Затверджено:
на засіданні кафедри
радіотехніки та телекомунікацій
Протокол № 5 від 05.04.2024 р.

Рекомендовано до видання НМК
факультету інформаційної безпеки та
електронних комунікацій
Протокол № 6 від 24.04.2024 р.

ЗМІСТ

1	Лабораторна робота №5 Моделювання каналу зв'язку з використанням бінарної фазової маніпуляції (BPSK).....	4
1.1	Мета роботи.....	4
1.2	Завдання на підготовку до виконання лабораторної роботи	4
1.2.1	Еквівалентність когерентного модему і модему в основній смузі частот	8
1.2.2	Структурна схема модему BPSK.....	13
1.3	Реалізація програмного алгоритму для AWGN каналу.....	19
1.3.1	Опис основної програми	19
1.4	Експериментальна частина	35
1.5	Зміст звіту.....	37
1.6	Контрольні запитання.....	38
2	Лабораторна робота №6 Дослідження каналу зв'язку з частотно-неселективними завмираннями.....	39
2.1	Мета роботи.....	39
2.2	Завдання на підготовку до виконання лабораторної роботи ...	39
2.2.1	Канал з релієвськими завмираннями	39
2.2.2	Модель каналу зв'язку з використанням BPSK за наявності частотно-неселективних або повільних завмирань	49
2.3	Експериментальна частина	53
2.4	Зміст звіту.....	54
2.5	Контрольні запитання.....	54
	Перелік джерел посилань.....	56
	Додаток А.....	57
	Додаток Б.....	62

1 ЛАБОРАТОРНА РОБОТА № 5

МОДЕЛЮВАННЯ КАНАЛУ ЗВ'ЯЗКУ З ВИКОРИСТАННЯМ БІНАРНОЇ ФАЗОВОЇ МАНІПУЛЯЦІЇ (BPSK)

1.1 Мета роботи

1.1.1 Ознайомитися з принципами побудови модуляторів і демодуляторів для сигналів з двійковою фазовою маніпуляцією ФМ-2 (BPSK) з використанням методів цифрової обробки сигналів.

1.1.2 За допомогою ЕОМ експериментально відпрацювати методику формування модулюючого сигналу, каналних і радіочастотних символів BPSK модуляторів, розглянути особливості побудови демодуляторів.

1.1.3 Визначити залежність коефіцієнта бітової помилки (BER) від енергетичних характеристик каналу.

1.2 Завдання на підготовку до виконання лабораторної роботи

Процес зміни поточної фази несучого коливання за законом модуляції відомий як фазова модуляція. У загальному вигляді фазо-модульоване коливання можна представити у вигляді [1]:

$$s [t, b(t)] = C \cdot \cos \{ 2\pi f_0 t + \varphi [b(t)] \} = \operatorname{Re} \{ C \cdot \exp \{ j\varphi [b(t)] \} \cdot \exp [j2\pi f_0 t] \}, \quad (1.1)$$

де C – амплітуда сигналу, f_0 – несуча частота, φ – значення фази модульованого сигналу, $b(t)$ – модулююча послідовність.

З аналізу (1.1) випливає, що інформація, яка міститься у модулюючому сигналі $b(t)$, закодована у комплексній обвідній сигналу, що передається:

$$C^* = C \cdot \exp \{ j\varphi [b(t)] \}. \quad (1.2)$$

Двійкова (бінарна) фазова маніпуляція (ФМ-2) або англійською BPSK – це найпростіша форма цифрової фазової модуляції. В залежності від значення модулюючого сигналу відхилення фази модульованого сигналу від фази немодульованого сигналу складає або 0° , або 180° , тобто, має два дискретних значення. Якщо для фазомодульованого сигналу прийняти загальний опис згідно (1.1), то для ФМ-2 повинні виконуватися рівності:

$$\varphi[b(t)] \equiv 0 \text{ при } b(t) \equiv 1, \quad \varphi[b(t)] \equiv \pi \text{ при } b(t) \equiv -1; \quad 0 \leq t \leq T.$$

В результаті модульований сигнал формується з двох радіосимволів $s_1(t)$, $s_2(t)$, що визначаються виразами:

$$s_1(t) = C \cos(\omega_0 t),$$

$$s_2(t) = -C \cos(\omega_0 t).$$

Комплексна обвідна цього сигналу не змінюється на інтервалі символу i , відповідно до (1.2), може приймати наступні значення:

$$C^*(t) = C \text{ при } b(t) \equiv 1, \quad C^*(t) = -C \text{ при } b(t) \equiv -1, \quad 0 \leq t \leq T.$$

Як відомо, модульований сигнал можна отримати шляхом перемноження модулюючої послідовності $b(t)$ на сигнал несучої частоти [1]:

$$s(t) = b(t) \cdot C \cos(\omega_0 t) \tag{1.3}$$

де $b(t)$ модулюючий сигнал у форматі NRZ без постійної складової.

Оскільки модулюючий сигнал $b(t)$ представляє собою синхронний випадковий двійковий сигнал з рівнями $+1$ і -1 в основній смузі частот з бітовою швидкістю $f_b = 1/T_b$, то модульований сигнал складається з протилежних (на 180°) фазоманіпульованих сигнальних радіосимволів $s_1(t)$ і $s_2(t)$. Відповідно інформація міститься у фазі модульованого сигналу:

$$s(t) = b(t) \cdot C \cos\{\omega_0 t + \varphi[b(t)]\}, \quad (1.4)$$

де $\varphi[b(t)] = 0^\circ$ або 180° .

Основною особливістю сформованого радіосигналу є стрибки фази на 180° під час зміни полярності модулюючого сигналу, тобто розривність фази. Саме ця розривність поточної фази призводить до того, що спектральна щільність потужності для ФМ-2 значно відрізняється від нуля у неприпустимій смузі частот.

Для зменшення смуги частот у системах із застосуванням ФМ-2 використовують додаткову фільтрацію.

На рис.1.1 представлена спрощена структурна схема реалізації каналу зв'язку з використанням BPSK.

На структурній схемі з метою спрощення аналізу високочастотні тракти передавача і приймача, а також властивості каналу зв'язку відображені еквівалентним смуговим фільтром, який імітує характеристики послідовного включення передавального смугового фільтра, каналного фільтра та приймального смугового фільтра [2].

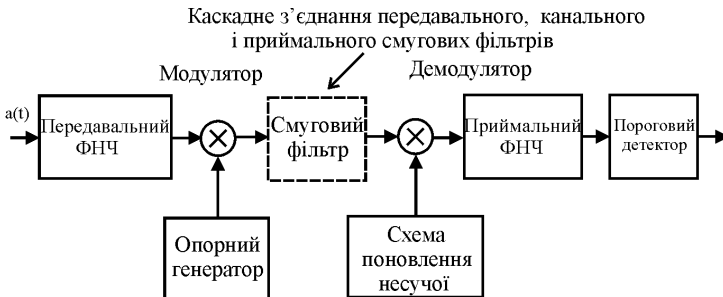


Рисунок 1.1 – Еквівалентна модель каналу зв'язку з використанням методу BPSK

Роботу когерентного демодулятора сигналів BPSK більш детально розглянемо за допомогою функціональної схеми, яку зображено на рис. 1.2.

Когерентна демодуляція передбачає формування на приймальному боці несучого коливання синфазного з несучим коливанням передавача (опорний сигнал $K_{on}(t)$ на рис.1.2). Для цього у

приймачі використовуються схеми поновлення несучого коливання з прийнятого сигналу.

Сам процес демодуляції реалізується шляхом перемноження прийнятого модульованого сигналу радіочастоти $r(t)$ (з обмеженою смугою) з поновленою немодульованою несучою $K_{\text{он}}(t) = K \cos \omega_0 t$.

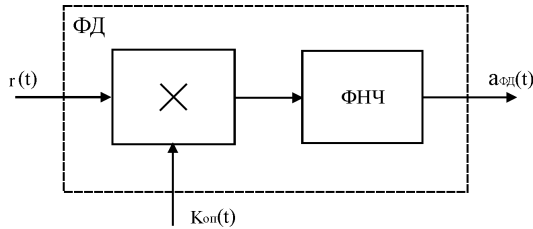


Рисунок 1.2 – Функціональна схема демодулятора для методу BPSK

Математично сигнал на виході перемножувача (як правило, цю роль виконує фазовий детектор) до НЧ фільтрації визначається виразом [2]:

$$\begin{aligned} s(t) &= r(t) \cdot K_{\text{он}}(t) = r(t) \cdot K \cos(\omega_0 t) = \\ &= C_r(t) \cos[\omega_0 t + \varphi(t)] \cdot K \cos(\omega_0 t) = \frac{1}{2} C_r K \{ \cos[\varphi(t)] + \cos[2\omega_0 t + \varphi(t)] \} \end{aligned} \quad (1.5)$$

де $r(t)$ сигнал на вході демодулятора, C_r – амплітуда прийнятого сигналу, K – амплітуда поновленого несучого коливання.

Фільтр нижніх частот (ФНЧ) на функціональній схемі демодулятора необхідний для придушення гармонічної складової з частотою $2\omega_0 t$. В результаті на виході ФНЧ формується корисний сигнал, який містить інформацію про значення фази $\varphi(t)$:

$$a_{\text{фд}}(t) = \frac{1}{2} C_r K \cos[\varphi(t)] \quad (1.6)$$

У цьому виразі $C_r K / 2$ – постійний коефіцієнт, $\cos[\varphi(t)]$ – змінний у часі модулюючий сигнал з обмеженою смугою, при $\varphi(t) = 0^\circ$ або 180° він рівняється $+1$ або -1 відповідно. Напряга демодульованого сигналу

пропорційна косинусу різниці фазових кутів між прийнятою модульованою несучою і поновленою з точністю до фази несучою – тобто, для системи, яка складається з перемножувача і ФНЧ, справедливий термін когерентний фазовий детектор або фазовий демодулятор.

На рис. 1.1 в схемі модулятора застосовується передмодуляційна низькочастотна фільтрація. Доведемо доцільність саме такої низькочастотної передмодуляційної фільтрації.

1.2.1 Еквівалентність когерентного модему і модему в основній смузі частот

В принципі, можлива реалізація лінійного модулятора за двома різними алгоритмами фільтрації, а саме, постмодуляційною фільтрацією і передмодуляційною фільтрацією – (рис. 1.3). Для першого варіанту (рис.1.3а) на виході модулятора застосовано радіочастотний смуговий фільтр (СФ).

Але, для великого класу систем модуляції і когерентної демодуляції складні смугові фільтри радіо – і проміжної частоти з вузькою смугою пропускання можуть бути замінені на відносно прості ФНЧ в основній смузі частот (рис.1.3 б) [2].

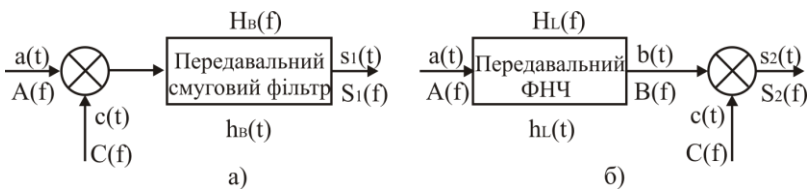


Рисунок 1.3 – Можливі реалізації модулятора BPSK з постфільтрацією (а) і передмодуляційною фільтрацією (б)

Дійсно, для основної смуги частот можна довести еквівалентність лінійно модульованих сигналів радіо і проміжної частот зі смуговою фільтрацією сигналам з низькочастотною передмодуляційною та післямодуляційною фільтрацією в основній смузі частот [2].

Лінійно модульований сигнал – це сигнал з лінійним перетворенням спектра або сигнал, який можна розглядати як результат амплітудної модуляції з двома бічними смугами і придушеною несучою частотою (англійською Double-sideband Suppressed-Carrier Amplitude Modulation, аббревіатура DSB-SC-AM).

Дійсно, в результаті такої операції спектр модулюючого сигналу переноситься таким чином, що він розташовується навколо несучої частоти. Розуміння цього факту дозволить застосовувати більш простий ФНЧ замість більш складного смугового фільтру.

Якщо лінійний модулятор містить тільки передмодуляційний фільтр НЧ, то у відповідності до структурної схеми модулятора (рис.1.36) модульований сигнал можна подати у вигляді [2]:

$$s(t) = [a(t) * h_L(t)] \cdot c(t), \quad (1.7)$$

де $a(t)$ – модулюючий сигнал, $h_L(t)$ – імпульсна характеристика ФНЧ, $*$ – означає операцію згортки, яка визначається як:

$$a(t) * h_L(t) = \int_{-\infty}^{\infty} a(\tau) \cdot h(t - \tau) d\tau.$$

Якщо застосувати перетворення Фур'є до (1.7), то з урахуванням того факту, що згортка функцій у часовій області відповідає перемноженню функцій в частотній області, а згортка функцій в частотній області відповідає перемноженню функцій в часовій області можна отримати:

$$S(f) = [A(f) \cdot H_L(f)] * C(f) \quad (1.8)$$

Аналіз виразу (1.8) свідчить, що згортка низькочастотного спектра сигналу (вираз у квадратних дужках) з сигналом синусоїдної несучої надає спектр з двома бічними смугами розташованими навколо несучої. Якщо модулюючий сигнал не містить постійної складової, то результуючий сигнал є DSB-SC-AM сигналом. Це ствердження є справедливим як для аналогових, так і цифрових

модуючих сигналів. Відфільтрований сигнал визначиться таким виразом:

$$S_2(f) = [A(f - f_0) \cdot H_L(f - f_0) + A(f + f_0) \cdot H_L(f + f_0)]. \quad (1.9)$$

Якщо ж лінійний модулятор містить тільки післямодуляційний смуговий фільтр (рис.1.3а), то

$$s(t) = [a(t) \cdot c(t)] * h_B(t), \quad (1.10)$$

де $h_B(t)$ – імпульсна характеристика смугового фільтра, а відповідне перетворення Фур'є надає наступне:

$$S_1(f) = [A(f) * C(f)] \cdot H_B(f) = [A(f - f_0) + A(f + f_0)] \cdot H_B(f). \quad (1.11)$$

Амплітудні спектри сигналів з передмодуляційною і постмодуляційною фільтрацією за виразами (1.9), (1.11) еквівалентні, якщо:

$$S_1(f) = S_2(f),$$

або

$$[H_L(f - f_0) + H_L(f + f_0)] = H_B(f). \quad (1.12)$$

З (1.12) виходить, що умова еквівалентності задовольняється, якщо $H_B(f)$ – передавальна функція смугового фільтра така ж, як передавальна функція $H_L(f)$ ФНЧ, але зміщена по частоті таким чином, щоб її центр розташовувався на несучій частоті. Для виконання умови еквівалентності СФ повинен мати передавальну функцію симетричну відносно несучої частоти. Висновки щодо еквівалентності смугового і низькочастотного каналів когерентного приймача практично тотожні [2].

Але, необхідно відмітити, що зменшення смуги частот шляхом низькочастотної фільтрації може супроводжуватися з'явленням

додаткової міжсимвольної інтерференції і, відповідно до зростання ймовірності бітової помилки.

Як відомо мінімальне значення смуги пропускання для виконання умови відсутності міжсимвольних спотворень надає теорема Найквіста: якщо короткі синхронні імпульси передаються з частотою повторення f_s , символів в секунду через канал, який має ідеальну прямокутну частотну характеристику з частотою зрізу $f_N=f_s/2$, то відклики на ці імпульси можна спостерігати незалежно, тобто без між символних спотворень.

Для синтезу каналу з прямокутною частотною характеристикою і нескінченним затуханням за межами смуги пропускання необхідно у фільтрі застосовувати нескінченну кількість ланок. До того ж спад бічних пелюсток імпульсної характеристики для такого фільтра стає дуже повільним, що призводило б до неприпустимого зростання міжсимвольних спотворень при будь-яких неточностях у фільтрації та символній синхронізації (значення ймовірності помилкового прийому одного біта P_b наближалось би до 0,5).

Тому, на практиці застосовують фільтри, які задовольняють теоремі Найквіста про часткову симетрію. Частотна характеристика таких фільтрів симетрична відносно частоти Найквіста, що сумісно з лінійною фазочастотною характеристикою забезпечує передачу без міжсимвольних спотворень.

Однією з функцій, що задовольняє теоремі Найквіста про часткову симетрію і найчастіше використовується при апроксимації частотної характеристики каналу є функція піднесеного косинусу. Амплітудно - частотна характеристика такого каналу визначається виразом [2]:

$$|H(j\omega)| = \begin{cases} 1 & 0 \leq \omega \leq \frac{\pi(1-\alpha)}{T_s} \\ \cos^2 \left\{ \frac{T_s}{4\alpha} \left[\omega - \frac{\pi(1-\alpha)}{T_s} \right] \right\}, & \frac{\pi(1-\alpha)}{T_s} \leq \omega \leq \frac{\pi(1+\alpha)}{T_s} \\ 0 & \omega > \frac{\pi(1+\alpha)}{T_s} \end{cases} \quad (1.13)$$

де T_s – тривалість символу, α – канальний коефіцієнт закруглення. При $\alpha=0$ формується фільтр з прямокутною ідеальною частотною характеристикою з мінімальною смугою пропускання $f_N=1/(2T_s)$, тобто фільтр, який неможливо реалізувати. При $\alpha=0,5$ відбувається розширення смуги пропускання на 50%, у той час як при $\alpha=1$ ширина смуги пропускання у два рази більша за мінімальну теоретичну смугу.

Теоретично на частоті $f=(1+\alpha)f_N$ затухання фільтра нескінченно велике. На практиці затухання, як правило, задається на рівні 20...50 дБ в залежності від рівня завад від сусіднього каналу.

Продемонструємо перевагу підходу еквівалентної НЧ фільтрації для систем з відносно низькою швидкістю передавання бітів.

Припустимо, що необхідно спроектувати лінійний двійковий модулятор для швидкості передачі $f_s=1/T_s=10$ кбіт/с. Модулюючий сигнал подається у форматі NRZ без постійної складової, частота несучого колювання 2000 МГц, допустиме розширення смуги частот 30% (тобто, визначається частота, на якій затухання фільтра стане рівним 30 дБ).

Модулятор реалізовано за допомогою змішувача (перемножувача) з передмодуляційним ФНЧ у першому варіанті і післямодуляційним СФ у другому варіанті. Значення коефіцієнта закруглення рівняється $\alpha=0,3$, оскільки допускається 30% розширення смуги частот.

На рис. 1.4а, 1.4б зображені частотні характеристики ФНЧ і смугового фільтра, які задовольняють теоремі Найквіста.

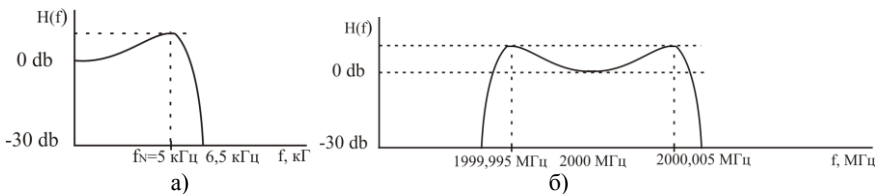


Рисунок 1.4 – АЧХ формуючих ФНЧ та смугового фільтрів, що задовольняють умові передачі без міжсимвольних спотворень

Значення смуги пропускання за рівнем -30 дБ для ФНЧ (рис.1.4а) рівняється:

$$f = (1 + \alpha)f_N = (1 + \alpha)/2T_s = (1 + 0,3) \cdot 5 = 6,5 [\text{Гц}].$$

Для смугового фільтра АЧХ, відповідно до проведеного аналізу, віддзеркалюється відносно частоти носія 2000 МГц (рис.1.4 б). На АЧХ фільтрів також відображено дію амплітудного коректора типу $x/\sin x$.

Для передачі без міжсимвольних спотворень необхідно забезпечити лінійність фазочастотної характеристики фільтрів. Але, для наведеного СФ практично неможливо реалізувати лінійну фазочастотну характеристику. Крім того, будь-яке зміщення центральної частоти фільтра (2000 МГц) призводить до неприпустимої асиметрії. У той же час еквівалентний ФНЧ може бути реалізовано відносно простими засобами як аналогової, так і цифрової обробки сигналів.

Тому за основу для побудови модулятора BPSK обирається варіант з передмодуляційною фільтрацією за допомогою еквівалентного ФНЧ Найквіста.

1.2.2 Структурна схема модему BPSK

На підставі моделі каналу зв'язку з використанням BPSK (рис.1.1) визначимо структурні схеми передавального і приймального трактів і визначимо основні задачі, які виносяться на моделювання.

Як випливає з попереднього аналізу, радіосигнал з BPSK модуляцією формується шляхом перемноження послідовності цифрових даних і гармонічної несучої. Для забезпечення передачі інформації в каналі зв'язку без міжсимвольної інтерференції необхідно для модулюючої послідовності застосувати фільтр Найквіста.

Відповідно в структурній схемі модулятора BPSK, яка буде використовуватися під час моделювання необхідно передбачити генерацію вхідних цифрових даних, перетворення їх до формату NRZ і подальшу обробку модулюючого сигналу у формуючому фільтрі.

Але, під час реалізації фільтра Найквіста необхідно врахувати той факт, що такій обробці підлягають інформаційні біти як у передавальному, так і в приймальному трактах. Тому, якщо позначити

передавальну функцію фільтра Найквіста (вираз 1.13) через G_N , то, як в передавальному, так і в приймальному трактах необхідно реалізувати однакові фільтри з передавальними функціями $G_r(f) = G_r(f) = \sqrt{G_N(f)}$. З (1.13) виходить, що передавальна функція формуючого фільтра модулятора визначиться як

$$|H(jf)| = \begin{cases} 1 & 0 \leq |f| \leq \frac{(1-\alpha)}{2T_s} \\ \cos \left\{ \frac{T_s}{4\alpha} \left[2\pi|f| - \frac{\pi(1-\alpha)}{T_s} \right] \right\}, & \frac{(1-\alpha)}{2T_s} \leq |f| \leq \frac{(1+\alpha)}{2T_s} \\ 0 & |f| > \frac{(1+\alpha)}{2T_s} \end{cases} \quad (1.14)$$

Такий фільтр відомий як корінь квадратний з фільтра Найквіста і позначається як \sqrt{RC} .

Імпульсна характеристика такого фільтра визначається як:

$$g_t(t) = g_r(t) = \frac{1}{\pi t} \cdot \frac{1}{1 - \left(\frac{4\alpha \cdot t}{T_b} \right)^2} \cdot \sin \left\{ 2\pi(1-\alpha) \frac{1}{T_b} \right\} + \frac{1}{\pi} \cdot \frac{4\alpha/T_b}{1 - \left(\frac{4\alpha \cdot t}{T_b} \right)^2} \cdot \cos \left\{ 2\pi(1+\alpha) \frac{1}{T_b} \right\}. \quad (1.15)$$

На практиці формуючий фільтр Найквіста реалізується як цифровий фільтр з кінцевою імпульсною характеристикою (FIR - фільтр) на підставі ліній затримки та суматорів.

Як правило, в сучасній апаратурі цифрова фільтрація виконується за допомогою так званої DSPH (digital signal processor hardware) апаратури, яка передбачає застосування цифрових методів обробки сигналів із використання цифрових сигнальних процесорів.

Одним з таких типових методів цифрової обробки є операція передискретизації.

Передискретизація (resampling) – це зміна частоти дискретизації дискретного (найчастіше цифрового) сигналу. Алгоритми передискретизації широко застосовуються під час обробки звукових сигналів, радіосигналів і зображень.

Відліки сигналів, що відповідають новому значенню частоти дискретизації обчислюються вже за відліками, які були отримані, і не несуть нової інформації. Процес підвищення частоти дискретизації називають інтерполяцією, а пониження – децимацією.

Передискретизація може бути як аналоговою, коли з підвищеною частотою беруться відліки початкового сигналу, так і цифровою коли вже між існуючими відліками вставляються додаткові, що розраховані за допомогою інтерполяції. Інший спосіб отримання проміжних відліків заключається у вставленні нульових відліків між існуючими, після чого послідовність підлягає цифровій фільтрації. В аналогово-цифровому перетворювачі використовується аналогова передискретизація, а у цифро-аналоговому перетворювачі – цифрова.

Ідеальна інтерполяція дозволяє точно поновити значення сигналу у проміжних відліках. Стандартний алгоритм інтерполяції сигналу з цілим коефіцієнтом заключається у наступному:

- 1) відбувається вставка нульових відліків на місце відліків, які необхідно обчислити
- 2) фільтрація сигналу цифровим фільтром нижніх частот, для того щоб усунути спектральні складові сигналу, яких завідомо не було у вхідному сигналі згідно з теоремою Котельникова

В англомовній технічній літературі перший з цих етапів позначається терміном *upsampling* (інколи термін *upsampling* взагалі використовують як синонім інтерполяції). Під час програмної реалізації інтерполяції нульові відліки не приймають участі у обчисленні вихідного сигналу фільтра, що дозволяє оптимізувати процес обчислення.

Коли передискретизація відбувається на частоті, яка у декілька разів перевищує значення частоти дискретизації за теоремою Котельникова, то використовують термін передискретизація із запасом по частоті (англомовний термін - *oversampling*).

Застосування такої передискретизації надає наступні переваги:

- можливість використовувати більш простий аналоговий фільтр для запобігання накладання спектрів дискретизованого сигналу;
- можливість використовувати АЦП з меншою розрядністю.

Такий же підхід використовується і для поновлення сигналу по його відлікам для спрощення аналогового фільтра поновлення або реконструкційного фільтра

Передискретизація із запасом по частоті застосовується для спрощення реалізації ЦАП і АЦП. Дійсно для аналогово-цифрових систем на вході АЦП і виході ЦАП необхідно встановлювати аналоговий фільтр з лінійною АЧХ у робочому діапазоні частот і з різким спадом за його межами.

Реалізація такого фільтра достатньо складна. Це пояснюється тим фактом, що спектр дискретизованого сигналу має періодичний характер, тобто в ньому з'являються дзеркальні відображення спектра недискретизованого сигналу на гармоніках частоти дискретизації, які необхідно придушити. У той же час під час збільшення частоти дискретизації дзеркальні відображення спектра на гармоніках відсуваються від спектра основного сигналу.

Іншими словами, коли між відліками вставляються додатково K нулів, то початковий спектр просто повторюється K разів на гармоніках нової частоти дискретизації. Але при збільшеній частоті ці складові розташовані значно далі по частоті від спектра основного сигналу, що суттєво зменшує вимоги до крутизни спаду частотної характеристики фільтра.

Друга перевага передискретизації із запасом по частоті полягає у тому, що помилки квантування що розподілюються по усьому спектру сигналу, що квантується, під час збільшення частоти дискретизації перерозподіляються по значно ширшій смузі частот, що зменшує шуми квантування для основного сигналу. Відзначається що збільшення частоти дискретизації удвічі зменшує рівень шумів квантування на 3дБ.

Тому, після генерації вхідних даних у вигляді цифрових відліків, в моделі каналу пропонується реалізувати операцію передискретизації із запасом по частоті. Порядок передискретизації прийнято рівним 8, що означає, що нова частота дискретизації у 8 разів більша за частоту слідкування цифрових вибірок сигналу. Реалізується

передискретизація шляхом встановлення 7 нульових відліків після кожного інформаційного біту.

Відповідно до фільтра надходить вже передискретизований сигнал, на виході фільтра формується відфільтрований сигнал у вигляді цифрових відліків.

Для реалізації безпосередньо фазової модуляції у змішувачі, який реалізує функцію перемножувача, необхідно перетворити сигнал на його вході до аналогової форми за допомогою цифро-аналогового перетворювача (ЦАП). Результуючий сигнал після перемножувача і є сигналом який передається через канал зв'язку.

Відповідна структурна схема модулятора BPSK сигналу, яка вибрана за основу для моделювання зображена на рис.1.5.



Рисунок 1.5 – Структурна схема модулятора BPSK сигналу

Розглянемо більш детальніше структурну схему приймального тракту і визначимо основні задачі, які необхідно відобразити в процесі моделювання.

Структурна схема приймального тракту для методу BPSK, яка пропонується для моделювання, зображена на рис.1.6.

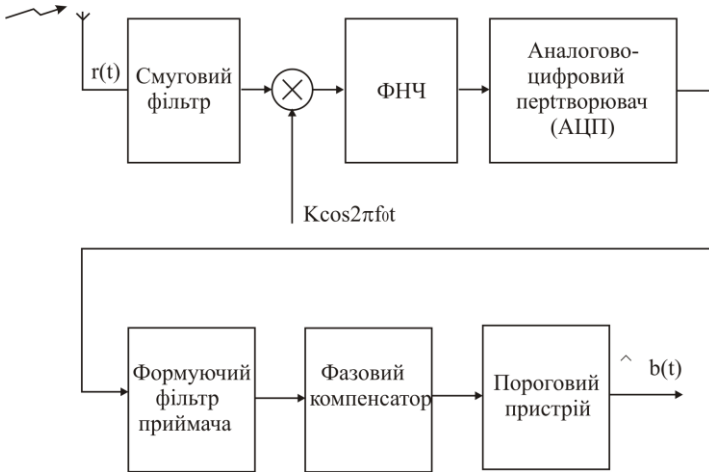


Рисунок 1.7 – Структурна схема приймального тракту для методу BPSK

На приймальному боці прийнятий антеною сигнал, по-перше, проходить через смуговий фільтр, який забезпечує придушення усіх позасмугових завадових сигналів. Наступним кроком виступає перенесення радіосигналу до основної смуги частот. Як випливає з рис.1.6, при цьому використовується схема когерентного прийому, для якої на приймальному боці формується поновлене несуче коливання. В результаті перемноження прийнятого сигналу і поновленої несучої після ФНЧ утворюється сигнал в основній смузі частот з якого необхідно виділити інформаційну послідовність.

Оскільки на приймальному боці застосовується такий же формуючий цифровий фільтр \sqrt{RC} , як і на передавальному боці, то сигнал знову необхідно подати у вигляді цифрових вибірок, для чого використовується аналогово-цифровий перетворювач (АЦП). Для відфільтрованого дискретизованого сигналу необхідно обрати точку синхронізації і провести операцію зворотну передискретизації.

Останнім кроком є застосування порогового пристрою, який приймає рішення за простим алгоритмом: якщо значення вибірки більше 0, то приймається рішення про прийом логічної “одиниці”,

якщо ж значення вибірки менше 0, то приймається рішення про прийом логічного “нуля”.

На виході порогового пристрою формується цифровий сингал $\left\{ \hat{b}_k \right\}$. Якщо завади і шум в системі малі, то $\hat{b}_k(t) = b_k(t - \tau)$, тобто вихідний демодульований або регенований сигнал рівняється сигналу джерела інформації із затримкою на τ (затримка в обладнанні і під час розповсюдження радіохвиль).

1.3 Реалізація програмного алгоритму для AWGN каналу

Основною метою лабораторної роботи є створення моделі, яка відповідає основним аспектам реалізації модему BPSK в основній смузі частот з урахуванням особливостей методів цифрової обробки сигналів. З метою певного спрощення задачі поновлення несучого коливання, а також питання тактової синхронізації не розглядаються.

На першому етапі досліджень головною задачею виступає створення моделі каналу зв'язку з урахуванням впливу тільки білого шуму – такий канал в технічній літературі відомий як AWGN (Additive White Gaussian Noise) канал.

Моделювання проводитиметься в середовищі Matlab, мова програмування якого алгоритмічно близька до програмних алгоритмів, що застосовуються у цифрових сигнальних процесорах, які звичайно використовуються для реалізації сучасних модемів.

У відповідності до проведеного аналізу на рис.1.7 зображена блок-схема алгоритму моделювання модема.

Для моделювання процесу модуляції за методом BPSK і розрахунку ймовірності бітової помилки (або BER – Bit Error Ratio) використовуються основна програма і три підпрограми, повний лістинг яких наведено у додатку А [3]. Основною програмою є програма bpsk.m.

1.3.1 Опис основної програми

По-перше вводяться початкові дані для моделювання. Введемо такі позначення:

sr [кбіт/с] – швидкість передачі символів;

br [кбіт/с] – швидкість передачі інформаційних бітів;

m_l – кількість бітів, що передаються за один символ. Як відомо, кількість бітів, що переносяться за один символ визначається наступним виразом: $m_l = \log_2 M$, де M – основа обраної системи числення. Оскільки аналізується двійкова фазова маніпуляція (BPSK, $M=2$), то очевидно, що для такого варіанту $m_l=1$;

pd – кількість переданих символів, для яких буде проводиться розрахунок ймовірності помилкового прийому за одну реалізацію випадкового процесу (один цикл моделювання);

ebn_0 – початкове значення енергії, що приходить на один біт E_b , до спектральної щільності шуму N_0 . Як відомо, у цифрових системах зв'язку аналогом відношення сигнал/шум, яке характеризує аналогові системи, виступає саме параметр E_b/N_0 , оскільки для цифрової системи необхідно враховувати швидкість передавання інформаційних бітів у визначеній смузі частот.

IPOINT – порядок передискритизації.

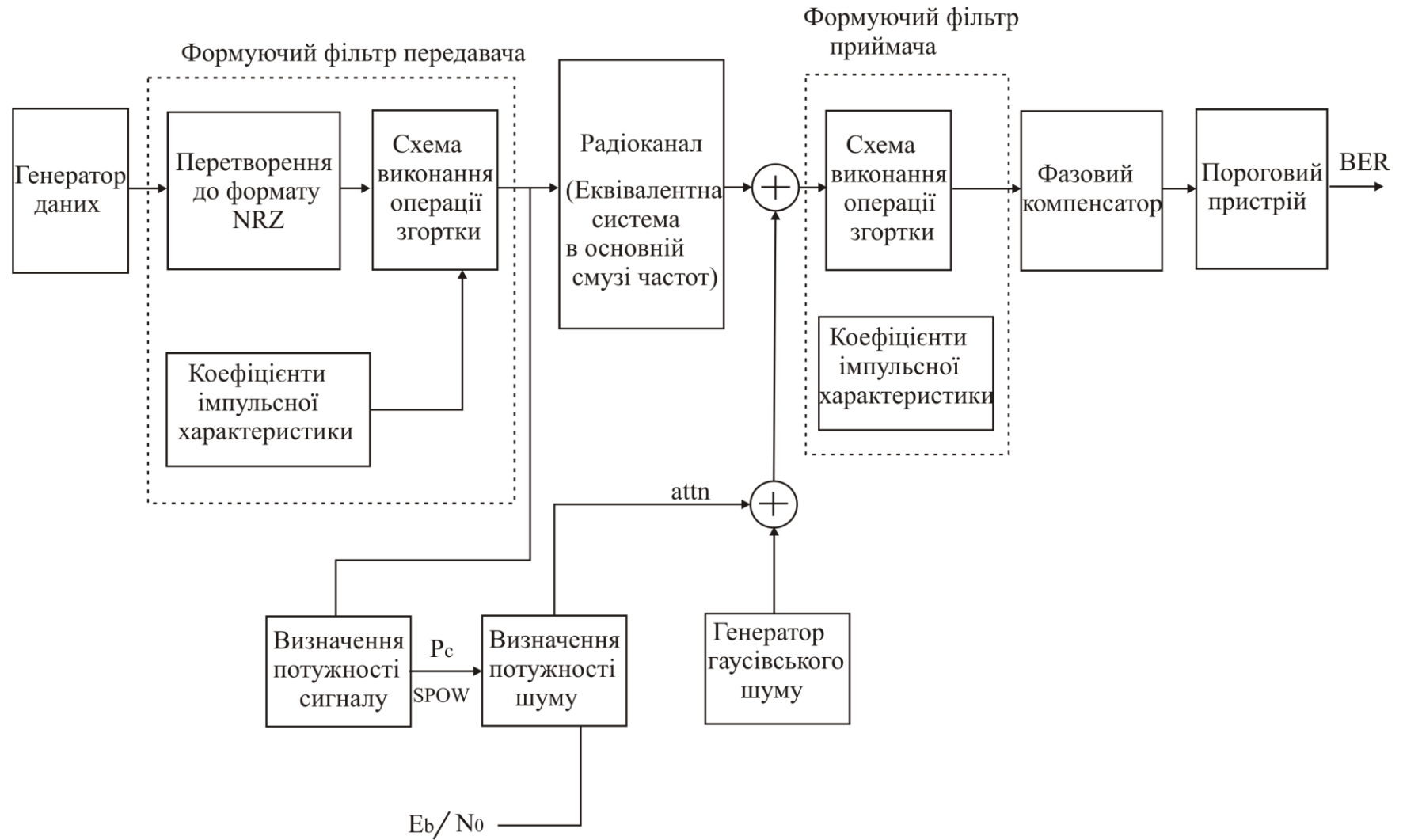


Рисунок 1.7 – Блок-схема алгоритму моделювання модему BPSK

В термінах алгоритмічної мови програмування середовища Matlab блок введення початкових даних набуває наступного вигляду [3]:

```
% bpsk.m
% Simulation program to realize BPSK transmission system
%*****Preparatin part*****
sr=256000.0;          % Symbol rate
ml=1;                % ml : Number of modulation levels
br=sr.*ml;           % bit rate
nd=100;              % Number of symbols
ebn0=3;              % Eb/N0
IPOINT=8;            % Number of oversamples
```

Перші три рядки, що починаються з символу %, є коментарями програми, які визначають ім'я головної програми bpsk.m, метод модуляції - BPSK і, власне, відокремлюють підготовчий блок для завдання основних параметрів каналу зв'язку.

Швидкість передавання символів sr задана рівною 256000 біт/с, або 256 кбіт/с, кількість бітів ml , що переносяться за один радіосимвол, як було визначено, для BPSK рівняється одиниці, відповідно бітова швидкість br визначається як результат перемноження символної швидкості на параметр ml , що у випадку BPSK означає, що бітова швидкість рівняється символній швидкості.

Кількість символів, що підлягають аналізу для однієї реалізації випадкового процесу прийнята рівною $nd=100$, початкове значення параметру $E_b/N_0=3$ дБ.

Оскільки розглядається цифрова реалізація модему, то застосовується операція передискретизації цифрових вибірок сигналу і, відповідно, порядок передискретизації в програмі прийнято рівним $IPOINT=8$.

Наступним блоком програми є завдання параметрів формуючого фільтра Найквіста.

Оскільки, формуючий фільтр застосовується як на передавальному, так і на приймальному боці, то застосовується фільтр "корінь квадратний" з фільтра Найквіста або фільтр типу \sqrt{RC} . Для моделювання відклику такого фільтра на вхідну послідовність даних необхідно обчислити імпульсну перехідну характеристику фільтра.

Коефіцієнти імпульсної характеристики фільтра \sqrt{RC} розраховуються на підставі виразу (1.15) за допомогою підпрограми **hrollcoef**(irfn, IPOINT,sr,alfs,1/0)

Відповідно наступний блок програми має такий вигляд:

```
% *****Filter initialization*****
irfn=21; % Number of taps
alfs=0.5; % Rolloff factor
[xh]=hrollcoef(irfn, IPOINT,sr,alfs,1);
% Tranamitter filter coefficients
[xh2]=hrollcoef(irfn, IPOINT,sr,alfs,0);
% Receiver filter coefficients
% *****
```

Параметрами підпрограми є кількість ліній затримки фільтра **irfn**, раніше визначені порядок передискретизації **IPOINT** і символічна швидкість **sr**, а також коефіцієнт закруглення **alfs**. Значення останнього п'ятого параметра функції **hrollcoef**, яке може приймати значення або 1, або 0, визначає приналежність фільтра до передавача каналу зв'язку (відповідний параметр рівняється 1), або до приймача (параметр прийнято рівним 0).

Для моделювання кількість ліній затримки прийнята рівною **irfn=21**, а значення коефіцієнта закруглення **alfs=0.5**.

Частота на якій згасання фільтра якнайменше на 30 дБ буде меншим порівняно зі смугою пропускання визначиться як:

$$f_{-30\text{dB}} = (1+0,5) \cdot 256 \cdot 10^3 / 2 = 1,5 \cdot 128 = 192[\text{кГц}].$$

Амплітудно-частотна і фазочастотна характеристики формуючого фільтра, що отримані за допомогою програми **hrollcoef**, для символічної швидкості 256 кбіт/с зображені на рис.1.8.

Як видно з рис. 1.8 фазочастотна характеристика фільтра лінійна у смузі пропускання фільтра, що є необхідною умовою під час реалізації формуючого фільтра.

На рис.1.9 представлена імпульсна характеристика формуючого фільтра \sqrt{RC} .

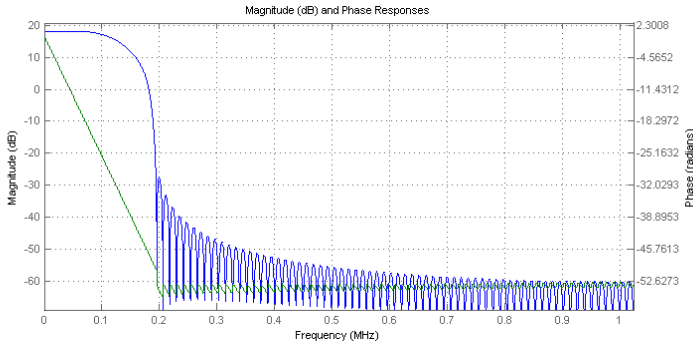


Рисунок 1.8 – Амплітудно-частотна і фазочастотна характеристики формуючого фільтра Найквіста

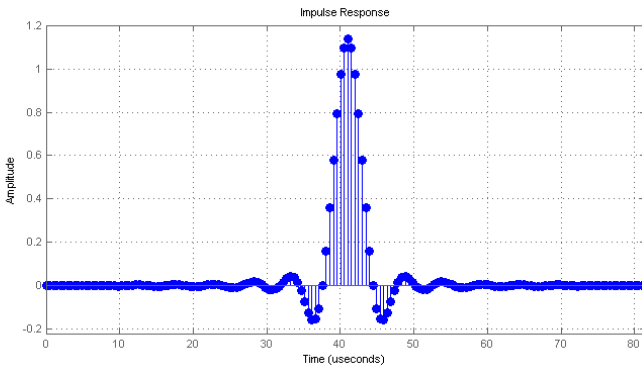


Рисунок 1.9 – Імпульсна характеристика формуючого фільтра Найквіста

Подалі починається основне тіло програми.

Введемо наступні позначення:

nloop – кількість циклів моделювання;

pod – кількість переданих бітів;

poe – кількість помилково декодованих бітів на приймальній стороні.

Для того, щоб адекватно визначити ймовірнісні характеристики каналу зв'язку задамо кількість циклів моделювання (кількість реалізацій випадкового процесу для послідовності з 100 біт) достатньо великою - приймаємо значення nloop=1000. Початкові значення

кількості переданих і помилково декодованих бітів приймаються рівними нулю.

Змінну циклу позначимо через **iii**, а початок циклу задається оператором **for iii=1:nloop**. Відповідний фрагмент програми набуває наступного вигляду:

```
% *****Start calculations*****
nloop=1000;    % Number of simulation loops
noe=0;        % Number of error data
nod=0;        % Number of transmitted data
for iii=1:nloop
...
Тіло циклу
...
end
```

Після визначення усіх змінних можна приступити безпосередньо до створення моделі каналу зв'язку з використанням методу BPSK у відповідності з блок-схемою алгоритму рис.1.7 (тіло циклу).

По-перше, генерується випадкова інформаційна послідовність усі елементи якої обираються з ансамблю $\{0,1\}$. Вхідні дані задаються у вигляді вектора розмірністю $1 \cdot nd \cdot ml$ за допомогою оператора:

```
% *****Data generation*****
data1=rand(1,nd*ml)>0.5    % rand: built in function.
% *****
```

Можлива реалізація послідовності вхідних даних поданих у вигляді цифрових відліків зображена на рис. 1.10 (показані перші двадцять відліків, кількість відліків визначається відповідним оператором виводу середовища Matlab).

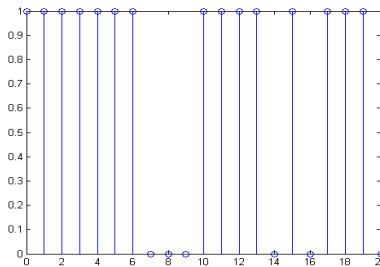


Рисунок 1.10 – Вхідна послідовність даних у вигляді відліків

Наступним кроком є перетворення однополярних цифрових відліків на двохполярну послідовність у форматі NRZ (тобто рівень логічної одиниці відповідає цифровому відліку з рівнем +1, а рівень логічного нуля цифровому відліку з рівнем -1):

```
%*****BPSK Modulation*****
data1=data.*2-1;
```

Результат перетворення послідовності вхідної послідовності відліків до формату NRZ зображено на рис.1.11.

Для виводу результатів моделювання у вигляді графіків рис.1.10, рис.1.11 (послідовності цифрових відліків) після закінчення тіла циклу застосовані такі оператори Matlab:

```
figure; stem(0:20,data(1:21),'DisplayName','data');
figure; stem(0:20,data1(1:21),'DisplayName','data1');
```

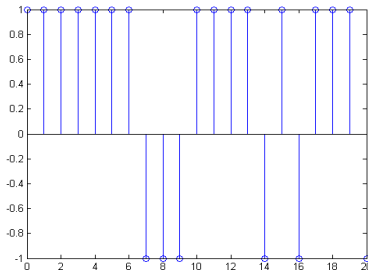


Рисунок 1.11 – Послідовність вхідних відліків у форматі NRZ

Далі проводиться операція передискретизації, а саме на часовому інтервалі, що відповідає тривалості одного біта проводиться додавання нульових відліків згідно з обраним порядком дискретизації.

Операція передискретизації виконується за допомогою підпрограми oversamp (data1,nd,IPOINT) (додаток A3):

```
[data2]= oversamp(data1,nd,IPOINT);
```

Результат виконання операції передискретизації для вхідної послідовності у форматі NRZ відображено на рис.1.12.

Сформовані відліки у відповідності з функціональною схемою модулятора підлягають обробці у формуючому фільтрі Найквіста. Як відомо сигнал на виході будь-якого фільтра можна отримати за

допомогою операції згортки вхідного сигналу й імпульсної характеристики фільтра.

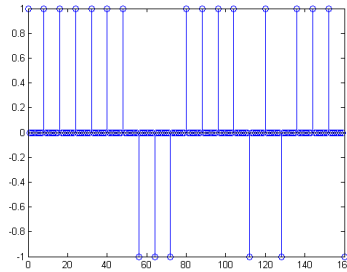


Рисунок 1.12 – Послідовність вхідних відліків після передискретизації

Для реалізації згортки застосовується вбудована функція Matlab **conv** (convolution):

```
% *****BPSK Modulation*****
data3=conv(data2,xh);
% *****
```

Вихідний сигнал формуючого ФНЧ у вигляді послідовності цифрових відліків, як результат виконання останнього оператора зображено на рис.1.13.

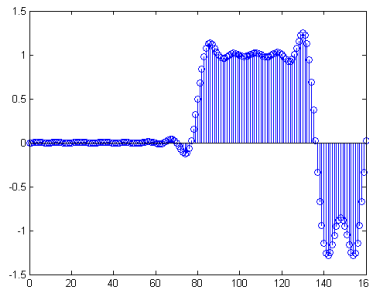


Рисунок 1.13– Сигнал на виході формуючого фільтра Найквіста

Оскільки безпосередньо процес радіочастотної модуляції, а саме перемноження відфільтрованого сигналу з сигналом несучої частоти здійснюється аналоговими засобами, то отриманий сигнал у вигляді

цифрових відліків необхідно подати до ЦАП і цей аналоговий сигнал подати до змішувача. Форма модулюючого сигналу після ЦАП показана на рис.1.14. Вивід графіка рис.1.14 (тобто вже реального аналогового модулюючого символу) реалізується таким оператором: `figure;plot(0:20*IPOINT,data3(1:20*IPOINT+1),'DisplayName','data');`

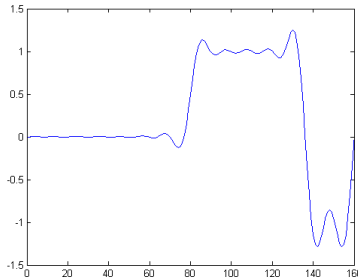


Рисунок 1.14 – Сформований сигнал на виході ЦАП

Сформований сигнал використовується для безпосередньої модуляції високочастотної несучої з наступним підсиленням у радіочастотному тракту і передачею до каналу зв'язку.

На приймальному боці сигнал підлягає впливу білого гаусівського шуму - (AWGN) канал.

Визначимо алгоритм введення білого шуму до приймального тракту каналу зв'язку, а також визначимо зв'язок між напругою шуму, що додається до каналу, і параметром E_b/N_0 . Енергія, що приходить на один біт E_b визначається з співвідношення [2,3]:

$$E_b = P_c T_b = \frac{P_c}{F} \quad [\text{Вт} \cdot \text{с/бит}], \quad (1.16)$$

де P_c – потужність сигналу, $F=1/T_b$ – бітова швидкість

Відповідно спектральну щільність шуму N_0 можна визначити як:

$$N_0 = \frac{P_{ш}}{R} \quad [\text{Вт/Гц}], \quad (1.17)$$

де $P_{ш}$ – потужність сигналу, R – символна швидкість, яку можна прийняти рівною смузі пропускання приймача.

Тоді поєднуючи (1.16), (1.17) можна отримати:

$$\frac{E_b}{N_0} = \left(\frac{P_c}{P_\phi} \right) \cdot \frac{R}{F} ,$$

або

$$P_{ш} = \frac{P_c}{F} \cdot \frac{R}{E_b/N_0} . \quad (1.18)$$

Оскільки E_b/N_0 задається, як правило, в децибелах, то можна останнє рівняння записати у вигляді:

$$P_{ш} = \frac{P_c}{F} \cdot \frac{R}{10^{\frac{E_b/N_0}{10}}} . \quad (1.19)$$

При моделюванні припускаємо, що амплітуда послідовності даних вимірюється у вольтах, тому шумовий сигнал, який додається до каналних символів, також повинен мати розмірність [Вольт].

Враховуючи, що для загального варіанту застосування I, Q модуляторів, шум однаково розподілений в I та Q каналах модулятора, то остаточно введемо шумову напругу $attn$ як

$$attn = \sqrt{\frac{1}{2} \cdot P_{ш}} . \quad (1.19)$$

В програмі цей алгоритм реалізовано наступними операторами:

```
% *****Attenuation Calculation*****
spow=sum(data3.*data3)/nd;
% sum: built in function
attn=0.5*spow*sr/br*10.^(-ebn0/10);
attn=sqrt(attn); % sqrt: built in function
*****Add White GaussianNoise (AWGN)*****
```

```
inoise=randn(1,length(data3)).*attn; % add white Gaussian noise
```

В результаті сигнал на низькочастотному вході демодулятора розглядається як сума сигналу переданого через канал зв'язку data3 і гаусівського шуму (AWGN канал):

```
data4=data3+inoise;
```

Сигнал основної смуги частот на вході демодулятора з врахуванням шумів каналу зв'язку, якого отримано за такою моделлю, зображено на рис.1.15.

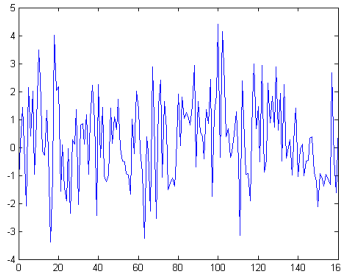


Рисунок 1.15 – Суміш сигналу з шумом на вході демодулятора

На приймальному боці застосовується такий же фільтр Найквіста, як і на передавальному боці. Коефіцієнти імпульсної характеристики фільтра \sqrt{RC} визначені у програмі як масив [xh2].

Відповідно для здійснення фільтрації необхідно аналогічно до модулятора реалізувати згортку вхідного сигналу демодулятора з імпульсною характеристикою фільтра:

```
data5=conv(data4,xh2);
```

Дискретний сигнал на виході формуючого фільтра (data5) зображено на рис.1.16.

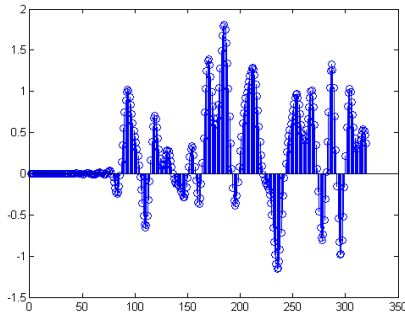


Рисунок 1.16 – Сигнал на виході формуючого фільтра Найквіста у приймачі (data5)

Але, з аналізу рис.1.16 очевидно, що в процесі демодуляції вихідних даних необхідно врахувати затримку дискретизованого сигналу, яка виникає через наявність ліній затримки у цифровому фільтрі.

Тривалість затримки визначається довжиною вектора [xh]. Відповідно для кожного застосування функції згортки conv.m час затримки визначиться як $(\text{irfn} + \text{IPOINT})/2$. Оскільки застосовуються дві функції conv.m, то повний час затримки складе $\text{irfn} + \text{IPOINT}$.

Оскільки після формуючого фільтра сигнал ще має вигляд передискретизованого сигналу, то зворотний процес (тобто позбавлення передискретизації) повинен починатися з відліку $\text{irfn} * \text{IPOINT} + 1$. Цю змінну позначимо як samp1.

Тепер, якщо з послідовності на виході формуючого фільтра виділити кожен 8-му вибірку починаючи з початкової точки samp1, то отримаємо демодульований дискретизований сигнал у вигляді двохполярних вибірок з частотою повторення 256 кбіт/с, які позначимо як data6:

```
samp1=irfn*IPOINT+1
data6=data5(samp1:8:8*nd+samp1-1);
```

Відповідно на рис.1.17 зображено послідовність двохполярних демодульованих відліків.

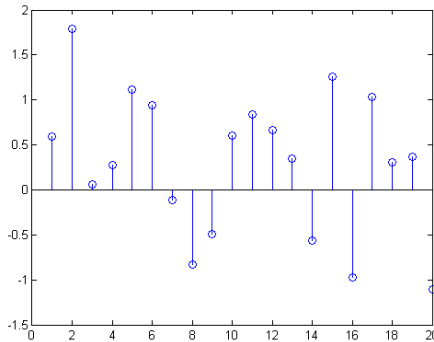


Рисунок 1.17 – Демодульована двополярна послідовність вихідних даних

Сам процес формування демодульованої послідовності у вигляді цифрових відліків, яка складається з ансамблю $\{0,1\}$ реалізується таким оператором:

```
%*****BPSK Demodulation*****
[demodata]=data6>0;.
```

В результаті роботи порогового пристрою на виході демодулятора буде сформована послідовність у вигляді демодульованих цифрових відліків, які складаються з ансамблю $\{0,1\}$.

Для розрахунку ймовірності помилкового прийому одного біта необхідно визначити відношення кількості помилково прийнятих бітів до загальної кількості усіх переданих бітів.

Для визначення кількості переданих даних необхідно розрахувати довжину вектора даних. В термінах Matlab таку операцію можна виконати за допомогою вбудованої функції “length”. Визначимо у програмі кількість переданих даних як змінну **nod** (number of data).

Розрахунок кількості помилок проводиться за наступним алгоритмом. Спочатку проводиться віднімання від однойменних бітів переданої вхідної послідовності даних (**data**) відповідних бітів демодульованої послідовності на приймальному боці (**demodata**).

Якщо помилок на прийимальному боці не виявлено, то отримаємо нульовий вектор довжиною “nod” (кількість даних). У випадку ж з’явлення помилково декодованих бітів, то отримаємо вектор довжиною “nod” в якому на позиціях, що відповідають помилково декодованим бітам з’являться значення +1 або -1.

Застосувавши операцію виділення абсолютного значення можна отримати вектор, який матиме одиничні значення на помилкових позиціях. Якщо тепер провести операцію арифметичного додавання усіх елементів сформованого вектора, то отримаємо загальну кількість помилок, які виникли під час демодуляції. Визначимо розраховану за таким алгоритмом кількість помилок як змінну “noe” (number of errors)

Після проведення зазначених процедур можна розрахувати ймовірність помилкового прийому одного біта як відношення кількості помилково прийнятих бітів до загальної кількості переданих бітів. В термінах Matlab, послідовність операцій така:

```
%*****Bit Error Rate (BER)*****
nod2=sum(abs(data-demodata));
    % sum: built in function
nod2=length(data1);    % length: built in function
noe=noe+nod2;
nod=nod+nod2;
end    % for iii=1:nloop
%*****Output result *****
ber=noe/nod;
```

Теоретичний вираз для ймовірності помилкового прийому одного біта або коефіцієнта бітової помилки (BER) для AWGN каналу з використанням BPSK при умові відносного кодування даних і реалізації когерентного детектування має такий вигляд: [2,3]:

$$\text{BER}_{\text{BPSK,AWGN}} = 0,5\text{erfc}(\sqrt{E_b/N_0}). \quad (1.20)$$

Для заданих у програмі **bpsk.m** початкових умов вивід результатів, щодо ймовірності помилкового прийому, буде виводитися тільки для одного конкретного значення E_b/N_0 .

Для отримання графічної залежності $\text{BER}=F(E_b/N_0)$ необхідно створити новий m-файл під назвою **bpsk_ber.m**. Практично він

повторює лістинг файлу bpsk.m, але, з метою формування графічної залежності $BER=F(E_b/N_0)$ перед основним циклом (оператор `for iii=1:nloop`) створюється додатковий цикл за параметром E_b/N_0 .

З цією метою вводяться три нових масиви:

- 1) `ebn0_array` – масив значень E_b/N_0 від 0 до 35 дБ з кроком 1 дБ;
- 2) `ber_array` – масив значень BER, який визначається за результатами моделювання для відповідних значень E_b/N_0 ;
- 3) `ber_array_theory` – масив значень BER, який визначається з теоретичного виразу (1.20) для відповідних значень E_b/N_0 .

Відповідно цикл реалізовано такими операторами:

```
ebn0_array=0:1:35;
ber_array=zeros(length(ebn0_array),1);
ber_array_theory=zeros(length(ebn0_array),1);
for ebn0_loop=1: length(ebn0_array) ;
ebn0= ebn0_array (ebn0_loop) ;
for iii=1:nloop
....
Тіло циклу
...
end
```

Для виводу результатів щодо BER необхідно після основного циклу визначити розраховані масиви помилок, а саме:

```
ber_array(ebn0_loop)=ber;
ber_array_theory(ebn0_loop)=0.5*erfc(sqrt(10^(ebn0_array...
(ebn0_loop)/10));
end % for ebn0_loop.
```

Графік необхідно вивести у напівлогарифмічному масштабі, тобто для коефіцієнта BER застосувати логарифмічний масштаб, а для E_b/N_0 застосувати лінійний масштаб у дБ. Для цього застосовуються такі оператори:

```
h=semilogy(ebn0_array, ber_array,'o', ebn0_array, ber_array_theory);
set(h,{'DisplayName'},{'Experiment','Theory'});
legend show.
```

Графічні залежності для BER, що виводяться як результат моделювання за запропонованим алгоритмом, відповідають експерименту (позначення як 'o') і теоретичному виразу (1.20), що дозволяє оцінити адекватність моделі каналу зв'язку з

використанням методу BPSK основним теоретичним положенням.

1.4 Експериментальна частина

1.4.1 Відкрити середовище Matlab. Створити нову директорію (папку) LAB5, в яку необхідно розмістити основну програму, а також усі підпрограми, що необхідні для моделювання.

1.4.2 Скопіювати файл основної програми `bpsk.m`, який знаходиться у додатку А методичних вказівок, і повернутися до середовища Matlab.

1.4.3 Відкрити на панелі інструментів меню File і у відкритому вікні обрати команду створення нового М-файлу. Вставити скопійований файл `bpsk` до редактора. Зберегти скопійований файл під ім'ям `bpsk.m` у створеній директорії LAB5.

1.4.4 Скопіювати у додатку А підпрограму `hrollcoef.m`, і повторити процедуру створення нового М-файлу в директорії LAB5. Файл зберегти під ім'ям, що повторює назву функції –`hrollcoef`.

1.4.5 Скопіювати у додатку А підпрограму `oversamp.m`, що виконує операцію передискретизації, і повторити процедуру створення нового М-файлу в директорії LAB2. Файл зберегти під ім'ям, що повторює назву функції –`oversamp`.

Після виконання визначених пунктів в основному вікні середовища Matlab усі три М-файли повинні відобразитися у полі поточної директорії (Current Directory) – рис.1.18.

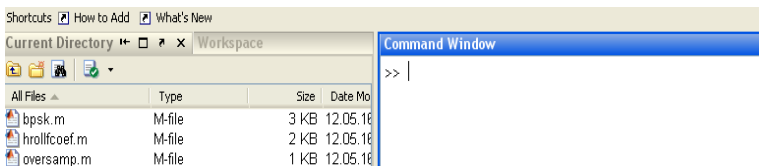


Рисунок 1.19 – Необхідний набір М-файлів у директорії LAB5

1.4.6 У командній строчці вікна (Command Window) набрати назву основної програми `bpsk` і запустити аналіз натисканням клавіші Enter.

Якщо синтаксичні помилки у програмах відсутні, то на дисплей у командному вікні повинен відобразитися потік значень BER для кожної реалізації випадкової послідовності з 100 біт відповідно до заданого циклу `loop` і для заданого значення E_b/N_0 .

1.4.7 Зафіксувати епюри коливань у контрольних точках схеми. Як було визначено відповідні дані виводяться у вигляді графіків за допомогою операторів **figure; stem (...)**, якщо графічна залежність відповідає дискретному сигналу, або **figure; plot (...)**, якщо відтворюється неперервний сигнал.

1.4.8. Повторити експеримент при більшому значенні $E_b/N_0=12$ дБ. Для цього необхідно відкрити головну програму `bpsk.m` і змінити у блоці завдання вхідних параметрів значення параметру `ebn0=12`.

Додатково, для більшої наочності демонстрації операції передискретизації провести вивід послідовності передискретизованих вхідних даних (у програмі це data2) тільки для двох інформаційних бітів.

Для цього у блоці операторів виводу ввести необхідні зміни в операторі виводу:

```
figure; stem(0:20*IPOINT,data2(1:20*IPOINT+1),'DisplayName','data2');
```

Зберігти зміни у файлі `bpsk.m` натисканням на кнопку Save.

1.4.9 Отримати графічну залежність ймовірності помилкового прийому біта (або коефіцієнта BER) від відношення E_b/N_0 .

Для цього необхідно в додатку А скопіювати програму `bpsk_ber` і розмістити її як М-файл у тій же директорії LAB5 (тобто в директорії LAB5 повинні знаходитися чотири М-файли `bpsk.m`, `bpsk_ber.m`, `hrollcoef.m`, `oversamp.m`).

Набрати у командній строчці `bpsk_ber` і запустити аналіз. Результатом роботи програми виступає графік залежності BER від E_b/N_0 .

Отриманий графік подається не досить у зручному вигляді. Дійсно, оскільки ймовірність може приймати значення від 0 до 1, то при автоматичному виборі масштабу для осі Y мінімальне значення ймовірності відповідає 10^{-50} (рис.1.19), що в принципі не має фізичного сенсу.

Тому необхідно на панелі інструментів відкрити вкладку Edit і ініціювати команду `Axes properties` (рис.1.20).

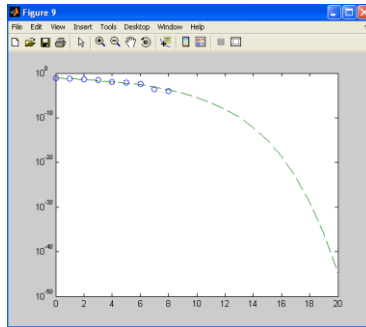


Рисунок 1.19 – Початковий вид залежності BER від E_b/N_0

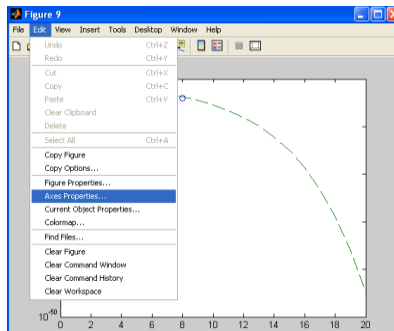


Рисунок 1.20 – Вибір масштабу графіка

Після завантаження відкривається поле, яке дозволяє встановлювати масштаб графіку. Натиснути кнопку Y Axis і встановити діапазон значень ймовірності по осі ординат від 0,00001 (10^{-5}) до 1. Натисканням на поле графічної залежності отримати графік у зручному вигляді для реальних значень бітової помилки.

1.5 Зміст звіту

1.5.1 Структурні схеми модуляторів і демодуляторів, що досліджуються. Блок-схема алгоритму програми.

1.5.2 Результати експериментальних досліджень у вигляді епюр коливань для усіх контрольних точок схеми та графічних залежностей.

1.5.3 Лістинги програм

1.5.4 Висновки за результатами роботи.

1.6 Контрольні запитання

1. Особливості цифрових методів модуляції.
2. Двійкова фазова маніпуляція. Поняття комплексної обвідної.
3. Поняття сигнального сузір'я.
4. Функціональна схема модулятора BPSK.
5. Функціональна схема когерентного демодулятора BPSK.
6. Призначення формуючого фільтра.
7. Вимоги до формуючого фільтра.
8. Як програмно реалізується операція фільтрації?
9. Яке співвідношення описує зв'язок між відношенням сигнал/шум і параметром E_b/N_0 ?
10. Як вводиться шум у моделі відповідно до заданого значення E_b/N_0 ?
11. Методика визначення BER.
12. Пояснити визначення функції $erfc$.
13. Пояснити сенс операції передискретизації у частотній області.
14. Привести епюри коливань для двох інформаційних бітів до передискретизацій і після передискретизації.

2 ЛАБОРАТОРНА РОБОТА № 6

ДОСЛІДЖЕННЯ КАНАЛУ ЗВ'ЯЗКУ З ЧАСТОТНО-НЕСЕЛЕКТИВНИМИ ЗАВМИРАННЯМИ

2.1 Мета роботи

2.1.1 Ознайомитися з основними аспектами проявлення завмирань у каналі зв'язку з рухомими об'єктами

2.1.2 За допомогою ЕОМ експериментально відпрацювати методику моделювання частотно-неселективних завмирань.

2.2 Завдання на підготовку до виконання лабораторної роботи

2.2.1 Канал з релеївськими завмираннями

На трасі від антени базової станції до антени мобільної станції сигнал підлягає впливу різноманітних перешкод, таких як дерева, будівлі, інші відбиваючі поверхні, а всередині будинків до відзначених перешкод додаються елементи внутрішнього інтер'єру. В результаті, через ефекти відбиття радіохвиль, дифракції та розсіювання утворюється багатопроменева картина розповсюдження. Хвиля від антени базової станції розповсюджується у всіх напрямках, і до мобільної станції з різною затримкою надходять окрім основного ще декілька променів (рис.2.1).

В результаті їх суперпозиції у точці прийому змінюється амплітуда сигналу, а також його фаза. Флуктуації сигналу за амплітудою, фазою та кутом напряму у точці прийому призводять до ефекту який отримав назву федингу. Фединг призводить до з'явлення частотно-селективних та швидких завмирань (релеївські завмирання) які призводять до неприпустимого збільшення ймовірності помилкового прийому або коефіцієнта BER.

Падаюча n -а хвиля (відбита або як результат дифракції на перешкоді) приходить до антени МС під кутом θ_n відносно напряму руху МС (рис. 2.1).

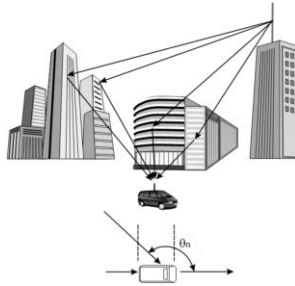


Рисунок 2.1 – Ілюстрація багатопроменевого розповсюдження

Рівняння такої хвилі буде мати вигляд

$$r_n(t) = \text{Re}[e_n(t) \exp j(2\pi f_c t)], \quad (2.1)$$

де $\text{Re}[\cdot]$ означає дійсну частину комплексного числа, яке репрезентує собою комплексну обвідну хвилі, що надходить з відповідного напрямку, f_c - частота сигналу, що передається БС, а $e_n(t)$ визначається таким виразом [3]:

$$e_n(t) = R_n(t) \exp j \left(-\frac{2\pi(L_n - vt \cos \theta_n)}{\lambda} + \phi_n \right) = x_n(t) + jy_n(t), \quad (2.2)$$

де L_n – довжина шляху від БС в метрах, v - швидкість МС в м/с, λ – довжина хвилі в метрах, R_n и ϕ_n – обвідна і фаза падаючої n -ї хвилі, $x_n(t)$ и $y_n(t)$ – синфазна и квадратурна складові $e_n(t)$ відповідно.

Падаюча n -а хвиля зсуває несучу частоту на $v \cos \theta_n / \lambda$ Гц, цей ефект відомий як доплерівський зсув. Максимальне значення доплерівського зсуву рівняється v / λ , коли напрям падаючої хвилі збігається з напрямом руху МС.

Це відображається у рівнянні (2.3) для прийнятої хвилі $r(t)$, яка є синтезом усіх відзначених падаючих хвиль.

$$r(t) = \sum_{n=1}^N r_n(t) = \operatorname{Re} \left[\left(\sum_{n=1}^N r_n(t) \right) \exp j(2\pi f_c t) \right] = \quad (2.3)$$

$$= \operatorname{Re}[(x(t) + iy(t))(\cos 2\pi f_c t + j \sin 2\pi f_c t)] = x(t) \cos 2\pi f_c t - y(t) \sin 2\pi f_c t,$$

де $x(t) = \sum_{n=1}^N x_n(t)$ и $y(t) = \sum_{n=1}^N y_n(t)$ характеризуються нормальним законом розподілу з середнім значенням рівним 0 и дисперсією σ . Функція розподілу щільності ймовірності має вигляд

$$p(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right), \quad (2.4)$$

де $x = x(t)$, $y = y(t)$.

Окрім того, та ж сама функція $r(t)$ може бути визначена через значення амплітуди і фази прийнятої хвилі:

$$r(t) = R(t) \cos(2\pi f_c t + \theta(t)), \quad (2.5)$$

де

$$R(t) = R = \sqrt{x^2 + y^2}, \quad (2.6)$$

$$\theta(t) = \theta = \frac{1}{\tan(y/x)}. \quad (2.7)$$

Замінюючи змінні, можна (2.4) перетворити на залежність $p(R, \theta)$:

$$p(R, \theta) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{R^2}{2\sigma^2}\right). \quad (2.8)$$

Інтегруючи (2.8) по θ на проміжку від 0 до 2π і по R на інтервалі від 0 до нескінченності відповідно отримуємо:

$$p(R) = \frac{R}{\sigma^2} \exp\left(-\frac{R^2}{2\sigma^2}\right), \quad (2.9)$$

$$p(\theta) = \frac{1}{2\pi}. \quad (2.10)$$

З цих рівнянь виходить, що флуктуації обвідної відбуваються за релєвським розподілом, а фазові флуктуації рівномірні в завмираннях на трасі розповсюдження.

Відповідно до рис. 2.1, МС приймає радіохвилі з кутом надходження θ_n , що розподілений рівномірно, а N – порядковий номер хвилі, що надходить до МС.

Тоді, флуктуації завмирань можна подати у вигляді [3]:

$$\begin{aligned} r(t) &= x(t) + j \cdot y(t) = \\ &= \left[\sqrt{\frac{2}{N_1 + 1}} \sum_{n=1}^{N_1} \sin\left(\frac{\pi n}{N_1}\right) \cos\left\{2\pi f_d \cos\left(\frac{2\pi n}{N_1}\right)t\right\} + \frac{1}{\sqrt{N_1 + 1}} \cos(2\pi f_d t) \right] + \\ &\quad + j \sqrt{\frac{2}{N_1}} \sum_{n=1}^{N_1} \sin\left(\frac{\pi n}{N_1}\right) \cos\left\{2\pi f_d \cos\left(\frac{2\pi n}{N_1}\right)t\right\}, \end{aligned} \quad (2.11)$$

де $N_1 = \frac{1}{2}\left(\frac{N}{2} - 1\right)$, f_d – частота доплерівського зсуву.

При цьому справедливі наступні співвідношення:

$$E[x_I^2(t)] = E[y_Q^2(t)] = \frac{1}{2},$$

(2.12)

$$E[x_I(t)y_Q(t)] = 0.$$

Алгоритм (2.11) реалізовано підпрограмою fade.m, повний лістинг якої наводиться у Додатку Б.

Механізм дії релієвських завмирань можна розглядати у двох аспектах: по-перше, це змінна у часі поведінка каналу зв'язку, а по-друге, часове розсіювання (time spreading) або часова дисперсія.

Основною задачею цієї лабораторній роботи є дослідження категорій погіршення якості зв'язку через змінну у часі поведінку каналу або часову варіабельність каналу.

Дійсно, канали, що використовуються у мобільному зв'язку є нестационарними, оскільки рух передавача і приймача призводить в результаті до зміни шляху розповсюдження. Для переданого неперервного коливання цей ефект проявиться у зміні амплітуди і фази сигналу в приймачі. Якщо усі розсіюючі елементи, які складають канал, є стаціонарними, то при зупинці руху амплітуда і фаза отриманого сигналу будуть залишатися постійними, тобто канал стає начебто стаціонарним у часі. Як тільки рух поновиться, поведінка каналу стає знову змінною у часі.

Відповідно, для випадку мобільного зв'язку проявляється часова варіабельність каналу, яка визначається або безпосередньо рухом мобільного абоненту відносно передавача базової станції, або відносним рухом навколишнього середовища (наприклад, рух автомобілів відносно неподвижного абоненту), що призводить до постійної зміни трас розповсюдження.

Швидкість зміни рівня сигналу часто характеризують терміном доплерівське розсіювання, яке визначає селективний (змінний) у часі випадковий фазовий шум через випадкову частотну модуляцію.

Характеристика нестационарної природи сигналу може бути представлена в області доплерівського зсуву частот. На рис. 2.2а зображена доплерівська спектральна щільність потужності (або доплерівський спектр) $S(\nu)$ у вигляді функції від доплерівського зсуву частот ν [4].

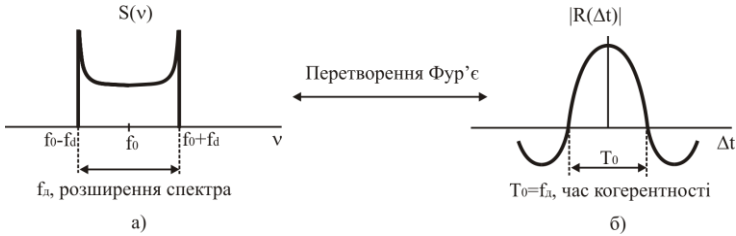


Рисунок 2.2 – Нестационарність каналу зв'язку у частотній (а) та часовій (б) областях

Для моделі з щільним розміщенням розсіюючих елементів, з вертикальною приймаючою антеною з постійним підсиленням, однорідним кутовим розподілом вхідного сигналу по усім кутам в інтервалі $(0, 2\pi)$ і немодульованим неперервним сигналом спектр сигналу в точці прийому можна подати як [4]:

$$S(\nu) = \frac{1}{\pi f_d \sqrt{1 - \left(\frac{\nu - f_0}{f_d}\right)^2}}.$$

Рівність зберігається для зсуву частот, що знаходиться в інтервалі $\pm f_d$ в околі несучої частоти f_0 , за межами цього інтервалу $S(\nu)$ рівняється нулю.

Загостреність і крутість спектра доплерівських частот є наслідком різкої верхньої межі доплерівського зсуву, який зумовлений переміщення рухомої антени серед стаціонарних розсіюючих елементів в моделі щільного розміщення.

Найбільша величина (нескінченість) $S(\nu)$ відповідає випадку, коли розсіюючий елемент знаходиться прямо перед рухомою платформою антени або позад неї. Для такого варіанту величина зсуву частот подається такою формулою:

$$f_d = \frac{V}{\lambda} = \frac{V}{c/f_0} = \frac{V \cdot f_0}{c} \quad (2.13)$$

де $c=3 \cdot 10^8$ м/с – швидкість розповсюдження електромагнітних хвиль, V – відносна швидкість руху мобільного об'єкту, f_0 - робоча частота.

Якщо передавач і приймач рухаються назустріч одне одному, то значення f_d , додатне, а якщо вони віддаляються одне від одного, то f_d має від'ємне значення. Якщо розсіючі елементи знаходяться у напрямку перпендикулярному руху антени, то для них величина частотного зсуву рівняється нулю.

Цей доплерівський спектр можна оцінити шляхом передачі гармонічного сигналу (вузькосмуговий сигнал) з наступним застосуванням Фур'є – аналізу для отримання спектра потужності прийнятої амплітуди.

Знання $S(v)$ надає можливість приблизної оцінки величини розширення спектра як функції швидкості зміни стану каналу. Ширина доплерівського спектра потужності f_d в технічній літературі може мати різні назви, зокрема: доплерівське розширення, швидкість завмирання, ширина смуги завмирання або спектральне розширення.

У звичайних умовах багатопроменового розповсюдження сигнал надходить за декількома перевідбитими шляхами, кожен з яких має відмінні від інших відстань і кут надходження. Доплерівський зсув для кожного з цих шляхів, як правило, різний. Тому вплив на прийнятий сигнал проявляється у вигляді саме розширення переданої частоти сигналу, а не як частотний зсув.

Тобто, доплерівське розсіювання визначається як ширина спектру прийнятого несучого коливання у випадку, коли через канал із багатопроменим розповсюдженням передається одне єдине несуче коливання на частоті f_0 . Якщо позначити максимальне значення частоти доплерівського зсуву f_d , то через доплерівське розсіювання приймається «розмазаний» спектр сигналу зі спектральними компонентами між частотами $f_0 - f_d$ і $f_0 + f_d$, тобто у смузі розсіювання f_d .

Оскільки характеристики каналу залежать від розташування передавача і приймача, то змінна у часі поведінка еквівалентна змінній просторовій поведінці. Відповідно часова варіабельність каналу зв'язку може бути розглянута і у часовій області.

Так, на рис. 2.2б зображена функція $R(\Delta t)$ яка визначає просторово-часову кореляційну функцію: це автокореляційна функція

прийнятого сигналу з переданою синусоїдою. Функція $R(\Delta t)$ визначає ступень кореляції між відкликом каналу на синусоїду, що передана у момент t_1 , з відкликом на таку ж синусоїду, яка передана у момент t_2 і, відповідно, $\Delta t = t_2 - t_1$.

За цією залежністю вводиться поняття **часу когерентності** T_0 – це міра очікуваного часу, за який характеристика каналу суттєво інваріантна. Функція $R(\Delta t)$ і параметр T_0 несуть у собі інформацію про швидкість завмирань у каналі. Для ідеального стаціонарного каналу (передавач і приймач абсолютно нерухомі) відклик каналу буде мати сильну кореляцію і, відповідно, $R(\Delta t)$ як функція від (Δt) буде мати постійне значення. Для визначеної моделі з щільним розташуванням розсіюючих елементів нормована $R(\Delta t)$ має такий вигляд [4, 5]:

$$R(\Delta t) = J_0(kV\Delta t),$$

де $J_0()$ – функція Бесселя першого роду нульового порядку, $V\Delta t$ – пройдений шлях, $k=2\pi/\lambda$ – фазова змінна для вільного простору.

Нестаціонарну природу, або механізм швидкості завмирань у каналі можна розглядати з позиції категорій погіршення якості передачі, що проявляються у вигляді швидких та повільних завмирань.

Термін “швидкі завмирання” використовуються для опису каналів, для яких час когерентності менший за тривалість переданого символу T_s , тобто $T_0 < T_s$. Швидкі завмирання описують умови, коли часовий інтервал, у межах якого поведінка каналу має явно виражений кореляційний характер, малий порівняно з часом необхідним для передачі символу. Відповідно характер завмирань в каналі буде змінюватися декілька разів за час передачі символу, що призведе до спотворення форми відеоімпульсу, а це, у свою чергу, сприяє частому з’явленню не виправимих помилок. Такі спотворені імпульси викликають проблеми синхронізації і також викликають певні проблеми для адекватної реалізації узгодженого прийому.

Якщо ж час когерентності більший за час передавання символу, тобто $T_0 > T_s$, то вважається, що канал вносить повільні завмирання. Для такої ситуації, можна очікувати, що стан каналу буде залишатися практично незмінним впродовж часу передавання символу. Основне

погіршення якості передачі в каналі з повільним завмиранням пов'язано зі зменшенням відношення сигнал/шум.

Необхідно відзначити, що час когерентності і доплерівське розширення зворотно пропорційні (з точністю до постійного множника) [4]:

$$T_0 = 1/f_{\dot{A}},$$

що і дозволяє розглядати доплерівське розширення як звичайну швидкість завмирання в каналі. На рис.2.2а T_0 визначався як інтервал часу впродовж якого відклик каналу суттєво інваріантний у часі.

Якщо визначати час когерентності більш точно, а саме, як інтервал часу впродовж якого коефіцієнт кореляції на менший за 0,5, то зв'язок між цими параметрами набуває вигляду [5]:

$$T_0 = 9/(16\pi f_d).$$

Проведений аналіз дозволяє визначити категорії погіршення характеристик внаслідок нестационарної природи каналу як в області доплерівського зсуву, так і у часовій області.

В каналі зв'язку присутні швидкі завмирання, якщо швидкість передачі символів $1/T_s$ (яку приблизно можна прийняти рівною смузі пропускання W) менша за швидкість завмирань $1/T_0$ (що приблизно рівняється доплерівському розширенню). Тобто, швидкі завмирання проявляються у каналі, якщо виконуються такі співвідношення:

$$W < f_{\dot{A}}, \quad (2.14)$$

або

$$T_s > T_0. \quad (2.15)$$

Навпаки, в каналі проявляються повільні завмирання, якщо швидкість передачі сигналів більша за швидкість завмирань. Тобто, щоб уникнути спотворень сигналу, до яких призводять швидкі завмирання, необхідно утворити канал, який підлягає впливу тільки

повільних завмирань, що і забезпечується за рахунок більшої швидкості передачі сигналу порівняно зі швидкістю завмирань [4]:

$$W > f_{\bar{\Delta}}, \quad (2.16)$$

або

$$T_s < T_0. \quad (2.17)$$

З рівняння (2.16) можна зробити висновок, що швидкість завмирань в каналі через доплерівське розширення встановлює нижню межу швидкості передачі сигналу, для якої відсутні спотворення, які властиві швидким завмиранням.

Але, на практиці виконання умов (2.16), (2.17) недостатньо для досягнення бажаних характеристик каналу. Кращим варіантом для усунення швидких завмирань є виконання умови $W \gg f_{\Delta}$ ($T_s \ll T_0$).

Якщо ця умова не виконується, то випадкова частотна модуляція, яка виникає через змінні доплерівські зсуви, буде суттєво погіршувати характеристики системи. Ефект Доплера призводить до частого з'явлення помилок, які не можливо компенсувати збільшенням відношення E_b/N_0 . Це особливо проявляється в системах передачі, які використовують фазову модуляцію.

Тому, на практиці для усунення спотворень, які зумовлені швидкими завмираннями і, як наслідок, частого з'явлення помилок, які неможливо виправити, швидкість передачі сигналу повинна перевищувати швидкість завмирань у 100...200 разів.

Інший механізм проявлення релеївських завмирань - це часова дисперсія сигналу. У середовищі з багатопроменевим розповсюдженням на антену приймача надходить не тільки пряма хвиля, а й хвилі, що надходять із затримкою, як результат перевідбиття, дифракції і розсіювання.

Численні перевідбиття можуть призводити до такої ситуації, коли один і той же сигнал приходить до приймача у різні моменти часу. Саме це явище і отримало назву часової дисперсії. Наявність різних часових затримок для відбитих сигналів зумовлює з'явлення міжсимвольної інтерференції (ефект накладання інформаційних бітів один на одне, яке приймач не спроможний відсортувати). Часова

дисперсія обумовлює частотно-селективний характер завмирань і може призводити до помітного погіршення якості зв'язку, а також виступає чинником, що обмежує швидкість передавання інформаційних символів у каналі.

Дослідження впливу частотно-селективних завмирань, які виникають через часову дисперсію, на характеристики каналу зв'язку буде проведено в лабораторній роботі №7.

У даній лабораторній роботі розглядається гіпотетичний канал, для якого частотно-селективні та швидкі завмирання відсутні, а проявляються тільки повільні завмирання, які зумовлені нестационарністю каналу.

Для дослідження впливу доплерівського розсіювання на характеристики каналу приймемо максимальну швидкість руху мобільного об'єкту рівною 120 км, а значення робочої частоти каналу 900 МГц. Згідно з (2.13) максимальне значення доплерівського зсуву рівняється: $f_d = V \cdot f/c = 120 \cdot 10^3 \cdot 900 \cdot 10^6 / 3600 \cdot 3 \cdot 10^8 = 100$ Гц.

Відповідно ширина доплерівського розсіювання або швидкість завмирань f_d рівняється подвійному значенню, тобто $f_d = 200$ Гц.

Для швидкості передачі інформації $R = 256$ кбіт/с, очевидно, що виконується умова $R \gg (100 \dots 200) f_d$. Як було визначено, таке співвідношення забезпечує відсутність швидких завмирань в каналі зв'язку і ефект доплерівського розсіювання проявляється як частотно-неселективні повільні завмирання.

2.2.2 Модель каналу зв'язку з використанням BPSK за наявності частотно-неселективних або повільних завмирань

Базовою програмою для моделювання каналу зв'язку з використанням BPSK за умови наявності завмирань у каналі зв'язку виступає програма `bpsk.m`, яка розглянута у ЛР №5.

Зміни у програмі підлягатимуть у наступному.

Після завдання вхідних тотожних даних для моделювання і ініціалізації параметрів формуючого фільтра \sqrt{RC} включається блок ініціалізації параметрів федингу [3].

Оскільки цей блок також передбачається використовувати і для моделювання частотно-селективних завмирань, то до нього включені

параметри, які характеризують як нестационарність каналу, так і часову дисперсію.

Для блоку ініціалізації федингу вводяться такі позначення:

$tstp$ – мінімальний крок моделювання, який визначається швидкістю передачі інформації і порядком дискретизації;

$i\tau$ – час надходження відповідної затриманої хвилі відносно прямої, нормалізований до кроку моделювання;

dvl – середня потужність для відповідної затриманої хвилі, що нормалізована до рівня прямої хвилі;

$th1$ – початкова фаза затриманої хвилі;

$n0$ – кількість хвиль достатня для проявлення доплерівського розсіювання;

$itnd0$ – значення індексу завмирань;

$itnd1$ – початкове значення лічильника індексу завмирань;

$pow1$ – сумарне значення кількості прямої та затриманих хвиль (кількість шляхів розповсюдження);

fd – частота доплерівського розсіювання;

$flat$ – змінна, яка визначає характер завмирань (якщо $flat=1$ моделюються частотно-неселективні завмирання, якщо $flat=0$ моделюються частотно-селективні завмирання).

Оскільки на лабораторну роботу №5 виноситься моделювання завмирань, які визначаються часовою варіабельністю каналу, то усі параметри, що відносяться до затриманих хвиль покладаються рівними нулю.

Тобто, розглядається тільки один шлях розповсюдження хвилі (за довжиною) і затримані у часі хвилі відсутні (відсутній ефект міжсимвольних спотворень). У такому разі завмирання будуть визначатися шириною доплерівського розсіювання.

Відповідно змінна $flat$ покладається рівній одиниці. При цьому вважається, що мінімальна кількість хвиль при якій проявляється ефект доплерівського розсіювання повинна бути не менш 6...7 [3]. Тому значення параметра $n0$ обираємо рівним 6.

Відповідний блок ініціалізації параметрів федингу приймає наступний вигляд:

```
%*****Fading initialization*****
```

```
 $tstp = 1/sr/IPOINT;$ 
```

```
 $i\tau = 0;$ 
```

```

dlvl =0;
th1=[0.0];
n0 =6;
itnd0=nd*IPOINT*100;
itnd1=[1000];
nowl =1;
fd =200;
flat=1;
%*****End of fading initialization*****

```

Після цього блоку процес моделювання сигналу у модуляторі подібний до алгоритму, якого розглянуто у програмі bpsk.m.

Зокрема генерація вхідних даних, перетворення вхідних даних до формату NRZ без постійної складової, виконання операції передискретизації, обробка сигналу у формуючому фільтрі Найквіста, повністю тотожні.

Введення доплерівського розсіювання і часової дисперсії сигналу реалізується підпрограмою sefade.m., до якої, як складові входять підпрограма формування доплерівського розсіювання fade.m і підпрограма формування затриманих хвиль delay.m.

Для умов, які розглядаються в лабораторній роботі – один шлях розповсюдження (за довжиною), наявність 6-7 хвиль, які приходять під різними кутами і відсутність затриманих хвиль, програма sefade.m практично транлює підпрограму fade.m до головної програми, яка забезпечує моделювання релєївського розсіювання для усіх випадкових послідовностей даних за алгоритмом, що визначається виразом (2.11).

Вхідними даними для підпрограми sefade.m виступають вихідний сигнал модулятора (**data3** у програмі і параметри, які введені у блоці ініціалізації федингу).

Окрім того, оскільки програми моделювання федингу необхідно розпоширити також і на методи багатопозиційної модуляції (QPSK, QAM), які реалізуються за допомогою I, Q модуляторів, то завмирання вводяться одночасно як в I, так і в Q – канали модулятора за однаковим алгоритмом. В термінах Matlab це реалізується наступною командою;

```

%*****Fading channel*****
[ifade,qfade]=sefade(data3,zeros(1,length(data3)),itau,...
dlvl,th1,n0,itnd1,nowl,length(data3),tstp,fd,flat);

```

Оскільки для методу BPSK використовується тільки один канал, то усі дані Q –каналу покладаються рівними нулю (оператор **zeros(1,length(data3))**).

Наступним кроком після введення замирань є оновлення значення лічильника (індексу) замирань:

```
% Update fading counter
```

```
itnd1=itnd1+itnd0;.
```

Введення цього параметру забезпечує різний характер замирань за релеївським законом для усіх хвиль, які надходять до приймача, Більш детально функцію цього оператора, а також роботу програми `sefade.m` буде розглянуто у лабораторній роботі №7.

Вплив на сигнал білого гаусівського шуму, а також визначення відношення сигнал/шум для заданого значення E_b/N_0 реалізується тотожно програмі `bpsk.m`.

В результаті вхідні дані у демодуляторі приймального тракту (в основній смузі частот) з урахуванням ефекту федингу і дії гаусівського шуму набувають вигляду:

```
%*****Add White Gaussian noise*****
```

```
inoise=randn(1,length(data3)).*attn;
```

```
data4=ifade+inoise;.
```

Далі сигнал підлягає обробці у формуючому фільтрі приймача з наступним вирішенням про прийнятий біт у пороговому пристрої подібно до чистого методу BPSK. Алгоритм визначення залежності BER від відношення E_b/N_0 не змінюється, за єдиною різницею - на графік додатково виводиться залежність BER, яка характеризує криву релеївського порогу, тобто, за умови проявлення тільки частотно-неселективних і повільних замирань.

Теоретичний вираз для ймовірності помилкового прийому одного біта або коефіцієнта бітової помилки (BER) для AWGN каналу за наявності частотно-неселективних і повільних замирань має такий вигляд:

$$BER_{BPSK-FADING} = \frac{1}{2} \left[1 - \frac{1}{\sqrt{1 + \frac{1}{E_b/N_0}}} \right]. \quad (2.18)$$

Результатом моделювання виступають графічні залежності для BER, які відповідають експерименту (позначення як 'o'), і теоретичному виразу (2.18) (позначення як '*'), а також залежності BER для AWGN каналу (позначення як '--'), що дозволяє оцінити адекватність моделі каналу зв'язку з використанням методу BPSK і проявленням ефекту часової варіабельності каналу основним теоретичним положенням.

2.3 Експериментальна частина

2.3.1 Відкрити середовище Matlab. Створити нову директорію (папку) LAB6, в яку необхідно розмістити основну програму, а також усі підпрограми, що необхідні для моделювання.

2.3.2 Скопіювати файл основної програми `bpsk_fading.m`, який знаходиться у додатку Б методичних вказівок. Вставити скопійований файл `bpsk_fading` до редактора Matlab. Зберегти скопійований файл під ім'ям `bpsk_fading.m` у створеній директорії LAB6.

2.3.3 Скопіювати у додатку А підпрограму `hrollcoef.m`, і повторити процедуру створення нового М-файлу в директорії LAB6.

2.3.4 Скопіювати у додатку А підпрограму `oversamp.m`, що виконує операцію передискретизації, і повторити процедуру створення нового М-файлу в директорії LAB6.

2.3.5 Скопіювати у додатку Б підпрограму `fade.m`, і повторити процедуру створення нового М-файлу в директорії LAB6. Файл зберегти під ім'ям, що повторює назву функції -- `fade`.

2.3.6 Скопіювати у додатку Б підпрограму `delay.m`, і повторити процедуру створення нового М-файлу в директорії LAB6. Файл зберегти під ім'ям, що повторює назву функції -- `delay`.

2.3.7 Скопіювати у додатку Б підпрограму `sefade.m`, і повторити процедуру створення нового М-файлу в директорії LAB6. Файл зберегти під ім'ям, що повторює назву функції -- `sefade`.

Після виконання вище визначених пунктів в основному вікні середовища Matlab усі 6 М-файлів (`bpsk_fading.m`, `hrollcoef.m`, `oversamp.m`, `fade.m`, `sefade.m`, `delay.m`) повинні відобразитися у полі поточної директорії (Current Directory).

2.3.8 У командній строчці вікна Command Window набрати назву основної програми `bpsk_fading` і запустити аналіз натисканням клавіші Enter.

Якщо синтаксичні помилки у програмах відсутні, то на дисплей у командному вікні повинні відобразитися значення BER для кожної реалізації випадкової послідовності відповідно до заданого циклу `ploop` і заданого значення E_b/N_0 .

2.3.9 Отримати графічну залежність ймовірності помилкового прийому біта (або коефіцієнта BER) від відношення E_b/N_0 . Графічну залежність подати у напівлогарифмічному масштабі з діапазоном зміни ймовірності від 10^{-5} до 1. Діапазон зміни відношення E_b/N_0 задати від 0 до 35 дБ.

2.3.10 Порівняти залежності отримані за результатами моделювання і теоретичним виразом (2.17), а також провести порівняння з аналогічною характеристикою для AWGN каналу для значень $BER=10^{-3}$ і $BER=10^{-4}$.

2.4 Зміст звіту

2.4.1 Структурні схеми модуляторів і демодуляторів, що досліджуються. Блок-схема алгоритму програми.

2.4.2 Графічні залежності $BER=f(E_b/N_0)$ за результатами моделювання та теоретичним виразом.

2.4.3 Лістинги програм.

2.4.3 Висновки за результатами роботи.

2.5 Контрольні запитання

1. Поняття нестационарності каналу зв'язку.
2. Пояснити сенс терміна доплерівське розширення.
3. Визначити максимальне значення доплерівського зсуву при швидкості руху 90 км/год.
4. Як залежить доплерівський зсув від напрямку руху абонента?
5. Поняття часу когерентності.
6. Зв'язок між часом когерентності і доплерівським розширенням.

7. Як визначається час когерентності за умови, що коефіцієнт кореляції на менший за 0,5,
8. Умови виникнення швидких завмирань.
9. Умови виникнення повільних завмирань.
10. Теоретичний вираз для залежності BER від відношення E_b/N_0 для AWGN каналу.
11. Теоретичний вираз для залежності BER від відношення E_b/N_0 для каналу з частотно-неселективними завмираннями.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Кабак В.С., Уваров Р.В. Функціональні пристрої телефонів мобільного зв'язку: Навчальний посібник. [Текст] / В.С. Кабак, Р.В. Уваров – Запоріжжя, 2007. – 375 с.
2. Феер К. Беспроводная цифровая связь. [Текст] / К. Феер; Пер. с англ.; – М.: Радио и связь, 2000. – 519 с.
3. Harada Н. Simulation and software radio for mobile telecommunications [Текст] / Н. Harada, R. Prasad; - Artech House, 2003. – 465 p.
4. Скляр Б. Цифровая связь. Теоретические основы и практическое применение [Текст] / Б. Скляр; Пер. с англ.; – М.: Издательский дом “Вильямс”, 2003. – 1104 с.
5. Ипатов В.П. Системы мобильной связи: Учебное пособие для вузов. [Текст] / В.П. Ипатов, В.К. Орлов, И.М. Самойлов, В.Н. Смирнов; под.ред. В.П. Ипатова. – М.: Горячая линия-Телеком, 2003. – 272 с.

ДОДАТОК А

Лістинги програм для моделювання методу BPSK для AWGN каналу

```

%bpsk.m
%*****Preparation part*****
sr=256000.0; % Symbol rate
ml=1;
br=sr.*ml;
nd=100;
ebn0=5;
IPOINT=8;
%*****Filter initialization*****
irfn=21;
alfs=0.5;
[xh]=hrollfcoef(irfn, IPOINT, sr, alfs, 1); % Transmission filter coefficients
[xh2]=hrollfcoef(irfn, IPOINT, sr, alfs,0); % Receiver filter coefficients
%***** Start calculations*****
nloop=100;
noe=0;
nod=0;

for iii=1:nloop
    %***** Data generation*****
    data=rand(1,nd*ml)>0.5;
    %***** BPSK Modulation*****
    data1=2.*data-1;
    [data2]=oversamp(data1, nd, IPOINT);
    data3=conv(data2, xh);
    %***** Attenuation Calculations*****
    spow=sum(data3.*data3)/nd;
    attn=0.5*spow*sr/br*10.^(-ebn0/10);
    attn=sqrt(attn);
    %***** Add White Gaussian Noise*****
    inoise=randn(1,length(data3)).*attn;
    data4=data3+inoise;
    data5=conv(data4,xh2);
    sampl=irfn*IPOINT+1;
    data6=data5(sampl:8*nd+sampl-1);
    %***** BPSK Demodulation*****
    [demodata]=data6>0;

```

```

% *****Bit Error Rate*****
noe2=sum(abs(data-demodata));
nod2=length(data);
noe=noe+noe2;
nod=nod+nod2;
fprintf('%d\t%e\n', iii, noe2/nod2);
end
% *****Ounput Results*****
ber=noe/nod;
fprintf('%d\t%d\t%d\t%e\n', ebn0, noe, nod, noe/nod);
fid=fopen('BERbpsk.dat','a');
fprintf(fid,'%d\t%e\t%f\t%e\n', ebn0, noe/nod, noe, noe);
fclose(fid);
% *****End of file*****
figure; stem(0:20,data(1:21),'DisplayName','data');
figure; stem(0:20,data1(1:21),'DisplayName','data');
figure; stem(0:20*IPOINT,data2(1:20*IPOINT+1),'DisplayName','data2');
figure; stem(0:20*IPOINT,data3(1:20*IPOINT+1),'DisplayName','data3');
figure; plot(0:20*IPOINT,data3(1:20*IPOINT+1),'DisplayName','data3');
figure; stem(0:20*IPOINT,data4(1:20*IPOINT+1),'DisplayName','data4');
figure; plot(0:20*IPOINT,data4(1:20*IPOINT+1),'DisplayName','data4');
figure; stem(0:40*IPOINT,data5(1:40*IPOINT+1),'DisplayName','data5');
figure; plot(0:40*IPOINT,data5(1:40*IPOINT+1),'DisplayName','data5');
figure; stem(0:20,data6(1:21),'DisplayName','data6');
figure; stem(0:20,demodata(1:21),'DisplayName','demodata');

% *****End *****

function [xh]=hrollfcoef(irfn,ipoint,sr,alfs,ncc)
% irfn: Number of symbols to use filtering
% ipoint: Number of samples in one symbol
% sr : symbol rate
% alfs: rolloff coefficients
% 1 - transmission filter 0 - receiving filter
xi=zeros(1,irfn*ipoint+1);
xq=zeros(1,irfn*ipoint+1);
point=ipoint;
tr=sr;
tstp=1.0./tr./ipoint;
n=ipoint .*irfn;

```

```

mid=(n./2)+1;
sub1=4.0*alfs.*tr; % 4* alpha*R_s
for i=1:n
    icon=i-mid;
    ym =icon;
    if icon==0.0
        xt=(1.0-alfs+4.0.*alfs/pi).*tr; % h(0)
    else
        sub2=16.0.*alfs.*alfs.*ym.*ym./ipoint./ipoint;
        if sub2~=1.0
            x1=sin(pi*(1.0-alfs)/ipoint*ym)./pi./(1.0-sub2)./ym./tstp;
            x2=cos(pi*(1.0+alfs)/ipoint*ym)./pi.*sub1./(1.0-sub2);
            xt=x1+x2;
        else
            xt=alfs.*tr.*((1.0-2.0/pi).*cos...
                (pi/4.0/alfs)+(1.0+2.0./pi).*sin...
                (pi/4.0/alfs))./sqrt(2.0);
        end % if sub2~=1.0
    end % if icon==0
    if ncc==0 % in case of receiver
        xh(i)=xt ./ipoint ./tr; % normalization
    else if ncc==1 % in the case of transmitter
        xh(i)=xt ./tr; % normalization
    else
        error('ncc error');
    end % if ncc==0
end % for i=1:n
end
% *****End of file*****

function [ out ] = oversamp ( indata, nsymb, sample )
% insert zero data to input data
% *****Variables*****
% indata : input sequences
% nsymb : number of symbols
% sample : number of oversamples
% *****
out=zeros(1,nsymb*sample);
out(1:sample:1+sample*(nsymb-1))=indata;
end
% *****End of file*****

```

```

%bpsk_ber.m
%*****Preparation part*****
sr=256000.0; % Symbol rate
ml=1;
br=sr.*ml;
nd=100;
ebn0=10;
IPOINT=8;
%*****Filter inialization*****
irfn=21;
alfs=0.5;
[xh]=hrollfcoef(irfn, IPOINT, sr, alfs, 1); % Transmission filter coefficients
[xh2]=hrollfcoef(irfn, IPOINT, sr, alfs,0); % Receiver filter coefficients
%***** Start calculations*****
nloop=1000;
noe=0;
nod=0;
ebn0_array=0:1:20;
ber_array=zeros(length(ebn0_array),1);
ber_array_theory=zeros(length(ebn0_array),1);
for ebn0_loop=1:length(ebn0_array)
    ebn0=ebn0_array(ebn0_loop);

for iii=1:nloop
    % ***** Data generation*****
    data=rand(1,nd*ml)>0.5;
    % ***** BPSK Modulation*****
    data1=2.*data-1;
    [data2]=oversamp(data1, nd, IPOINT);
    data3=conv(data2, xh);
    % ***** Attenuation Calculations*****
    spow=sum(data3.*data3)/nd;
    attn=0.5*spow*sr/br*10.^(-ebn0/10);
    attn=sqrt(attn);
    % ***** Add White Gaussian Noise*****
    inoise=randn(1,length(data3)).*attn;
    data4=data3+inoise;
    data5=conv(data4,xh2);
    sampl=irfn*IPOINT+1;
    data6=data5(sampl:8*nd+sampl-1);
    % *****BPSK Demodulation*****

```

```

[demodata]=data6>0;

% *****Bit Error Rate*****
noe2=sum(abs(data-demodata));
nod2=length(data);
noe=noe+noe2;
nod=nod+nod2;
fprintf('%d\t%e\n', iii, noe2/nod2);
end
% *****Ounput Results*****
ber=noe/nod;
noe=0; nod=0;
ber_array(ebn0_loop)=ber;
ber_array_theory(ebn0_loop)=0.5*erfc(sqrt(10^(ebn0_array(ebn0_loop)/10)));
end

figure;
h=semilogy(ebn0_array,ber_array,'o',ebn0_array,ber_array_theory,'-');
set(h,{'DisplayName','DisplayName'},{'Experiment','Theory'})
legend show

fprintf('Eb/N0=%d\t BER=%e\t NoE=%d\t NoD=%e\n', ebn0, ber, noe, nod);
fid=fopen('BERbpsk.dat','a');
fprintf(fid,'Eb/N0=%d\t BER=%e\t NoE=%f\t NoD=%d\t\n', ebn0, ber, noe, nod);
fclose(fid);
% *****End of file*****

```

ДОДАТОК Б

Лістинги програм для моделювання методу BPSK для каналу з релієвськими завмираннями

```
%bpsk_fading.m
%*****Preparation part*****
sr=256000.0; % Symbol rate
ml=1;
br=sr.*ml;
nd=100;
ebn0=35;
IPOINT=8;
%*****Filter inialization*****
irfn=21;
alfs=0.5;
[xh]=hrollfcoef(irfn, IPOINT, sr, alfs, 1);
[xh2]=hrollfcoef(irfn, IPOINT, sr, alfs,0);% Receiver filter coefficients
%*****Fading initialization*****
% time resolution
tstp=1/sr/IPOINT;
% set only direct wave
itau=[0];
dlvl=[0];
% number of waves
n0=[6];
% initial phase of delayed waves
th1=[0.0];
% number of fading counter to skip
itnd0=nd*IPOINT*100;
% one fading counter is needed
itnd1=[1000];
% number of direct waves+ number of delayed waves
nowl=1;
% maximum Doppler frequency
fd=200;
% flat fading
flat=1;
%***** Start calculations*****
nloop=1000;
```

```

noe=0;
nod=0;

ebn0_array=0:1:35;
ber_array=zeros(length(ebn0_array),1);
ber_array_theory=zeros(length(ebn0_array),1);

for ebn0_loop=1:length(ebn0_array)
    ebn0=ebn0_array(ebn0_loop);

for iii=1:nloop
    % ***** Data generation*****
    data=rand(1,nd*ml)>0.5;
    % ***** BPSK Modulation*****
    data1=2.*data-1;
    [data2]=oversamp(data1,nd,IPOINT);
    data3=conv(data2,xh);
    % *****Attenuation Calculations*****
    spow=sum(data3.*data3)/nd;
    attn=0.5*spow*sr/br*10.^(-ebn0/10);
    attn=sqrt(attn);
    % *****Fading channel*****
    [ifade,qfade]=sefade(data3,zeros(1,length(data3)),itau,...
    dlvl,th1,n0,itnd1,nowl,length(data3),tstp,fd,flat);
    % Update fading counter
    itnd1=itnd1+itnd0;
    % *****Add white Gaussian noise*****
    inoise=randn(1,length(data3)).*attn;
    data4=ifade+inoise;
    data5=conv(data4,xh2);
    sampl=irfn*IPOINT+1;
    data6=data5(sampl:8:8*nd+sampl-1);

    % *****BPSK Demodulation*****
    [demodata]=data6>0;
    % *****Bit Error Rate*****
    noe2=sum(abs(data-demodata));
    nod2=length(data);
    noe=noe+noe2;
    nod=nod+nod2;
    fprintf('%d\t%e\n', iii, noe2/nod2);

```

end

```
% *****Output Results*****
```

```
ber=noe/nod;
```

```
noe=0;
```

```
nod=0;
```

```
ber_array(ebn0_loop)=ber;
```

```
ber_array_theory(ebn0_loop)=0.5*erfc(sqrt(10^(ebn0_array(ebn0_loop)/10)));
```

```
ber_array_theory1(ebn0_loop)=0.5*(1-1./(sqrt(1+1/(10^(ebn0_array...  
(ebn0_loop)/10)))));
```

end

```
fprintf('Eb/N0=%d\t BER=%e\t NoE=%d\t NoD=%e\n', ebn0, ber, noe, nod);
```

```
fid=fopen('BERbpsk.dat','a');
```

```
fprintf(fid,'Eb/N0=%d\t BER=%e\t NoE=%f\t NoD=%d\t\n', ebn0, ber, noe, nod);
```

```
fclose(fid);
```

```
figure;
```

```
h=semilogy(ebn0_array,ber_array,'o',ebn0_array,ber_array_theory,'--
```

```
',ebn0_array,ber_array_theory1,'*');
```

```
set(h,{'DisplayName'},{'Experiment','Theory','Theory1'})
```

```
legend show
```

```
% *****End of file*****
```

```
% sefade.m
```

```
% This function generates frequency selecting fading...
```

```
function[iout,qout,ramp,rcos,rsin]=sefade(idata,qdata,itau,dlvl,th,n0,itn,n1,nsamp,...  
tstp,fd,flat)
```

```
% ***** variables *****
```

```
% idata input Ich data
```

```
% qdata input Qch data
```

```
% iout output Ich data
```

```
% qout output Qch data
```

```
% ramp : Amplitude contaminated by fading
```

```
% rcos : Cosine value contaminated by fading
```

```
% rsin : Cosine value contaminated by fading
```

```
% General Definition of Simulation Tools $# % itau : Delay time for each  
multipath fading
```

```
% dlvl : Attenuation level for each multipath fading
```

```

% th : Initialized phase for each multipath fading
% n0 : Number of waves in order to generate each
% multipath fading
% itn : Fading counter for each multipath fading
% n1 : Number of summation for direct and delayed
% waves
% nsamp : Total number of symbols
% tstp : Minimum time resolution
% fd : Maximum doppler frequency
% flat flat fading or not
% (1-flat (only amplitude is fluctuated),0-normal(phase
% and amplitude are fluctuated))
% *****
iout = zeros(1,nsamp);
qout = zeros(1,nsamp);
total_attn = sum(10.^(-1.0.* dlvl./ 10.0));
for k=1:n1
atts = 10.^(-0.05.* dlvl(k));
if dlvl(k) == 40.0
atts = 0.0;
end
theta = th(k).* pi./180.0;
[itmp,qtmp] = delay (idata,qdata,nsamp,िताu(k));
[itmp3,qtmp3,ramp,rcos,rsin] = fade (itmp,qtmp,...
nsamp,tstp,fd,n0(k),itn(k),flat);
iout = iout + atts .* itmp3 ./ sqrt(total_attn);
qout = qout + atts .* qtmp3 ./ sqrt(total_attn);
end
% ***** end of file *****

```

```

% delay.m
% Gives delay to input signal%
function [iout,qout] = delay(idata,qdata,nsamp,idel )
% ***** variables *****
% idata input Ich data
% qdata input Qch data
% iout output Ich data
% qout output Qch data
% nsamp Number of samples to be simulated
% idel Number of samples to be delayed

```

```

% *****
iout=zeros(1,nsamp);
qout=zeros(1,nsamp);
if idel ~= 0
iout(1:idel) = zeros(1,idel);
qout(1:idel) = zeros(1,idel);
end
iout(idel+1:nsamp) = idata(1:nsamp-idel);
qout(idel+1:nsamp) = qdata(1:nsamp-idel);
% ***** end of file *****

```

```

% fade.m
% Generate Rayleigh fading

```

```

function [iout,qout,ramp,rcos,rsin]=fade(idata,qdata,...
nsamp,tstp,fd,no,counter,flat)

```

```

% ***** variables *****
% idata : input Ich data
% qdata : input Qch data
% iout : output Ich data
% qout : output Qch data
% ramp : Amplitude contaminated by fading
% rcos : Cosine value contaminated by fading
% rsin : Cosine value contaminated by fading
% nsamp : Number of samples to be simulated
% tstp : Minimum time resolution
% fd : maximum doppler frequency
% no : number of waves in order to generate fading
% counter : fading counter
% flat : flat fading or not
% (1-flat (only amplitude is fluctuated),0-normal
% (phase and amplitude are fluctuated))

```

```

% *****
if fd ~= 0.0
ac0 = sqrt(1.0 ./ (2.0.*(no + 1)));
% power normalized constant(ich)
as0 = sqrt(1.0 ./ (2.0.*no));

```

```

% power normalized constant(qch)
ic0 = counter;
% fading counter
pai = 3.14159265;
wm = 2.0.*pai.*fd;
n = 4.*no + 2;
ts = tstp;
wmts = wm.*ts;
paino = pai./no;
xc=zeros(1,nsamp);
xs=zeros(1,nsamp);
ic=[1:nsamp]+ic0;
for nn=1:no
cwn = cos( cos(2.0.*pai.*nn./n).*ic.*wmts );
xc = xc + cos(paino.*nn).*cwn;
xs = xs + sin(paino.*nn).*cwn;
end
cwmt = sqrt(2.0).*cos(ic.*wmts);
xc = (2.0.*xc + cwmt).*ac0;
xs = 2.0.*xs.*as0;
ramp=sqrt(xc.^2+xs.^2);
rcos=xc./ramp;
rsin=xs./ramp;
if flat ==1
iout = sqrt(xc.^2+xs.^2).*idata(1:nsamp);
% output signal(ich)
qout = sqrt(xc.^2+xs.^2).*qdata(1:nsamp);
% output signal(qch)
else
iout = xc.*idata(1:nsamp) - xs.*qdata(1:nsamp);
% output signal(ich)
qout = xs.*idata(1:nsamp) + xc.*qdata(1:nsamp);
% output signal(qch)
end
else
iout=idata;
qout=qdata;
end
% ***** end of file *****

```