

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-22.06- 05.01/172.00.1/Б/ОК27- 1-2025
	<i>Випуск 1</i>	<i>Зміни 0</i>	<i>Екземпляр № 1</i>	<i>Арк 127 / 1</i>

ЗАТВЕРДЖЕНО

науково-методичною радою
 Державного університету
 Житомирська політехніка
 протокол від «12» червня 2025 р.
 №4

МЕТОДИЧНІ ВКАЗІВКИ до лабораторних занять з навчальної дисципліни **«ПРИЙМАННЯ ТА ОБРОБЛЕННЯ** **СИГНАЛІВ І ЗОБРАЖЕНЬ»**

для студентів освітнього рівня «бакалавр»
 спеціальності 172 «Електронні комунікації та радіотехніка»
 освітньо-професійних програм «Телекомунікації та радіотехніка»,
 «Інформаційні відеосистеми та системи контролю доступу»
 факультет інформаційно-комп'ютерних технологій
 кафедра біомедичної інженерії та телекомунікацій

Розглянуто і рекомендовано
 на засіданні кафедри
 біомедичної інженерії та
 телекомунікацій
 протокол від «14» квітня 2025 р.
 №3

Розробники: к.т.н., доцент Чухов В. В., к.т.н., доцент Горшенін О. Є.,
 ст. викладач Андрусенко Є. М., ст. викладач Корніюк А.В.

Житомир
 2025 р.

Чухов В. В. Методичні вказівки до лабораторних занять з навчальної дисципліни «Приймання та оброблення сигналів і зображень» для студентів освітнього ступеня «бакалавр» спеціальності 172 «Електронні комунікації та радіотехніка» освітньо-професійних програм «Телекомунікації та радіотехніка», «Інформаційні відеосистеми та системи контролю доступу» / В. В. Чухов., О. Є. Горшенін, Є. М. Андрусенко, А. В. Корніюк – Житомир: Житомирська політехніка, 2025.– 127 с.

Зміст

Вступ.....	4
Лабораторна робота №1. Апаратура для імітації радіосигналів.....	5
Лабораторна робота №2. Вхідні кола приймачів.....	9
Лабораторна робота №3. Шумові властивості та чутливість приймального пристрою.....	20
Лабораторна робота №4. Амплітудний детектор у режимах малого та великого сигналів.....	25
Лабораторна робота №5. Чутливість приймального пристрою.....	28
Лабораторна робота №6. Розподіл діапазону робочих частот на піддіапазони.....	32
Лабораторна робота №7. Частотні фільтри у приймальних пристроях.....	38
Лабораторна робота №8. Дослідження перетворювача частоти.....	45
Лабораторна робота №9. Колірні простори RGB та HSV.....	48
Лабораторна робота №10. Вивчення основних параметрів діодного екрана.....	77
Лабораторна робота №11. Дослідження основних параметрів відеосигналу в системі обробки та відтворення зображень.....	99
Лабораторна робота №12. Знайомство з виробничими процесами пайки друкованих плат.....	113
Література.....	127

Вступ

Метою даних методичних вказівок є практичне вивчення основ фізичних процесів, теорії та принципів побудови і функціонування пристроїв приймання та оброблення сигналів, використовуваних у різних телекомунікаційних і радіотехнічних системах.

До початку лабораторної роботи кожен студент повинен ознайомитись з методичними вказівками даної лабораторної роботи та отримати допуск до її виконання, продемонструвавши викладачу чітке розуміння ходу виконання лабораторної роботи та здавши необхідний мінімум теоретичних знань за темою цієї роботи.

Перед включенням вимірювальної схеми студенти на робочих місцях повинні продемонструвати викладачу зібрану вимірювальну схему, яку той повинен перевірити. Якщо вимірювальну схему зібрано правильно, то викладач дає дозвіл на її включення до та проведення вимірювань.

Звіт з виконаної лабораторної роботи оформлюється кожним студентом окремо на папері формату А4, з дотриманням вимог оформлення такої документації. Після оформлення звіту він захищається студентом у встановленому порядку, після чого такий звіт здається на кафедру.

Автори будуть вдячні за пропозиції щодо покращення змісту цього видання.

Лабораторна робота №1
Апаратура для імітації радіосигналів

Мета заняття:

1. Закріплення знань про математичні моделі типових радіосигналів.
2. Набуття навичок з підбору та роботи з апаратурою для імітації радіосигналів.

Основні питання заняття:

1. Міжнародна система позначення радіовипромінювань.
2. Апаратура для імітації радіосигналів.

Хід заняття

1. Опрацювати лекційний матеріал за темою заняття.
2. Розв'яжіть задачі, які наведено нижче.

Задача 1. Використовуючи систему позначення радіовипромінювань, записати вибраний вами варіант (таблиця 1.1), без додаткової характеристики.

Кожен вибирає лише один варіант.

Таблиця 1.1

<i>Номер варіант</i>	<i>Діапазон частот сигналу МГц</i>	<i>Вид модуляції</i>	<i>Характер модуючого повідомлення</i>	<i>Вид повідомлення</i>
<i>a</i>	<i>2</i>	<i>3</i>	<i>4</i>	<i>5</i>
1	0,52...1,6	DSBFC	Один канал з аналоговою інформацією	Звукове радіомовлення
2	0,15...0,40	VSB	Один канал з аналоговою інформацією	Звукове радіомовлення
3	0,3...1,10	ISBFC	Телебачення	Звукове радіомовлення
4	1,9...1,95	SSBSC	Один канал з аналоговою інформацією	Звукове радіомовлення

5	144...146	SSBRC	Один канал з аналоговою інформацією	Звукове радіомовлення
6	1400...1450	Відсутня	Повідомлення не передається	Відсутнє
7	110...150	FM	Телебачення	Звукове радіомовлення
8	21...24	PM	Два і більше каналів з аналоговою інформацією	Передача даних
9	110...150	FM	Телебачення	Звукове радіомовлення
10	21...24	VSB	Один канал з аналоговою інформацією	Звукове радіомовлення
11	0,8...1,6	PAM	Один канал з аналоговою інформацією	Передача даних
12	8...15	DSBFC	Один канал з аналоговою інформацією	Звукове радіомовлення
13	0,8...1,6	PWM	Один канал з аналоговою інформацією	Передача даних
14	0,8...1,6	PPM	Один канал з аналоговою інформацією	Передача даних
15	21...24	DSBFC	Один канал з аналоговою інформацією	Звукове радіомовлення

Задача 2. Використовуючи таблицю 1.2, розрахуйте ширину спектра вибраного вами сигналу.

Кожен вибирає лише один варіант.

Таблиця 1.2

Номер варіанта	Діапазон частот сигналу МГц	Вид модуляції
-------------------	--------------------------------	---------------

1	2	3
1	0,52...1,6	DSBFC
2	0,15...0,40	VSБ
3	0,3...1,10	ISBFC
4	1,9...1,95	SSBSC
5	144...146	SSBRC
6	1400...1450	VSБ
7	21...24	VSБ
8	8...15	DSBFC
9	21...24	DSBFC

Задача 3. Що обмежує використання ЧМ у діапазонах кілометрових та гектометрових хвиль? Скільки радіомовних станцій з ЧМ можна розташувати у діапазонах СХ та УКХ, якщо на один радіоканал відвести смугу у 250 кГц?

Задача 4. Відповідними нормативними документами встановлено вимоги до апаратури імітації радіосигналів, де, серед іншого, рекомендовано використовувати такі типи генераторів:

1) генератор сигналів низькочастотний з неперервним генеруванням синусоїдальної напруги;

2) генератор сигналів високочастотний з нерервним генеруванням синусоїдальної напруги, внутрішньою амплітудною, частотною та імпульсною модуляцією синусоїдальною напругою, а також регульованим затуханням вихідної напруги для формування стандартних вхідних рівнів сигналу радіоприймача.

3) генератор шумових сигналів з неперервним генеруванням процесу виду «білий шум» та відхиленням частотної характеристики спектральної густини потужності ± 1 дБ. Миттєві значення напруги на виході генератора мають бути розподілені за нормальним законом.

Використовуючи вибраний вами варіант сигналу з таблиці 1.1, підберіть під його приймач імітаційні генератори з таким доповненням: тип 1 – амплітуда коливання 1 В; тип 2 – відхилення затухання вихідної напруги ± 1 дБ.

Інтернет-ресурси

1. Техенком. Вимірювальні прилади та обладнання – Режим доступу: <https://www.tehencom.com>
2. Masteram, магазин інструментів – Режим доступу: <https://masteram.com.ua>
3. Gtest. Магазин вимірювальної техніки та електроніки – Режим доступу: <https://gtest.com.ua>
4. «Виміряємо все» – магазин професійного вимірювального, діагностичного та метрологічного обладнання – Режим доступу: <http://www.izmerimvse.ua/>

Лабораторна робота №2

Вхідні кола приймачів

Мета заняття:

1. Навчитись здійснювати кількісний розрахунок селективності вхідних кіл.
2. Навчитись визначати значення коефіцієнтів зв'язку для забезпечення потрібного узгодження контуру з антеною та навантаженням.
3. Навчитись налаштовувати та регулювати вхідні кола.

Основні питання заняття:

1. Параметри коливальних кіл.
2. Приклади розрахунку вхідних кіл приймачів.

Короткі теоретичні відомості

Резонансні контури поділяють на послідовні (рисунк 3.1, а) та паралельні (рисунк 3.1, б). Опір r_k є еквівалентом втрат у реактивних елементах на робочій частоті. При цьому вважають, що втрати у котушках індуктивності L_k та конденсаторах C_k відсутні. Джерело енергії є ідеальним джерелом з синусоїдальною формою напруги e_k (струму i_k).

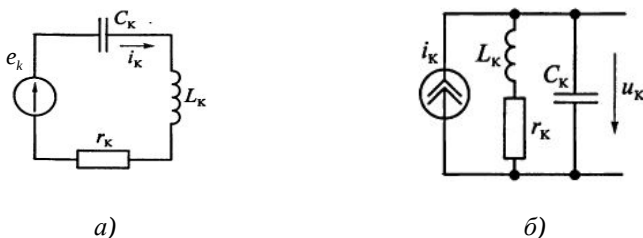


Рисунок 3.1 – Види коливальних контурів

Добротність (якість) резонансного контуру $Q_k = 2\pi W_{\text{зан}} / W_{\text{спож}}$ – це відношення енергії, запасеної у контурі протягом одного періоду

$W_{\text{зан}}$, до енергії, спожитої від джерела $W_{\text{спож}}$ за цей же період синусоїдальної напруги. Енергія $W_{\text{зан}} = I_m^2 L_k / 2$, яку запасено у коливальному контурі на резонансній частоті $f_0 = 1 / (2\pi \sqrt{L_k C_k})$, визначається амплітудою синусоїдального струму I_m . Енергія, яку спожито опором r_k протягом одного періоду, така:

$$W_{\text{спож}} = I_m^2 r_k / (2 f_0). \text{ звідси}$$

$$Q_k = \frac{2\pi f_0 L_k}{r_k} = \frac{1}{2\pi f_0 C_k r_k} = \frac{1}{r_k} \sqrt{\frac{L_k}{C_k}} = \frac{\rho_k}{r_k},$$

де $\rho_k = 2\pi f_0 L_k = 1 / (2\pi f_0 C_k) = \sqrt{L_k / C_k}$ – характеристичний опір коливального контуру.

Для паралельного контуру втрати зручно відображати паралельно включеною еквівалентною провідністю G_k (рисунок 3.2).

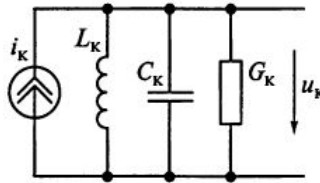


Рисунок 3.2 – Спосіб врахування втрат у паралельному коливальному контурі

У цьому випадку добротність контуру

$$Q_k = \frac{1}{\rho_k G_k} = \frac{1}{2\pi f_0 L_k G_k} = \frac{2\pi f_0 C_k}{G_k},$$

а його затухання $d_k = 1 / Q_k$.

Для описання властивостей контуру використовують параметри:

абсолютна розстройка частоти джерела вхідного гармонічного сигналу f відносно резонансної частоти f_0 :

$$\Delta f = |f - f_0| \text{ або } \Delta \omega = |\omega - \omega_0|,$$

відносна розстройка частоти джерела вхідного гармонічного сигналу f відносно резонансної частоти f_0 :

$$\nu = \frac{\omega}{\omega_0} - \frac{\omega_0}{\omega} = \frac{f}{f_0} - \frac{f_0}{f},$$

узагальнена розстройка частоти джерела вхідного гармонічного сигналу f відносно резонансної частоти f_0 :

$$\xi = \frac{1}{r_k} \left(2\pi f L_k - \frac{1}{2\pi f C_k} \right) = Q_k \left(\frac{\omega}{\omega_0} - \frac{\omega_0}{\omega} \right) = Q_k \nu.$$

На рисунку 3.3 показано залежність напруги на коливальному контурі u_k від колової частоти $\omega = 2\pi f$ вхідних коливачів струму з однаковими амплітудами I_m при фіксованих значеннях індуктивності L_k та ємності C_k , але двох різних еквівалентних провідностей G_k контуру. Як бачимо, чим менша провідність чи опір втрат контуру, тим більша амплітуда та гостріший резонанс.

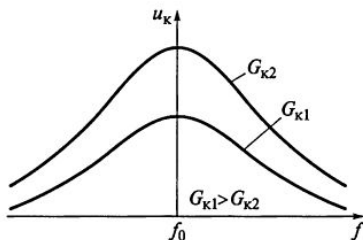


Рисунок 3.3 – Вплив провідності навантаження контуру на амплітуду коливачів у ньому

Розглянемо деякі приклади.

Приклад 1. Яке послаблення створює резонансний контур, власна резонансна частота якого $f_0 = 1$ МГц, добротність $Q_k = 50$ при абсолютній розстройці $\Delta f = 20$ кГц?

Розв'язок. Послаблення контуру визначається узагальненою розстройкою частоти джерела вхідного гармонічного сигналу:

$$\xi = Q_k \left(\frac{f}{f_0} - \frac{f_0}{f} \right) = 50 \left(\frac{1,02 \text{ МГц}}{1 \text{ МГц}} - \frac{1 \text{ МГц}}{1,02 \text{ МГц}} \right) = 1,98,$$

звідки послаблення резонансним контуром частот (1000 ± 20) кГц становить $\sigma = \sqrt{1 + \xi^2} = \sqrt{1 + 1,98^2} = 2,2$.

Зверніть увагу: оскільки у даному випадку $\Delta f \ll f_0$, то розрахунок узагальненої розстройки можна зробити за наближеною формулою

$$\xi = Q_k \left(\frac{f}{f_0} - \frac{f_0}{f} \right) \approx Q_k \frac{2\Delta f}{f_0} = 50 \frac{2 \cdot 20 \text{ кГц}}{1000 \text{ кГц}} = 2,2,$$

звідки також отримуємо, що $\sigma = \sqrt{1 + \xi^2} = \sqrt{1 + 2^2} \approx 2,2$. Цей результат дозволяє використовувати для оцінки селективності по сусідньому каналу наближену формулу

$$\sigma = \sqrt{1 + \xi^2} \approx \sqrt{1 + (2Q_k \Delta f / f_0)^2}.$$

Приклад 2. Обчислити смугу пропускання контуру, резонансна частота якого становить 460 кГц, а добротність 46.

Розв'язок. Смугу пропускання по рівню 0,707 визначають як

$$B = f_0 / Q_k = 460 \text{ кГц} / 46 = 10 \text{ кГц}.$$

Приклад 3. Визначити добротність контуру, який забезпечить послаблення $\sigma = 5$ при розстройці частоти джерела вхідного

гармонічного сигналу відносно резонансної частоти $f_0 = 1500 \text{ кГц}$ при $\Delta f = 750 \text{ кГц}$.

Розв'язок. За умовою задачі селективність контуру $\sigma = \sqrt{1 + \xi^2} = 5 \Rightarrow \xi = \sqrt{\sigma^2 - 1} \approx 4,9$. Оскільки послаблення контуру оцінюють на частоті вхідного гармонічного сигналу та визначають з виразу $f = f_0 + \Delta f = 1500 \text{ кГц} + 750 \text{ кГц} = 2250 \text{ кГц}$, то добротність контуру

$$Q_k = \xi / \left(\frac{f}{f_0} - \frac{f_0}{f} \right) = 5 / \left(\frac{2250 \text{ кГц}}{1500 \text{ кГц}} - \frac{1500 \text{ кГц}}{2250 \text{ кГц}} \right) \approx 5,9.$$

Приклад 4. Вхідне коло з подвійним автотрансформаторним зв'язком (рисунок 3.4) налаштовано на частоту $f_0 = 1 \text{ МГц}$. Індуктивність контуру $L_k = 16 \text{ мкГн}$ а власна провідність $G_k = 100 \text{ мкСм}$. Провідність антени $G_{ant} = 1,5 \text{ мСм}$, а навантаження $G_n = 2 \text{ мСм}$. Визначити оптимальні коефіцієнти включення котушки індуктивності m і n при заданій смузі пропускання $B = 50 \text{ кГц}$.

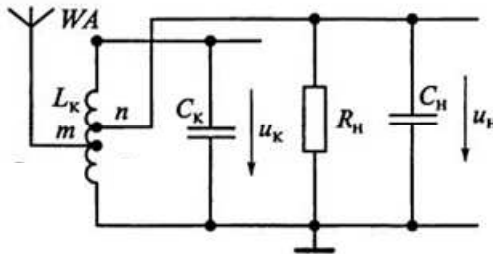


Рисунок 3.4 – Вхідне коло з подвійним автотрансформаторним зв'язком

Розв'язок. Визначимо еквівалентну добротність навантаженого контуру

$$Q_{eq} = f_0 / B = 10^6 / (5 \cdot 10^4) = 20 /$$

Знаючи характеристичний опір контуру

$$\rho_k = 2\pi f_0 L_k = 2\pi \cdot 10^6 \cdot 16 \cdot 10^{-6} \approx 100 \text{ Ом},$$

знаходимо еквівалентну провідність навантаженого контуру

$$G_{eq} = 1 / (\rho_k Q_{eq}) = 1 / (100 \text{ Ом} \cdot 20) = 500 \text{ мкСм}.$$

При побудові вхідних кіл для діапазону помірно високих частот доцільно забезпечити максимальну селективність до першого підсилювального каскаду, тому вибір коефіцієнтів включення котушки індуктивності з антеною m та навантаженням n здійснюють з умови допустимого розширення смуги пропускання конструктивно виконуваного коливального контуру. При цьому узгодження контуру з неналаштованою антеною та навантаженням з метою передачі максимально можливої потужності від джерела сигналу до навантаження забезпечується при виконанні умови

$$m^2 G_{ac} + \frac{G_k}{2} = n^2 G_n + \frac{G_k}{2} = \frac{G_{eq}}{2}.$$

Враховуючи те, що для вхідного кола з автотрансформаторним зв'язком контуру з антеною $G_{ac} = G_{ant}$, отримуємо оптимальні значення коефіцієнтів включення

$$m_{opt} = \sqrt{\frac{G_{eq} - G_k}{2G_{ac}}} = \sqrt{\frac{(500 - 100)10^{-6}}{2 \cdot 1,5 \cdot 10^{-3}}} = 0,365,$$

$$n_{opt} = \sqrt{\frac{G_{eq} - G_k}{2G_n}} = \sqrt{\frac{(500 - 100)10^{-6}}{2 \cdot 2 \cdot 10^{-3}}} = 0,316.$$

Хід заняття

1. Опрацювати лекційний матеріал за темою заняття.
2. Розв'яжіть задачі, які наведено нижче.

Задача 1. Яке послаблення створює резонансний контур, власна резонансна частота якого f_0 , добротність Q_k при абсолютній розстройці Δf кГц? Варіанти завдань наведено у таблиці 3.1.

Кожен вибирає лише один варіант.

Таблиця 3.1

Номер варіанта	f_0 , МГц	Q_k	Δf , кГц
1	2	3	4
1	1,6	50	20
2	2,4	60	20

Продовження таблиці 3.1

1	2	3	4
3	1,1	70	20
4	1,9	50	20
5	1,44	60	20
6	1,4	70	20
7	2,1	50	20
8	1,5	60	20
9	2,4	70	20
10	2,8	50	20
11	1,1	40	20
12	1,9	45	20
13	1,44	50	20
14	1,4	55	20
15	2,1	60	20
16	1,5	65	20
17	2,4	70	20
18	2,8	40	20

19	1,4	45	20
20	2,1	50	20
21	1,5	55	20
22	2,4	60	20
23	2,8	65	20
24	1,6	70	20
25	1,7	50	20

Задача 2. Обчислити смугу пропускання контуру, резонансна частота якого становить f_0 , а добротність Q_k . Варіанти завдань наведено у таблиці 3.2.

Кожен вибирає лише один варіант.

Таблиця 3.2

Номер варіанта	$f_0, \text{МГц}$	Q_k
1	1,6	50
2	2,4	60
3	1,1	70
4	1,9	50
5	1,44	60
6	1,4	70
7	2,1	50
8	1,5	60
9	2,4	70
10	2,8	50
11	1,6	70
12	2,4	50
13	1,1	60
14	1,9	70
15	1,44	50
16	1,4	60
17	2,1	70
18	1,5	70

19	2,4	50
20	2,8	60
21	1,9	70
22	1,44	50
23	1,4	60
24	2,1	70
25	2,4	55

Задача 3. Визначити добротність контуру, який забезпечить послаблення σ при розстройці частоти джерела вхідного гармонічного сигналу відносно резонансної частоти f_0 при заданому значенні Δf . Варіанти завдань наведено у таблиці 3.3.

Кожен вибирає лише один варіант.

Таблиця 3.3

Номер варіанта	f_0 , МГц	Δf , кГц	σ
1	2	3	4
1	1,6	500	5
2	2,4	600	5
3	1,1	500	5
4	1,9	600	5
5	1,44	500	5
6	1,4	600	5
7	2,1	500	5
8	1,5	600	5
9	2,4	500	5
10	2,8	600	5
11	1,6	100	5
12	2,4	150	5
13	1,1	200	5
14	1,9	250	5
15	1,44	300	5
16	1,4	350	5

17	2,1	400	5
18	1,5	450	5
19	2,4	500	5

Продовження таблиці 3.3

1	2	3	4
20	1,9	150	5
21	1,44	200	5
22	1,4	250	5
23	2,1	300	5
24	1,5	350	5
25	2,4	400	5

Задача 4. Вхідне коло з подвійним автотрансформаторним зв'язком (рисунок 3.4) налаштовано на частоту f_0 . Індуктивність контуру L_k а власна провідність G_k . Провідність антени G_{ant} , а навантаження G_n . Визначити оптимальні коефіцієнти включення котушки індуктивності m і n при заданій смузі пропускання $B=20$ кГц. Варіанти завдань наведено у таблиці 3.4.

Кожен вибирає лише один варіант.

Таблиця 3.4

Номер варіанта	f_0 , МГц	L_k , мкГн	G_k , мкСм	G_{ant} , мСм	G_n , мСм
1	2	3	4	5	6
1	1,6	15	100	1,5	2
2	2,4	10	75	2	2,5
3	1,1	12	95	2,5	2,7
4	1,9	18	97	1,7	2,9
5	1,44	13	105	1,9	2,1
6	1,4	15	100	1,4	2,4
7	2,1	10	91	1,5	2,5
8	1,5	16	98	2,1	3,1

Продовження таблиці 3.4

1	2	3	4	5	6
9	2,4	18	108	2,2	3,2
10	2,8	15	101	1,9	3,9
11	1,6	12	95	2,5	2,7
12	2,4	18	97	1,7	2,9
13	1,1	13	105	1,9	2,1
14	1,9	15	100	1,4	2,4
15	1,44	10	91	1,5	2,5
16	1,4	16	98	2,1	3,1
17	2,1	12	95	2,5	2,7
18	1,5	18	97	1,7	2,9
19	1,1	13	105	1,9	2,1
20	1,9	15	100	1,4	2,4
21	1,44	10	91	1,5	2,5
22	1,4	16	98	2,1	3,1
23	2,1	10	91	1,5	2,5
24	1,5	16	98	2,1	3,1
25	1,8	12	95	2,5	2,7

Лабораторна робота №3

Шумові властивості та чутливість приймального пристрою

Мета заняття:

1. Визначення шумових властивостей лінійного тракту приймального пристрою.
2. Визначення чутливості (реальної та порогової) приймального пристрою.
3. Визначення шумових властивостей лінійного тракту приймального пристрою за допомогою спеціалізованого програмного забезпечення.
4. Вивчення основ роботи у калькуляторі NoiseCalc програми AppCAD.

Основні питання заняття:

1. Алгоритм розрахунку шумових властивостей лінійного тракту приймального пристрою на прикладі послідовного з'єднання чотириполіусників.
2. Алгоритм розрахунку чутливості (реальної та порогової) приймального пристрою.
3. Алгоритм розрахунку шумових властивостей лінійного тракту приймального пристрою на прикладі послідовного з'єднання чотириполіусників.
4. Вивчення основ роботи у калькуляторі NoiseCalc програми AppCAD.

Короткі теоретичні відомості

Розглянемо деякі приклади.

Приклад 1. Обчислити обмежену підсиленням чутливість приймального пристрою, якщо коефіцієнт підсилення лінійного тракту $K = 90$ дБ, а амплітуда сигналу на вході детектора $U_d = 1$ В ?

Розв'язок. Чутливість, обмежена підсиленням, визначається ЕРС E_d , наведеною в антені, яка потрібна для отримання заданої

амплітуди на вході детекторі U_d . Тому шукану чутливість можна розрахувати так:

$$E_d = \frac{U_d}{K}.$$

Звідси, з урахуванням перерахунку $K = 90 \text{ дБ} \Rightarrow K = 31623$, отримуємо:

$$E_d = \frac{U_d}{K} = \frac{1 \text{ В}}{31623} = 32 \text{ мкВ}.$$

Зверніть увагу! Чутливість, обмежена підсиленням, властива для приймальних пристроїв потужних сигналів в умовах, коли завади мало впливають на прийом.

Хід заняття

1. Опрацювати лекційний матеріал за темою заняття.
2. Розв'яжіть задачі, які наведено нижче.

Задача 1. Чотири підсилювачі з'єднано послідовно (вхід $\rightarrow$ №1 $\rightarrow$ №2 $\rightarrow$ №3 $\rightarrow$ №4 $\rightarrow$ вихід). Коефіцієнти підсилення за потужністю K_{p1} та коефіцієнти шуму $K_{ш1}$ кожного з них задано у таблиці 4.1. Розрахуйте сумарний коефіцієнт шуму такого чотирикаскадного підсилювача. Яка його шумова температура?

Кожен вибирає лише один варіант.

Задача 2. Чи можливо шляхом зміни послідовності з'єднання підсилювачів (див. задачу №1) зменшити сумарний коефіцієнт шуму такого підсилювача? На скільки вдалось зменшити сумарний коефіцієнт шуму такого оптимізованого підсилювача?

Таблиця 4.1

Номер

K_{p1} , дБ

$K_{ш1}$, дБ

варіанта	№1	№2	№3	№4	№1	№2	№3	№4
1	2	3	4	5	6	7	8	9
1	10	5	12	20	2	1,5	3	4
2	5	10	20	15	1,5	2	4	2,5
3	10	10	18	25	5	5	3	4,5
4	15	10	20	5	2,5	5	4	1,5
5	35	20	20	15	4,7	4	4	2,5
6	30	10	20	3	2,7	5	4	0,9
7	30	15	15	10	2,7	2,5	2,5	5
8	2	10	25	25	0,5	5	4,5	4,5
9	4	20	20	20	0,8	4	4	4
10	11	15	20	25	5	2,5	4	4,5
11	15	4	17	11	2,3	4,5	3,5	1,9
12	12	24	27	6	1,9	3,5	2,3	4,2
13	30	10	20	3	5	2,5	4	4,5
14	15	4	17	11	1,7	2,4	3,6	2,5
15	9	17	4	3	4	2,5	1,9	1,7
16	12	24	27	6	2,3	4,5	3,5	1,9
17	30	10	20	3	5	2,5	4	4,5
18	15	4	17	11	1,5	2	4	2,5
19	9	17	4	3	1,9	3,5	2,3	4,2
20	12	24	27	6	1,7	2,4	3,6	2,5
21	30	10	20	3	5	2,5	4	4,5
22	15	4	17	11	2,3	4,5	3,5	1,9
23	9	17	4	3	4	2,5	1,9	1,7
24	12	24	27	6	1,9	3,5	2,3	4,2
25	30	10	20	3	1,7	2,4	3,6	2,5

Задача 3. Обчислити обмежену підсиленням чутливість приймального пристрою, якщо коефіцієнт підсилення лінійного тракту $K + 20$ дБ (K – коефіцієнт підсилення вашого чотирикаскадного підсилювача, див. задачу №1), а амплітуда сигналу на вході детектора $U_d = 1$ В?

Задача 4. Обчислити реальну та порогову чутливості приймального пристрою P_{ex} , якщо відношення сигнал/шум на виході лінійної частини становить γ_{out} (таблиця 4.2), коефіцієнт шуму $K_{ш}$, а ефективна шумова смуга $\Delta f_{ef} = 8$ МГц. При розрахунках вважати втрати у фідері відсутніми, а нормовану шумову температуру антени t_{ef} рівною нулю. Нормальну температуру вважати рівною 300 К.

Кожен вибирає лише один варіант.

Таблиця 4.2

Номер варіанта	γ_{out}	$K_{ш}$, дБ
1	2	3
1	2	10
2	2,5	20
3	4	30
4	2,7	15
5	3	25
6	4,5	35
7	3	15
8	3,2	24
9	2,8	19

Продовження таблиці 4.2

1	2	3
10	2	9
11	3,1	5
12	2,1	10
13	2,2	12
14	3,2	15
15	2,4	18
16	2,2	20
17	2	25
18	2,3	30

19	1,6	16
20	2,9	28
21	2,4	11
22	1,7	15
23	2,5	30
24	1,8	27
25	3	35

4. Ознайомитись з вхідною інформацією щодо призначення, можливостей та вигляду інтерфейсів онлайн-калькуляторів [1, 2] заняття.

5. Використовуючи онлайн-калькулятори [1, 2], розв'язати задачі №1, 2 до лабораторної роботи №4.

6. Ознайомитись з вхідною інформацією щодо призначення, можливостей та вигляду інтерфейсу калькулятора NoiseCalc програми AppCAD [3] заняття.

8. Використовуючи калькулятор NoiseCalc програми AppCAD, розв'язати задачі №1, 2.

Інтернет-ресурси

1. Cascaded Noise Figure Calculator Режим доступу:
<https://www.ainfoinc.com/t-calculator-noise-figure>

2. Noise Figure - Noise Temperature Calculator Режим доступу:
<https://www.ainfoinc.com/t-calculator-noisefigure-noisetemp>

3. AppCAD Design Assistant Режим доступу:
<https://www.broadcom.com/appcad>

Лабораторна робота №4

Амплітудний детектор у режимах малого та великого сигналів

Мета заняття:

1. Дослідження залежностей вихідної напруги від вхідної потужності для діодного детектора Шотткі за довільної температури, а також чутливості детектора від вхідної потужності за допомогою програми AppCAD.
2. Підбір номіналів елементів діодного детектора Шотткі.
3. Вивчення основ роботи у калькуляторі Large Signal Detector програми AppCAD.
4. Вивчення основ роботи у калькуляторі Small Signal Detector програми AppCAD.

Основні питання заняття:

1. Дослідження залежностей вихідної напруги від вхідної потужності для діодного детектора Шотткі за довільної температури, а також чутливості детектора від вхідної потужності за допомогою програми AppCAD.
2. Підбір номіналів елементів діодного детектора Шотткі.
3. Вивчення основ роботи у калькуляторі Large Signal Detector програми AppCAD.

Хід заняття

1. Ознайомитись з вхідною інформацією щодо призначення, можливостей та вигляду інтерфейсу калькулятора Large Signal Detector програми AppCAD.
2. Виберіть для досліджень один з таких діодів Шотткі: HSCN-5512, HSCN-5531, HSCN-5312, HSCN-5314, HSCN-5310, HSCN-5330, HSCN-5340, HSCN-5332. Скачайте їхні листки технічної документації.
3. Введіть параметри вибраного вами діода до поля «Enter diode parameters and circuit values». Температуру залиште без змін, 25°C.
4. Числові значення у полі «Enter Pin range» залиште без змін.
5. Задайте такі параметри схеми: системний імпеданс Z_0 50 Ом, опір навантаження детектора R_l 1000 Ом, опір узгоджувального шунта

Rm 68 Ом. Отримайте залежності вихідної напруги від вхідної потужності (V_{out} vs. P_{in}) для діодного детектора Шотткі, а також чутливості детектора (Γ vs P_{in}) від вхідної потужності при цих параметрах схеми.

6 Повторіть п. 5, змінюючи послідовно опір навантаження детектора Rl на 2500 та 5000 Ом, опір узгоджувального шунта Rm 47 Ом та 91 Ом. **Для п. 5, 6 наведення графіків розраховуваних характеристик є обов'язковим!**

7 За якої комбінації параметрів схеми буде досягнуто максимального значення вихідної напруги детектора за відсутності ділянки насичення? За якої комбінації параметрів схеми буде досягнуто максимального значення чутливості схеми за мінімального рівня вхідного сигналу? Чи співпадають числові значення комбінацій параметрів схеми в обох цих випадках? Сформулюйте власні рекомендації щодо оптимальних (у сенсі найкращих значень досліджуваних характеристик) параметрів елементів схеми детектора для вибраної моделі діода Шотткі.

8 Ознайомитись з вхідною інформацією щодо призначення, можливостей та вигляду інтерфейсу калькулятора Small Signal Detector програми AppCAD.

9 Дослідження проведіть для того ж самого діода Шотткі, що і у роботі №6.

10 Введіть параметри Rs, EG, n, Is (Iso) вибраного вами діода до поля «Diode parameters». Температуру T зовнішній струм зміщення Io залиште без змін, 25°C та 0 мкА.

11 Числові значення у полі «Enter Pin range» залиште без змін.

12 Задайте такі параметри схеми: системний імпеданс Zo 50 Ом, опір навантаження детектора Rl 100 кОм. Отримайте залежності вихідної напруги від вхідної потужності (V_{out} vs. P_{in}) для діодного детектора Шотткі, а також чутливості детектора (Γ vs P_{in}) від вхідної потужності при цих параметрах схеми.

13 Повторіть п. 5, змінюючи послідовно опір навантаження детектора Rl на 27 кОм та 68 кОм. **Для п. 5, 6 наведення графіків розраховуваних характеристик є обов'язковим!**

14 За якої комбінації параметрів схеми буде досягнуто максимального значення чутливості схеми у всьому діапазоні значень потужності вхідного сигналу? Сформулюйте власні рекомендації щодо оптимальних (у сенсі найкращих значень досліджуваних характеристик) параметрів елементів схеми детектора для вибраної моделі діода Шотткі.

15 Для рекомендованих вами у п. 7 оптимальних параметрів елементів схеми детектора для вибраної моделі діода Шотткі дослідіть зміну характеристик детектора V_{out} vs. P_{in} та Γ vs P_{in} при збільшенні температури спочатку до 50°C , а потім і до 80°C . Опишіть характер спостережуваних вами змін.

Лабораторна робота №5

Чутливість приймального пристрою

Мета заняття:

- 1 Розрахункове визначення чутливості (реальної та порогової) приймального пристрою.
- 2 Розрахункове визначення допустимого коефіцієнта шуму лінійного тракту приймального пристрою.

Основні питання заняття:

- 1 Відпрацювання алгоритму розрахунку чутливості (реальної та порогової) приймального пристрою.
- 2 Відпрацювання алгоритму розрахунку допустимого коефіцієнта шуму лінійного тракту приймального пристрою.

Короткі теоретичні відомості

Розглянемо деякі приклади.

Приклад 1. Обчислити реальну та порогову чутливості приймального пристрою P_{ex} , якщо відношення сигнал/шум на виході лінійної частини становить $\gamma_{out}=3$, коефіцієнт шуму $K_{u\text{ dB}}=4$ дБ, а ефективна шумова смуга $\Delta f_{ef}=10$ МГц. При розрахунках вважати втрати у фідері відсутніми, а нормовану шумову температуру антени t_{ef} рівною нулю. Нормальну температуру вважати рівною 300 К.

Розв'язок. У загальному випадку, чутливість приймального пристрою:

$$P_{ex} = \gamma_{out} k \Delta f_{ef} T_n \left(t_{ef} - 1 + \frac{K_u}{K_{PF}} \right).$$

За умовою задачі втрати у фідері вважаються відсутніми, тобто $K_{PF} = 1$. Значення коефіцієнта шуму, у разях, становить

$$K_{\text{ш}} = 10^{K_{\text{шдБ}}/10} = 10^{4/10} = 2,5.$$

Звідси, для реальної чутливості, маємо:

$$P_{\text{ex}} = \gamma_{\text{out}} k \Delta f_{\text{ef}} T_{\text{ш}} \left(t_{\text{ef}} - 1 + \frac{K_{\text{ш}}}{K_{\text{PF}}} \right) = 3 \cdot 1,38 \cdot 10^{-23} \frac{\text{Вт} \cdot \text{с}}{\text{К}} \cdot 10 \text{ МГц} \times \\ \times 300 \text{ К} \cdot \left(0 - 1 + \frac{2,5}{1} \right) = 1,863 \cdot 10^{-13} \text{ Вт}.$$

Значення ж порогової (граничної) чутливості відповідає значенню $\gamma_{\text{out}} = 1$. Звідси, скориставшись тим самим розрахунковим виразом, отримаємо $P_{\text{ex min}} = 6,21 \cdot 10^{-14} \text{ Вт}$.

Приклад 2. Виразити значення реальної чутливості приймального пристрою, отримане у попередньому прикладі, в одиницях напруги, якщо активний опір приймальної антени становить $R_A = 50 \text{ Ом}$.

Розв'язок. Оскільки в умові задачі нічого не сказано про рівень узгодженості приймальної антени, вважатимемо її узгодженою з фідером приймача, а його, у свою чергу, узгодженим з входом приймального пристрою. Також, оскільки в умові задачі нічого не сказано про втрати у фідері, вважатимемо їх відсутніми. Таким чином, номінальна потужність на виході антени P_n і потужність на вході приймального пристрою P_{ex} дорівнюватимуть одна одній, а її значення відповідатиме випадку узгодження генератора з навантаженням. Тому

$$P_{\text{ex}} \equiv P_n = U_{\text{ex}}^2 / (4 \cdot R_A) \Rightarrow U_{\text{ex}} = \sqrt{4 \cdot R_A \cdot P_{\text{ex}}}.$$

Таким чином, чутливість в одиницях напруги на вході приймального пристрою

$$U_{\text{ex}} = \sqrt{4 \cdot R_A \cdot P_{\text{ex}}} = \sqrt{4 \cdot 50 \text{ Ом} \cdot 1,863 \cdot 10^{-13} \text{ Вт}} = 6,104 \cdot 10^{-6} \text{ В} \approx 6,1 \text{ мкВ}.$$

Хід заняття

- 1 Повторити матеріал лекції за темою «Чутливість приймача».
- 2 Розв'язати задачі:

Задача 1. Обчислити реальну та порогову чутливості приймального пристрою P_{ex} , якщо відношення сигнал/шум на виході лінійної частини становить γ_{out} , коефіцієнт шуму $K_{\text{ш}}$, та ефективна шумова смуга Δf_{ef} (таблиця 8.1). При розрахунках вважати втрати у фідері відсутніми, а нормовану шумову температуру антени t_{ef} рівною нулю. Нормальну температуру вважати рівною 300 К.

Кожен вибирає лише один варіант.

Таблиця 8.1

Номер варіанта	γ_{out}	$K_{\text{ш}}$, дБ	Δf_{ef} , МГц
1	2	3	4
1	2	10	8
2	2,5	20	10
3	4	30	20
4	2,7	15	8
5	3	25	10
6	4,5	35	20
7	3	15	8
8	3,2	24	10
9	2,8	19	20
10	2	9	8
11	3,1	5	10
12	2,1	10	20

13	2,2	12	8
14	3,2	15	10
1	2	3	20
15	2,4	18	8
16	2,2	20	10
17	2	25	20
18	2,3	30	8
19	1,6	16	10
20	2,9	28	20
21	2,4	11	8
22	1,7	15	10
23	2,5	30	20
24	1,8	27	8
25	3	35	10

Задача 2. Виразити значення реальної чутливості приймального пристрою, отримане у попередній задачі, в одиницях напруги, якщо активний опір приймальної антени становить $R_A = 50 \text{ Ом}$.

Лабораторна робота №6

Розподіл діапазону робочих частот на піддіапазони

Мета заняття:

- 1 Розрахунковий розподіл діапазону робочих частот на піддіапазони.
- 2 Порівняння між собою різних способів розподілу діапазону робочих частот на піддіапазони.

Основні питання заняття:

- 1 Відпрацювання алгоритмів розподілу діапазону робочих частот на піддіапазони.

Короткі теоретичні відомості

Діапазон робочих частот приймача розподіляють на піддіапазони, якщо коефіцієнт перекриття діапазону приймача K_{bp} більший за коефіцієнт перекриття діапазону K_b використовуваних резонансних систем зі змінним настроюванням, а також якщо потрібно отримати більш високі та постійні за діапазоном чутливість і селективність, плавніше настроювання, більшу точність частоти настроювання приймача. Але при цьому потрібно враховувати, що при збільшенні кількості піддіапазонів ускладнюється схема, конструкція та експлуатація приймача, збільшується його об'єм та маса, знижується надійність, дорожчає виробництво, збільшується час перестроювання приймача. Тому при поділі діапазону робочих частот на піддіапазони приймають компромісне рішення.

Зазвичай використовують три способи розподілу діапазону робочих частот на піддіапазони:

- 1) **спосіб рівних коефіцієнтів перекриття піддіапазону;**
- 2) **спосіб рівних частотних інтервалів;**
- 3) **комбінований спосіб** (нижню ділянку спільного діапазону розбивають за способом рівних коефіцієнтів перекриття піддіапазону, верхню – за способом рівних частотних інтервалів).

Оцінювання діапазону робочих частот здійснюють за коефіцієнтом перекриття діапазону

$$K_{b\,p} = f_{p\,\max} / f_{p\,\min} , \quad (9.1)$$

де $f_{p\,\min}$, $f_{p\,\max}$ – мінімальна та максимальна робочі частоти приймача. Якщо значення $K_{b\,p}$ більше за коефіцієнт перекриття діапазону K_b вибраної резонансної системи (таблиця 9.1), то розподіл на піддіапазони неминучий.

Таблиця 9.1

<i>Вид резонансної системи</i>	K_b
Контур з зосередженими параметрами, перестроюваний:	
<i>конденсатором змінної ємності.....</i>	<i>2,5...3,5</i>
<i>катушкою змінної ємності.....</i>	<i>1,4...3,0</i>
<i>варикапом.....</i>	<i>2,3...2,7</i>
<i>транзистором.....</i>	<i>1,1...1,15</i>
Чвертьхвильовий відрізок коаксіальної лінії, перестроюваний:	
<i>короткозамикаючим плунжером.....</i>	<i>1,4...2,0</i>
<i>підстроювальним конденсатором.....</i>	<i>1,08...1,1</i>
Об'ємний резонатор, перестроюваний:	
<i>підстроювальними дисками та штирями....</i>	<i>1,05...2,0</i>
<i>рухомими стінками-плунжерами.....</i>	<i>1,1...1,8</i>
<i>механічною деформацією стінок.....</i>	<i>1,03...1,05</i>

При розподілі діапазону робочих частот приймача на піддіапазони за способом рівних коефіцієнтів перекриття піддіапазону контури тракту ПЧ містять мінімальну кількість елементів, прості за схемою та конструкцією. Для спряження настроювань контурів тракту ПЧ і гетеродина у контури гетеродина доводиться включати додаткові елементи, що зменшує стабільність частоти гетеродина. При розподілі за способом рівних частотних інтервалів різко

збільшується кількість піддіапазонів, і, як наслідок, ускладнюється схема та конструкція приймача. Проте за такого способу розподілу легше виконати вимоги до точності встановлення частоти на верхніх піддіапазонах.

Розбиття робочого діапазону частот на діапазони здійснюють не «впритул», а з «запасом» на 3...5 %.

Питання, пов'язані з щільністю та плавністю настроювання та необхідним запасом перекриття діапазонів відпадають при використанні в якості гетеродину супергетеродинного приймача високостабільного синтезатора частоти з дискретним встановленням частоти.

Розподіл на діапазони способом рівних коефіцієнтів перекриття піддіапазону:

При розподілі за цим способом коефіцієнти перекриття у всіх діапазонах однакові $K_{bi} \approx K_b$ та не повинні перевищувати, з однієї сторони, коефіцієнти перекриття контурів у даному діапазоні частот, а з іншої сторони, допустимого значення при якому досягається задана точність встановлення частоти настроювання приймача.

Коефіцієнт перекриття робочого діапазону (9.1) дорівнює також добутку перекриттів i -х діапазонів:

$$K_{bp} = \prod_{i=1}^m \frac{K_{bi}}{1,03...1,05}.$$

Оскільки за даного способу $K_{bi} \approx K_b$, то, логарифмуючи та округлюючи до цілого числа, отримуємо кількість піддіапазонів:

$$m \approx \frac{\lg K_{bp}}{\lg [K_b / (1,03...1,05)]}.$$

За кінцеве значення m приймають більше ціле число.

Після цього визначають розрахунковий коефіцієнт перекриття піддіапазону:

$$K'_b = 1,06 \sqrt[m]{\frac{f_{p \max}}{f_{p \min}}},$$

та обчислюють граничні частоти кожного піддіапазону:

$$f_{1 \min} = 0,97 \cdot f_{p \min}; f_{1 \max} = K'_b \cdot f_{1 \min};$$

$$f_{2 \min} = 0,94 \cdot f_{1 \max}; f_{2 \max} = K'_b \cdot f_{2 \min};$$

...

$$f_{m \min} = 0,94 \cdot f_{(m-1) \max}; f_{n \max} = 1,03 \cdot f_{p \max}.$$

Приклад 1. Приймач повинен забезпечувати настройку у діапазоні частот 3,5 ... 30 МГц. На скільки піддіапазонів частот потрібно розбити даний діапазон?

Розв'язок. Оскільки коефіцієнт перекриття діапазону робочих частот

$$K_{bp} = \frac{f_{p \max}}{f_{p \min}} = \frac{30 \text{ МГц}}{3,5 \text{ МГц}} = 8,57$$

перевищує 3,5 (максимальне значення, з наведених у таблиці 9.1), то без розподілу на піддіапазони не обійтись. Якщо взяти за основу власне контур з зосередженими параметрами, перестроюваний конденсатором змінної ємності (максимальне значення $K_b = 3,5$), то у цьому випадку при 3 % запасі мінімально необхідна кількість піддіапазонів:

$$m \approx \frac{\lg K_{bp}}{\lg[K_b/1,03]} = \frac{\lg 8,57}{\lg[3,5/1,03]} = 1,756 \Rightarrow m = 2.$$

Розрахунковий коефіцієнт перекриття піддіапазону:

$$K'_b = 1,06_m \sqrt{\frac{f_{p\max}}{f_{p\min}}} = 1,06_2 \sqrt{\frac{30 \text{ МГц}}{3,5 \text{ МГц}}} = 3,103,$$

Звідси граничні частоти кожного з цих піддіапазонів:

$$f_{1\min} = 0,97 \cdot f_{p\min} = 0,97 \cdot 3,5 \text{ МГц} = 3,395 \text{ МГц};$$

$$f_{1\max} = K'_b \cdot f_{1\min} = 3,103 \cdot 3,395 \text{ МГц} = 10,536 \text{ МГц};$$

$$f_{2\min} = 0,94 \cdot f_{(m-1)\max} = 0,94 \cdot 10,536 \text{ МГц} = 9,904 \text{ МГц};$$

$$f_{n\max} = 1,03 \cdot f_{p\max} = 1,03 \cdot 30 \text{ МГц} = 30,9 \text{ МГц}.$$

Остаточно маємо: перший піддіапазон 3,4...10,54 МГц; другий піддіапазон 9,9...30,9 МГц.

Хід заняття

1 Повторити матеріал лекції за темою «Настроювання контурів. Перекриття діапазону частот».

2 Розв'язати задачу:

Задача 1. Приймач повинен забезпечувати настройку у заданому діапазоні частот (таблиця 9.2). На скільки піддіапазонів частот потрібно розбити даний діапазон, якщо взяти за основу контур з зосередженими параметрами, перестроюваний конденсатором змінної ємності та варикапом?

Кожен вибирає лише один варіант.

Таблиця 9.2

Номер варіанта	Діапазон частот приймача, МГц
1	2
1	3...28
2	3,5...30
3	1,6...30
4	0,1...30

5	65,8...74
6	100...108
7	5...20
8	9,5...40
9	0,525...1,605
10	4...30
11	4...15
12	10...30
13	65,8...108
14	1,5...20
15	10...30
16	5...30
17	5,25...16,05
18	65,8...100
19	10...50

Продовження таблиці 9.2

1	2
20	3,5...35
21	52,5...160,5
22	4...30
23	30...100
24	10...70
25	2...20

Лабораторна робота №7
Частотні фільтри у приймальних пристроях

Мета заняття:

1 Практичне знайомство з методикою розрахунку частотних фільтрів нижніх частот та смугових фільтрів за допомогою спеціалізованих онлайн-калькуляторів.

2 Розрахунок частотних фільтрів нижніх частот та смугових фільтрів приймальних пристроїв.

Основні питання заняття:

- 1 Методика розрахунку частотних фільтрів.
- 2 Онлайн-калькулятори для розрахунку частотних фільтрів.
- 3 Відпрацювання алгоритму розрахунку частотних фільтрів нижніх частот та смугових фільтрів приймальних пристроїв.

Короткі теоретичні відомості

Терміном фільтр (частотний фільтр) називають прилад, який здатен здійснювати частотну селекцію чи виділення електричних сигналів певних частот.

Вимоги до частотних характеристик фільтрів задають, вказуючи частотну смугу пропускання, у межах якої внесене послаблення $A = 20 \lg(U_{in}/U_{out})$ не повинно перевищувати деякого допустимого значення. Поза смугою пропускання внесене послаблення має бути якомога більшим. Іноді оговорюють частотну залежність фази коефіцієнта передачі, найчастіше у вигляді допустимого відхилення від лінійного закону при зміні частоти у смузі пропускання. Під U_{in} розуміють потужність падаючої хвилі на вході фільтра, а під потужністю U_{out} – потужність хвилі на його виході.

У смузі пропускання ідеального фільтра $A = 0$, а в смузі затримки $A = \infty$. За взаємним розташуванням смуг пропускання та затримки, тобто за формою амплітудно-частотної характеристики (АЧХ), фільтри поділяють на:

– **фільтр нижніх частот (low pass filter)** (ФНЧ) пропускає лише частоти, які розташовані нижче за певну задану частоту (частоту зрізу). Всі частоти вищі за частоту зрізу послаблюються (рисунок 10.1, а);

– **фільтр верхніх частот (high pass filter)** (ФВЧ) пропускає лише частоти, які розташовані вище, ніж частота зрізу (рисунок 10.1, б);

– **смугові фільтри (band pass filter)** (СФ) пропускають лише ті частоти, які розташовані у певній смузі та послаблюють всі частоти поза цією смугою. Смуговий фільтр має верхню та нижню частоти зрізу. Відповідний вибір цих частот дозволяє виділяти різні частотні ділянки ФКГ сигналу (рисунок 10.1, в);

– **режекторні фільтри (band stop filter)** (РФ) послаблюють всі частоти між двома частотами зрізу (рисунок 10.1, г).

В ідеальних фільтрах у смузі затримки вхідна потужність не приходить на вихід. Вона або повністю відбивається від входу фільтра, або поглинається в його елементах. У першому випадку фільтри відносять до фільтрів відбивного типу, в другому – до фільтрів поглинаючого типу.

Зауважимо, що повне значення внесених фільтром втрат складається з теплових втрат та втрат, спричинених відбиттям частини енергії від його входу.

Для зменшення теплових втрат фільтри відбивного типу зазвичай виконують з реактивних елементів. Параметри реактивних елементів підбирають так, щоб на частотах смуги пропускання відбиті від них хвилі компенсували одна одну на вході фільтра. При цьому потужність на вході фільтра проходить на його вихід практично без відбиття. На частотах смуги затримки компенсація відбитих хвиль відсутня і потужність на вході фільтра практично повністю відбивається від нього.

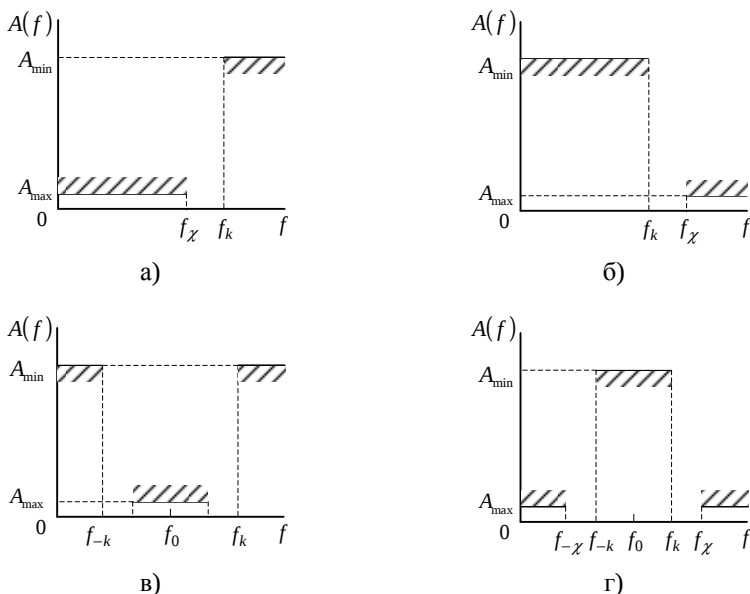


Рисунок 10.1 – АЧХ фільтрів: ФНЧ (а), ФВЧ (б), СФ (в), РФ (г)

Синтез фільтрів умовно можна розбити на три етапи. На першому етапі розв'язується **задача апроксимації**. Це пошук аналітичної функції (математичного описання), яка з потрібною точністю відтворює вид заданої характеристики пристрою (фільтра у даному випадку). Такою характеристикою може бути частотна, фазова або перехідна. Згідно цього, синтез розподіляють на синтез у частотній області (частотні та фазові характеристики) і на синтез у часовій області (перехідні характеристики). Синтез у частотній області поширеніший, ніж синтез в часовій області. Далеко не кожен вид частотної (фазової) характеристики можна отримати за допомогою використовуваних каскадів, тому на апроксимуючі функції накладається обмеження.

На наступному етапі розв'язується **задача реалізації** – пошук сукупності кіл, що містять вихідні елементи (пасивні елементи, або активні ланки потрібного порядку), які дають в результаті

характеристики, досить близькі до вибраної апроксимуючої функції. Зазвичай другим етапом є пошук способу з'єднання використовуваних ланок в єдиний пристрій (*структурний синтез*). У зв'язку з тим, що будь-яку необхідну характеристику можна реалізувати великою кількістю різних способів, структурний синтез є неоднозначною задачею. Як правило, розглядають декілька найбільш простих реалізацій (наприклад, ланки першого та другого порядків), які потім уточнюють, залучаючи додаткові критерії оптимізації вибраного рішення. Критеріями оптимізації можуть бути мінімальна кількість активних або пасивних елементів, чутливість до зміни їхніх параметрів і т.д.

Третім, заключним етапом є *вибір виду самих ланок*, тобто електричних схем тих вихідних елементів, з яких складається весь пристрій. Безпосередня реалізація апроксимуючої функції високого порядку є доволі складною. Тому її частіше представляють у вигляді комбінації більш простих парціальних функцій не вище другого порядку.

Зазвичай синтезують ФНЧ, а фільтри інших типів зводять до АЧХ ФНЧ за допомогою спеціальних частотних перетворень.

Ідеальну АЧХ для ФНЧ неможливо тримати за допомогою схеми зі скінченної кількості елементів n . Тому зазвичай використовують прийнятніший для практики спосіб задання вимог до АЧХ: у смузі пропускання значення внесеного згасання не повинно перевищувати задане значення $A_{\max}$, а у смузі затримки внесене згасання має бути більшим за задане значення $A_{\min}$ (рисунк 10.1). Таку АЧХ апроксимують тією чи іншою функцією.

На практиці найчастіше зустрічається два способи апроксимації ідеальної АЧХ, які дозволяють для даної кількості елементів n максимально наблизити реальну характеристику фільтра до ідеальної.

За першого способу параметри елементів фільтра підбирають таким чином, щоб отримати максимально плоску АЧХ (характеристику Батерворта). Ця характеристика називається максимально плоскою тому, що згасання монотонно збільшується зі збільшенням частоти.

За другого способу апроксимації реалізують рівномірно пульсуючу характеристику (характеристику Чебишева). Ця характеристика найменше відхиляється від ідеальної у смузі пропускання фільтра. За однакової кількості елементів n характеристика Чебишева має більшу стрімкість за характеристику Батерворта поза смугою пропускання, але при цьому у неї гірша рівномірність у смузі пропускання.

Хід заняття

1 Опрацювати теоретичний матеріал, наведений у «Коротких теоретичних відомостях».

2 Використовуючи калькулятор Analog Filter Wizard [1], синтезувати схему електричну принципову ФНЧ, варіанти вихідних даних до якого наведено у таблиці 10.1.

Кожен вибирає лише один варіант.

Таблиця 10.1

Номер варіанта	Верхня частота фільтра, МГц	Початок смузи затримки, МГц	Рівень мінімального затухання у смузі затримки, дБ
1	3	5	– 30
2	30	55	– 40
3	16	25	– 30
4	10	15	– 40
5	50	80	– 30
6	10	25	– 40
7	20	27	– 30
8	40	60	– 40
9	0,525	1,0	– 30
10	4	10	– 40
11	15	35	– 30
12	10	35	– 40
13	44	80	– 30

14	20	40	– 40
15	15	29	– 30
16	35	50	– 30
17	16,5	30	– 30
18	8,5	18	– 40
19	33,6	45	– 30
20	44	52	– 40
21	21,5	47	– 30
22	30	47	– 40
23	30,5	50	– 30
24	47	68	– 40
25	46	70	– 30

3 Використовуючи калькулятор Analog Filter Wizard [1], синтезувати схему електричну принципову смугового фільтра, варіанти вихідних даних до якого наведено у таблиці 11.1.

Кожен вибирає лише один варіант.

Таблиця 11.1

<i>Номер варіанта</i>	<i>Центральна частота, кГц</i>	<i>Смуга пропускання фільтра, кГц</i>	<i>Ширина смуги затримки, кГц</i>	<i>Рівень мінімального затухання у смугі затримки, дБ</i>
1	45	3	10	– 30
2	25	20	60	– 40
3	120	16	50	– 30
4	125	30	80	– 40
5	307	20	60	– 30
6	336	16	50	– 40
7	343	30	80	– 30

8	150	20	60	– 40
9	168	16	50	– 30
10	85	30	80	– 40
11	125	20	60	– 30
12	215	16	50	– 40
13	300	30	80	– 30
14	125	20	60	– 40
15	300	16	50	– 30
16	85	30	80	– 30
17	125	20	60	– 30
18	215	16	50	– 40
19	300	30	80	– 30
20	125	20	60	– 40
21	215	16	50	– 30
22	85	30	80	– 40
23	125	20	60	– 30
24	215	16	50	– 40
25	85	30	80	– 30

Інтернет-ресурси

1. Filter design tool. Analog Filter Wizard – Режим доступу:
<https://tools.analog.com/en/filterwizard/>

Лабораторна робота №8
Дослідження перетворювача частоти

Мета заняття:

- 1 Дослідження виразу для розрахунку комбінаційних частот за допомогою програми AppCAD.
- 2 Вивчення основ роботи у модулі Mixer Spur Finder програми AppCAD.

Основні питання заняття:

- 1 Дослідження залежностей вихідної напруги від вхідної потужності для діодного детектора Шоттки за довільної температури, а також чутливості детектора від вхідної потужності за допомогою програми AppCAD.
- 2 Вивчення основ роботи у модулі Mixer Spur Finder програми AppCAD.

Короткі теоретичні відомості

Змішувачі генерують гармоніки вхідного радіочастотного сигналу та частоти гетеродина. Це означає, що існує широкий діапазон небажаних (паразитних) частот, багато з яких можуть лежати за межами бажаної вхідної смуги пропускання, що дасть небажані відгуки у діапазоні ПЧ. Модуль Mixer Spur Finder розраховує всі помилкові комбінаційні частоти для вибраних користувачем частот, смуг пропускання та гармонік гетеродина та сигналу.

Фізично модуль Mixer Spur Finder обчислює частоти, які задовольняють рівнянню

$$f_{IF} = \pm m f_{RF} \pm n f_{LO},$$

де f_{IF} – діапазон проміжних частот змішувача ; f_{RF} – діапазон вхідних радіочастот; f_{LO} – діапазон частот гетеродина; m і n – порядок гармонік радіочастоти та гетеродина, для яких виконується розрахунок.

Результатом обчислення є список діапазонів радіочастот (з індексами m і n включно), які призводять до помилкового відгуку у межах зазначеного діапазону проміжних частот – паразитних частот.

Модуль Mixer Spur Finder має два варіанти конфігурації:

- 1) змінного діапазону частот гетеродина (Variable RF – Variable LO);
- 2) фіксована частота гетеродина (Variable RF – Fixed LO).

Хід заняття

1 Ознайомитись з вхідною інформацією щодо призначення, можливостей та вигляду інтерфейсу модуля Mixer Spur Finder програми AppCAD.

2 Використовуючи модуль модуля Mixer Spur Finder, підібрати у режимі «Variable RF – Variable LO» такий діапазон частот коливача гетеродина та кількість гармонік цих частот, щоб отримати мінімальну кількість паразитних частот (кількість гармонік частот вхідного радіосигналу змінювати від 3 до 5). Значення проміжної частоти обирайте з ряду: 85; 125; 215; 300 465; 500 кГц. Варіанти вихідних даних наведено у таблиці 12.1.

Кожен вибирає лише один варіант.

Таблиця 12.1

Номер варіанта	Смуга проміжних частот, кГц	Смуга вхідних радіочастот
1	3	527...1607 кГц
2	20	9,5...9,8 МГц
3	160	149...284 МГц
4	300	149...284 МГц
5	20	11,7...12,1 МГц
6	16	65...74 МГц
7	300	100...108 МГц
8	20	12...14 МГц

9	160	156...162 МГц
10	300	146...174 МГц
11	200	300...307 МГц
12	160	336...343 МГц
13	30	47...49 МГц
14	20	56...58 МГц
15	160	150...169 МГц
16	300	440...450 МГц
17	20	11,7...12,1 МГц
18	16	65...74 МГц
19	30	100...108 МГц
20	20	12...14 МГц
21	160	156...162 МГц
22	300	146...174 МГц
23	200	527...1607 кГц
24	16	9,5...9,8 МГц
25	300	149...284 МГц

Лабораторна робота №9

Колірні простори RGB та HSV

Мета заняття:

- 1 Практичне знайомство з існуючими системами представлення кольору.
- 2 Розрахунок в математичному представленні.
- 3 Фактична реалізація блоку перетворення.

Основні питання заняття:

- 1 Методика розрахунку.
- 2 Огляд онлайн перетворювачів та їх налаштувань.
- 3 Відпрацювання алгоритму перетворення, знаходження помилки перетворення.

Короткі теоретичні відомості

Колірний простір – це певна організація кольорів, яка є набором інформації та даних, які визначають, як система або пристрій відображає колір користувачеві. Кольорові простори тісно пов'язані з методами опису та визначення цих кольорів за допомогою їхніх відповідних колірних кодів. Для різних завдань перевага надається різним кольоровим кодам. Наприклад, кольори RGB призначені для Інтернету, а кольори CMYK – для друку.

Колірний простір RGB (Red, Green, Blue) складається з усіх можливих кольорів, які можна отримати шляхом змішування червоного, зеленого та синього. Ця кольорова модель популярна у фотографії, телевізійній та комп'ютерній графіці. Значення RGB задаються як ціле число від 0 до 255 (хоча може бути і від 0 до 1023 для 10bit,...). Так, наприклад, `rgb(255,0,0)` відображається червоним. Для червоного параметра встановлено найвище значення (255), а для решти встановлено значення 0. HSV означає Hue, Saturation і Value (значення яскравості). HSV схожий на HSL, але це дві різні моделі кольорів (рисунок 9.1 для візуалізації відмінностей):

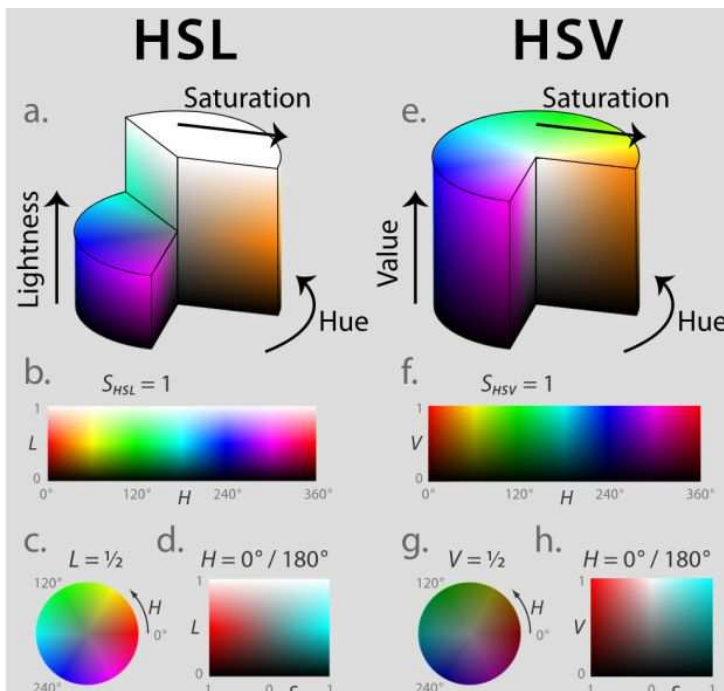


Рисунок 9.1

Обидва вони базуються на циліндричних геометріях, але HSV базується на моделі «гексанусу», а HSL – на моделі «бі-гексанусу».

Вибір кольору HSV починається з вибору одного з доступних відтінків, а потім налаштування значень відтінку та яскравості. Відтінок задає положення кольору на колірному колі (від 0 до 360). Насиченість – це відсоткове значення насичення (від 0 % до 100 %). Яскравість – це відсоток яскравості (від 0 % до 100 %).

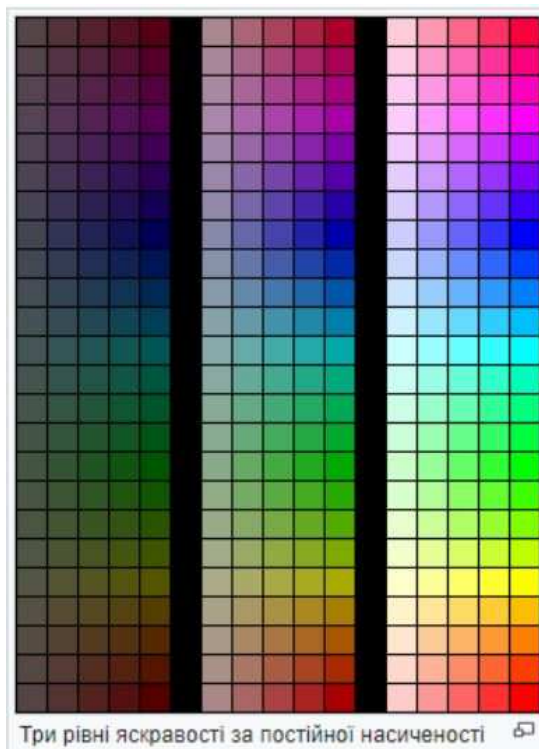


Рисунок 9.2

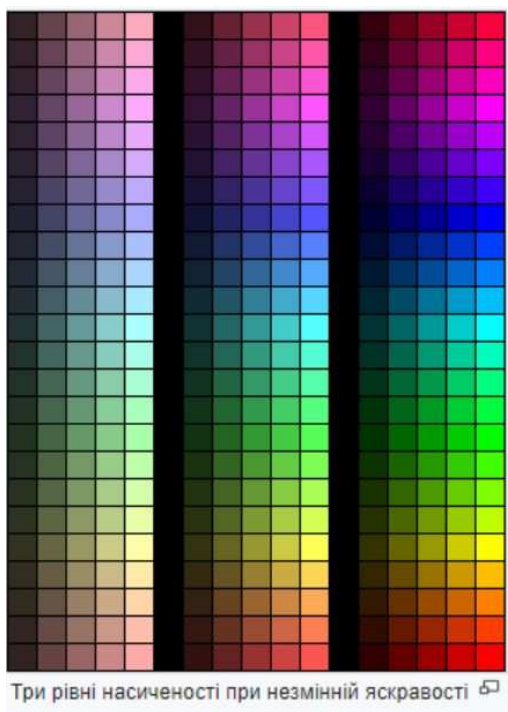


Рисунок 9.3

Часто художники вважають за краще використовувати HSV замість інших моделей, таких як RGB і CMYK, тому що вони вважають, що пристрій HSV ближче до людського сприйняття кольорів. Значення кольору у HSV відповідає на питання – *Що це за колір? Наскільки він насичений? Наскільки він світлий чи темний?*

у системах відображення також використовують налаштування HSV, для цього вхідний RGB сигнал пропускають через два блоки перетворення RGB2HSV та HSV2RGB, а у проміжку між ними додають блок корекції під вказівки користувача.

Порядок розрахунку значень HSV показано на рисунку 9.4

■ colour cone

- $H = \text{hue} / \text{colour in degrees} \in [0, 360]$
- $S = \text{saturation} \in [0, 1]$
- $V = \text{value} \in [0, 1]$

■ conversion RGB $\rightarrow$ HSV

- $V = \max(R, G, B), \quad \min = \min(R, G, B)$
 - $S = (\max - \min) / \max \quad (\text{or } S = 0, \text{ if } V = 0)$
 - $H = 60 \times \begin{cases} 0 + (G - B) / (\max - \min), & \text{if } \max = R \\ 2 + (B - R) / (\max - \min), & \text{if } \max = G \\ 4 + (R - G) / (\max - \min), & \text{if } \max = B \end{cases}$
- $H = H + 360, \text{ if } H < 0$

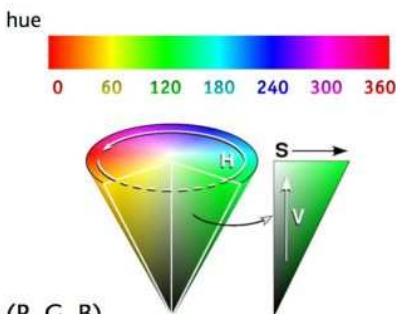


Рисунок 9.4 – Перетворення RGB в HSV

Розрахунок значень RGB зі значень HSV виконується з помилкою – тому якщо значення RGB перетворити в HSV, а потім в зворотному напрямку значення можуть дещо відрізнятись від початкових. Сам розрахунок виконується наступним чином (рисунок 9.5):

Отримані значення червоного, зеленого та синього каналів RGB обчислюються у відсотках. Щоб привести їх у відповідність до поширеного подання необхідно помножити кожне з них на (255/100).

Загалом окрім HSV та RGB можливі перетворення і в інші кольорові простори такі як HEX, HSL, HWB, LAB, CMYK, XYZ, LCH, OKLAB тощо (в залежності від сфери застосування).

Для будь-яких відтінків $H \in [0, 360]$, насиченості $S \in [0, 100]$ та яскравості $V \in [0, 100]$:

$$H_i = \left\lfloor \frac{H}{60} \right\rfloor \bmod 6$$

$$V_{min} = \frac{(100 - S) * V}{100}$$

$$a = (V - V_{min}) * \frac{H \bmod 60}{60}$$

$$V_{inc} = V_{min} + a$$

$$V_{dec} = V - a$$

H_i	R	G	B
0	V	V_{inc}	V_{min}
1	V_{dec}	V	V_{min}
2	V_{min}	V	V_{inc}
3	V_{min}	V_{dec}	V
4	V_{inc}	V_{min}	V
5	V	V_{min}	V_{dec}

Рисунок 9.5 – Перетворення HSV в RGB

Приклад 1. Є значення точки в системі RGB (10;69;128). Потрібно знайти значення в системі HSV.

Розв'язок.

$$V = \max = \max(10/255; 69/255; 128/255) = 128/255 = 0.501961 = 50.1961\%, \quad (\max = B) \quad \min = \min(10/255; 69/255; 128/255) = 10/255 = 0.039216 = 3.9216\%$$

$$S = (\max - \min) / \max = (128 - 10) / 128 = 0.921875 = 92.1875\%$$

$$H = 60 * (4 + (10 - 69) / (128 - 10)) = 210^\circ$$

Приклад 2. Є зразок зображення (1920x1080px) в системі RGB (8bit). Потрібно реалізувати функцію зменшення насиченості зображення на 30%.

Розв'язок.

```
Void EditSaturation(char *p_fileIN, char *p_fileOUT)
{
```

```
    float saturation_shift = -0.30;
    int x,y;
```

```

for(x=0;x<1920;x++)
{
    for(y=0;y<1080;y++)
    {
        Rin= (unsigned char) in_image[x][y][2];
        Gin= (unsigned char) in_image[x][y][1];
        Bin= (unsigned char) in_image[x][y][0];
        RGB2HSV(R_in, G_in, B_in, H_out, S_out, V_out);
        HSV2RGB( H_out, (S_out + S_out *saturation_shift), V_out,
R_out, G_out, B_out,);
        out _image[x][y][2]= (unsigned char) R_out;
        out _image[x][y][1]= (unsigned char) G_out;
        out _image[x][y][0]= (unsigned char) B_out;

    }
}
}

```

* детальне виконання в додаток 1

Приклад 3. Є потік даних зображення в системі RGB (3*10bit).

Потрібно реалізувати мовою Verilog функцію зменшення насиченості зображення на 10% та вказати результуючу затримку даних.

Розв’язок. Реалізація мовою Verilog модулю **edit_hsv.v**

```

module edit_hsv(
    input wire [9:0] R_in,
    input wire [9:0] G_in,
    input wire [9:0] B_in,

    input wire      CLK,

    output reg [9:0] R_out,
    output reg [9:0] G_out,
    output reg [9:0] B_out
);

```

```

wire [24:0] rgb_h;
wire [17:0] rgb_s;
wire [17:0] rgb_v;
wire [17:0] rgb_s_plus;

rgb2hsv(
    .R(R_in),
    .G(G_in),
    .B(B_in)

    .CLK(CLK),

    .H(rgb_h), // [24:0] [0..360] in 8.16
fixed point
    .S(rgb_s), // [17:0] [0..1] in 2.16 fixed
point
    .V(rgb_v) // [17:0] [0..1] in 2.16 fixed
point
);

hsv2rgb(
    .H(rgb_h), // [24:0] [0..360] in 8.16
fixed point
    .S(rgb_s_plus), // [17:0] [0..1] in 2.16 fixed
point
    .V(rgb_v), // [17:0] [0..1] in 2.16 fixed
point

    .CLK(CLK),

    .R(hsv_r),
    .G(hsv_g),
    .B(hsv_b)
);

assign rgb_s_plus = rgb_s - rgb_s *18'h0199A; //edit saturation -10%

```

```

always@(posedge CLK)
    begin
        R_out<=hsv_r;
        G_out<=hsv_g;
        B_out<=hsv_b;
    end

endmodule

```

** реалізація мовою Verilog модулю **rgb2hsv** в додатку 3

*** реалізація мовою Verilog модулю **hsv 2rgb** в додатку 4

Хід заняття

1. Опрацювати лекційний матеріал за темою заняття.
2. Пошук, ознайомлення та освоєння онлайн-ресурсу для перетворення кольірних просторів.
3. Виконати задачі 1 - 3 наведені нижче.
4. Запустити лістинг програми наведеної в додатку 2.

Задача 1. Є значення кольору в системі RGB. Потрібно знайти значення в системі HSV.

Змінити обраний параметр (Hue/saturation/value) відповідно до варіанту. Отриманий результат перетворити в систему RG B. Перевірити отримані значення в обраному онлан-ресурсі.

Варіант1 RGB (0;231;12), Hue + 60°

Варіант2 RGB (15;255;50), saturation – 60%

Варіант3 RGB (96;96;96), value + 75%

Задача 2. Є зразок зображення (1920x1080px) в системі RGB (8bit).

Потрібно реалізувати функцію зменшення насиченості зображення на 30%.

Варіант1 - Потрібно реалізувати функцію зміни відтінку на 65°

Варіант2 - Потрібно реалізувати функцію зменшення насиченості зображення на 54%

Варіант3 - Потрібно реалізувати функцію збільшення яскравості зображення на 36%

Задача 3. Є потік даних зображення в системі RGB (3*10bit).

Варіант1 - Потрібно реалізувати функцію зміни відтінку на 65°

Варіант2 - Потрібно реалізувати функцію зменшення насиченості зображення на 54%

Варіант3 - Потрібно реалізувати функцію збільшення яскравості зображення на 36%

Додаток1. Перетворення на rgb2hsv та hsv2rgb мовою c++

```
#define min_f(a, b, c) (fminf(a, fminf(b, c)))  
#define max_f(a, b, c) (fmaxf(a, fmaxf(b, c)))
```

```
void rgb2hsv(const unsigned char &src_r, const unsigned char &src_g, const  
unsigned char &src_b, unsigned char &dst_h, unsigned char &dst_s, unsigned  
char &dst_v)  
{  
    float r = src_r / 255.0f;  
    float g = src_g / 255.0f;  
    float b = src_b / 255.0f;
```

```

float h, s, v; // h:0-360.0, s:0.0-1.0, v:0.0-1.0

float max = max_f(r, g, b);
float min = min_f(r, g, b);

v = max;

if (max == 0.0f) {
    s = 0;
    h = 0;
}
else if (max - min == 0.0f) {
    s = 0;
    h = 0;
}
else {
    s = (max - min) / max;

    if (max == r) {
        h = 60 * ((g - b) / (max - min)) + 0;
    }
    else if (max == g) {
        h = 60 * ((b - r) / (max - min)) + 120;
    }
    else {
        h = 60 * ((r - g) / (max - min)) + 240;
    }
}

if (h < 0) h += 360.0f;

dst_h = (unsigned char)(h / 2); // dst_h : 0-180
dst_s = (unsigned char)(s * 255); // dst_s : 0-255
dst_v = (unsigned char)(v * 255); // dst_v : 0-255
}

```

```

void hsv2rgb(const unsigned char &src_h, const unsigned char &src_s, const
unsigned char &src_v, unsigned char &dst_r, unsigned char &dst_g, unsigned
char &dst_b)
{
    float h = src_h * 2.0f; // 0-360
    float s = src_s / 255.0f; // 0.0-1.0
    float v = src_v / 255.0f; // 0.0-1.0

    float r, g, b; // 0.0-1.0

    int hi = (int)(h / 60.0f) % 6;
    float f = (h / 60.0f) - hi;
    float p = v * (1.0f - s);
    float q = v * (1.0f - s * f);
    float t = v * (1.0f - s * (1.0f - f));

    switch(hi) {
        case 0: r = v, g = t, b = p; break;
        case 1: r = q, g = v, b = p; break;
        case 2: r = p, g = v, b = t; break;
        case 3: r = p, g = q, b = v; break;
        case 4: r = t, g = p, b = v; break;
        case 5: r = v, g = p, b = q; break;
    }

    dst_r = (unsigned char)(r * 255); // dst_r : 0-255
    dst_g = (unsigned char)(g * 255); // dst_r : 0-255
    dst_b = (unsigned char)(b * 255); // dst_r : 0-255
}

```

Додаток 2 Повне вирішення Приклад 2

```

// file edit bmp image by HSV chanel
#define _CRT_SECURE_NO_DEPRECATED
#include <stdio.h>
#include <math.h>
#include <malloc.h>

```

```

#define min_f(a, b, c) (fminf(a, fminf(b, c)))
#define max_f(a, b, c) (fmaxf(a, fmaxf(b, c)))

const int BYTES_PER_PIXEL = 3; // red, green, & blue
const int FILE_HEADER_SIZE = 14;
const int INFO_HEADER_SIZE = 40;

void generateBitmapImage(unsigned char* image, int height, int width, char*
imageFileName);
unsigned char* createBitmapFileHeader(int height, int stride);
unsigned char* createBitmapInfoHeader(int height, int width);

void rgb2hsv(const unsigned char& src_r, const unsigned char& src_g, const
unsigned char& src_b, float& dst_h, float& dst_s, float& dst_v)
{
    float r = src_r / 255.0f;
    float g = src_g / 255.0f;
    float b = src_b / 255.0f;

    float h=0, s=0, v=0; // h:0-360.0, s:0.0-1.0, v:0.0-1.0

    float max = max_f(r, g, b);
    float min = min_f(r, g, b);

    v = max;

    if (max == 0.0f) {
        s = 0;
        h = 0;
    }
    else if (max - min == 0.0f) {
        s = 0;
        h = 0;
    }
    else {
        s = (max - min) / max;

```

```

if (max == r) {
h = 60 * ((g - b) / (max - min)) + 0;
}
else if (max == g) {
h = 60 * ((b - r) / (max - min)) + 120;
}
else {
h = 60 * ((r - g) / (max - min)) + 240;
}
}

if (h < 0) h += 360.0f;

dst_h = (unsigned char)(h); // dst_h : 0-180
dst_s = (unsigned char)(s * 100); // dst_s : 0-255
dst_v = (unsigned char)(v * 100); // dst_v : 0-255
}

void hsv2rgb( const float& src_h, const float& src_s, const float& src_v,
unsigned char& dst_r, unsigned char& dst_g, unsigned char& dst_b)
{
float h = src_h; // 0-360
float s = src_s / 100.0f; // 0.0-1.0
float v = src_v / 100.0f; // 0.0-1.0

if (h > 360) h = 360;
if (s > 1) s = 1;
if (v > 1) v = 1;
if (h < 0) h += 360;
if (s < 0) s = 0;
if (v < 0) v = 0;

float r=0, g=0, b=0; // 0.0-1.0

int hi = (int)(h / 60.0f) % 6;
float f = (h / 60.0f) - hi;

```

```

float p = v * (1.0f - s);
float q = v * (1.0f - s * f);
float t = v * (1.0f - s * (1.0f - f));

switch (hi) {
case 0: r = v, g = t, b = p; break;
case 1: r = q, g = v, b = p; break;
case 2: r = p, g = v, b = t; break;
case 3: r = p, g = q, b = v; break;
case 4: r = t, g = p, b = v; break;
case 5: r = v, g = p, b = q; break;
}

dst_r = (unsigned char)(r * 255); // dst_r : 0-255
dst_g = (unsigned char)(g * 255); // dst_r : 0-255
dst_b = (unsigned char)(b * 255); // dst_r : 0-255
}

//***** M A I N
//*****

int main()
{
    int width, height, padding, bitcount, size;
    int i, j;
    unsigned char* in_image = 0;
    unsigned char* image = 0;

    unsigned char info[54] = { 0 };
    char* imageInputFileName = (char*)"D:\\VIDEO\\test_1.bmp";
    char* imageFileName = (char*)"D:\\VIDEO\\test_2.bmp";

    unsigned char R_in, G_in, B_in, R_out, G_out, B_out;
    float H_out, S_out, V_out;
    float saturation_shift = -30;

    FILE* file = fopen(imageInputFileName, "rb");

```

```

if (!file)
return 0;

fread(info, 1, 54, file);
width = *(int*)(info + 18);
height = *(int*)(info + 22);
bitcount = *(int*)(info + 28);
size = ((width * bitcount + 31) / 32) * 4 * height;
padding = width % 4;
if (bitcount != 24) //this code works for 24-bit bitmap only
goto error;

in_image = (unsigned char*) malloc(size);
image = (unsigned char*) malloc(size);
fread(in_image, 1, size, file);

for (i = 0; i < height; i++) {
for (j = 0; j < width; j++) {

R_in = in_image[i * width * 3 + j * 3 + 0];
G_in = in_image[i * width * 3 + j * 3 + 1];
B_in = in_image[i * width * 3 + j * 3 + 2];

rgb2hsv(R_in, G_in, B_in, H_out, S_out, V_out);
hsv2rgb(H_out, (S_out + S_out * saturation_shift / 100), V_out, R_out, G_out,
B_out);

image[i * width * 3 + j * 3 + 0] = (unsigned char)R_out; //red
image[i * width * 3 + j * 3 + 1] = (unsigned char)G_out; //green
image[i * width * 3 + j * 3 + 2] = (unsigned char)B_out; //blue
}
}

generateBitmapImage((unsigned char*)image, height, width, imageFileName);
printf("Image generated!!");
return 0;
error:

```

```

printf("Image NOT generated!!!");
if (file) fclose(file);
if (in_image) free(in_image);
if (image) free(image);
return -1;
}

```

```

void generateBitmapImage(unsigned char* image, int height, int width, char*
imageFileName)
{
    int widthInBytes = width * BYTES_PER_PIXEL;

    unsigned char padding[3] = { 0, 0, 0 };
    int paddingSize = (4 - (widthInBytes % 4)) % 4;

    int stride = (widthInBytes)+paddingSize;

    FILE* imageFile = fopen(imageFileName, "wb");
    //FILE* imageFileIN =
    fopen("https://upload.wikimedia.org/wikipedia/commons/4/47/PNG_transparen
cy_demonstration_1.png", "rb");

    unsigned char* fileHeader = createBitmapFileHeader(height, stride);
    fwrite(fileHeader, 1, FILE_HEADER_SIZE, imageFile);

    unsigned char* infoHeader = createBitmapInfoHeader(height, width);
    fwrite(infoHeader, 1, INFO_HEADER_SIZE, imageFile);

    int i;
    for (i = 0; i < height; i++) {
        fwrite(image + (i * widthInBytes), BYTES_PER_PIXEL, width, imageFile);
        fwrite(padding, 1, paddingSize, imageFile);
    }

    fclose(imageFile);
}

```

```

unsigned char* createBitmapFileHeader(int height, int stride)
{
    int fileSize = FILE_HEADER_SIZE + INFO_HEADER_SIZE + (stride *
height);

    static unsigned char fileHeader[] = {
        0,0, // signature
        0,0,0,0, // image file size in bytes
        0,0,0,0, // reserved
        0,0,0,0, // start of pixel array
    };

    fileHeader[0] = (unsigned char)('B');
    fileHeader[1] = (unsigned char)('M');
    fileHeader[2] = (unsigned char)(fileSize);
    fileHeader[3] = (unsigned char)(fileSize >> 8);
    fileHeader[4] = (unsigned char)(fileSize >> 16);
    fileHeader[5] = (unsigned char)(fileSize >> 24);
    fileHeader[10] = (unsigned char)(FILE_HEADER_SIZE +
INFO_HEADER_SIZE);

    return fileHeader;
}

```

```

unsigned char* createBitmapInfoHeader(int height, int width)
{
    static unsigned char infoHeader[] = {
        0,0,0,0, // header size
        0,0,0,0, // image width
        0,0,0,0, // image height
        0,0, // number of color planes
        0,0, // bits per pixel
        0,0,0,0, // compression
        0,0,0,0, // image size
        0,0,0,0, // horizontal resolution
        0,0,0,0, // vertical resolution
    }
}

```

```

0,0,0,0, /// colors in color table
0,0,0,0, /// important color count
};

```

```

infoHeader[0] = (unsigned char)(INFO_HEADER_SIZE);
infoHeader[4] = (unsigned char)(width);
infoHeader[5] = (unsigned char)(width >> 8);
infoHeader[6] = (unsigned char)(width >> 16);
infoHeader[7] = (unsigned char)(width >> 24);
infoHeader[8] = (unsigned char)(height);
infoHeader[9] = (unsigned char)(height >> 8);
infoHeader[10] = (unsigned char)(height >> 16);
infoHeader[11] = (unsigned char)(height >> 24);
infoHeader[12] = (unsigned char)(1);
infoHeader[14] = (unsigned char)(BYTES_PER_PIXEL * 8);

```

```

return infoHeader;

```

```

}

```

Додаток 3 rgb2hsv.v

Реалізація мовою Verilog модулю

```

module rgb2hsv(
    input wire [9:0] R,
    input wire [9:0] G,
    input wire [9:0] B,
    input wire CLK,

    output reg [24:0] H, // 0 - 360
    output reg [17:0] S, // 0 - 1 float 2_16
    output reg [17:0] V // 0 - 1 float 2_16
);

```

```

wire [2:0] rgb_se;
wire [7:0] RO, GO, BO, D, MAX, d_del;
wire [24:0] hue;
wire [17:0] ro, go, bo, top, max,min, max_min, saturation, value;

```

```

wire [31:0] hdivisor, sdivisor;
reg [31:0] cutoffhmult;

reg [71:0] hmult;
reg [63:0] smult;
reg [31:0] hr;

wire [2:0] rgb_se_del; //direction

rgb_divider RGB_D_(
.R ( R[9:2] ), // [7:0]
.G ( G[9:2] ), // [7:0]
.B ( B[9:2] ), // [7:0]
.CLK ( CLK ),

.RO ( RO ), // [7:0]
.GO ( GO ), // [7:0]
.BO ( BO ), // [7:0]
.ro ( ro ), // [17:0] [0..1] in 2.16 fixed point
.go ( go ), // [17:0] [0..1] in 2.16 fixed point
.bo ( bo ) // [17:0] [0..1] in 2.16 fixed point
);
// 1 CLK

max_min_selector MAX_MIN_SEL_(
.R ( RO ), // [7:0]
.G ( GO ), // [7:0]
.B ( BO ), // [7:0]
.r ( ro ), // [17:0] [0..1] in 2.16 fixed point
.g ( go ), // [17:0] [0..1] in 2.16 fixed point
.b ( bo ), // [17:0] [0..1] in 2.16 fixed point
.CLK ( CLK ),

.max_min ( max_min ), // [17:0] [0..1] in 2.16 fixed point
.max ( max ), // [17:0] [0..1] in 2.16 fixed point
.min ( min ), // [17:0] [0..1] in 2.16 fixed point
.D ( D ), // [7:0]

```

```

.MAX ( MAX ), // [7:0]
.TOP ( top ), // [17:0] [0..1] in 2.16 fixed point
.RGB_SE ( rgb_se ) // [2:0]
);
// 1 CLK

// Instantiate the LUT of (60/D) values, D = max - min 255 .. 0
reciprocallUT DIV_D_ (
.in ( D ),
.out ( hdivisor )
);

// Instantiate the LUT of (1/max) values, max 255 .. 0
divbymaxLUT DIV_MAX_(
.in ( MAX ),
.out ( sdivisor )
);

reg [17:0] t, r;
reg [31:0] hdiv, sdiv;

always@( posedge CLK )
begin

    t <= top;
    r <= max_min;
    hdiv <= hdivisor;
    sdiv <= sdivisor;

    hmult <= { 14'h0, t } * hdiv;
    smult <= { 14'h0, r } * sdiv;

end
// 2 CLK

Del #( .delay ( 2 ), .d_width ( 8 ) ) //2
D_DEL_(

```

```
.in ( D ),
.en ( 1'b1 ),
.out ( d_del ),
.clk ( CLK )
);
```

```
Del #( .delay ( 3 ), .d_width ( 3 ) ) //2
SE_DEL_(
.in ( rgb_se ),
.en ( 1'b1 ),
.out ( rgb_se_del ),
.clk ( CLK )
);
```

```
Del #( .delay ( 2 ), .d_width ( 18 ) ) //1
S_DEL_(
.in ( smult[33:16] ),
.en ( 1'b1 ),
.out ( saturation ),
.clk ( CLK )
);
```

```
Del #( .delay ( 4 ), .d_width ( 18 ) ) //3
V_DEL_(
.in ( max ),
.en ( 1'b1 ),
.out ( value ),
.clk ( CLK )
);
```

```
assign hue = hr[24:0];
```

```
// + - 120 240 360
always @ (posedge CLK)
begin
```

```
cutoffhmult <= ( d_del > 8'h0 ) ? { 9'h0, hmult[38:16] } : 32'hF00000;
```

```

case (rgb_se_del)

3'b000:
    //b g r
    hr <= 32'hF00000 - cutoffhmult;//-

3'b001:
    //g b r
    hr <= 32'h780000 + cutoffhmult;//+

3'b011:
    //g r b
    hr <= 32'h780000 - cutoffhmult;//-

3'b100:
    //b r g
    hr <= 32'hF00000 + cutoffhmult;//+

3'b110:
    //r b g
    hr <= 32'h1680000 - cutoffhmult;//-

3'b111:
    //r g b
    hr <= cutoffhmult;//+

default:
    hr <= 32'h0;

endcase

H <= hue;
S <= saturation;
V <= value;

end
// 2 CLK

```

endmodule

Додаток 4

Реалізація мовою Verilog модулю **hsv2rgb.v**

```
module hsv2rgb(  
input wire [24:0]      H, // [24:0] [0..360] in 8.16 fixed point  
input wire [17:0]      S, // [17:0] [0..1] in 2.16 fixed point  
input wire [17:0]      V, // [17:0] [0..1] in 2.16 fixed point  
input wire             CLK,  
  
output reg [9:0]       R,  
output reg [9:0]       G,  
output reg [9:0]       B  
);  
  
wire [17:0] saturation, value, rw, gw, bw, fract;  
wire [2:0] h_del;  
wire [24:0] hinw;  
  
reg [34:0] hdiv_60;  
reg [34:0] hue_1;  
reg [26:0] hue_2;  
reg [17:0] v, s;  
reg [17:0] r_tmp, g_tmp, b_tmp;  
reg [24:0] hin, hue;  
reg [63:0] p_tmp, q_tmp, t_tmp;  
  
reg [63:0] smultf,  
           smultominf;  
  
reg [17:0] omins,  
           ominf,  
           ominsmultf,  
           ominsmultominf;  
  
wire [17:0] omins_del, ominsmultf_del;
```

```

Del #( .delay ( 4 ), .d_width ( 3 ) )
H_DEL_(
.in ( hdiv_60[34:32]),
.en ( 1'b1 ),
.out ( h_del ),
.clk ( CLK )
);

```

```

Del #( .delay ( 4 ), .d_width ( 18 ) )
S_DEL_(
.in ( S ),
.en ( 1'b1 ),
.out ( saturation ),
.clk ( CLK )
);

```

```

Del #( .delay ( 7 ), .d_width ( 18 ) )
V_DEL_(
.in ( V ),
.en ( 1'b1 ),
.out ( value ),
.clk ( CLK )
);

```

```

Del #( .delay ( 2 ), .d_width ( 18 ) )
omins_DEL_(
.in ( omins ),
.en ( 1'b1 ),
.out ( omins_del ),
.clk ( CLK )
);

```

```

Del #( .delay ( 1 ), .d_width ( 18 ) )
ominsmultf_DEL_(
.in ( ominsmultf ),
.en ( 1'b1 ),
.out ( ominsmultf_del ),

```

```

.clk ( CLK )
);

assign hinw = hin >= 25'h1680000 ? 25'h0 : hin ;
assign fract = { 2'h0, hdiv_60[31:16] };

always@( posedge CLK )
begin

    hin <= H;
    hue <= hinw;

    hue_1 <= {hue,10'h0} + {hue,6'h0}; // hue * 1024 + hue*64 + hue*4 =
    hue*1092 ~ hue/60
    hue_2 <= {hue,2'h0};

    hdiv_60 <= hue_1 + hue_2;

    omins                <= 18'h10000 - saturation;
    ominf                <= 18'h10000 - fract;
    smultf                <= {14'h0,saturation} * {14'h0,fract};
    s                    <= saturation;

    ominsmultf            <= 18'h10000 - smultf[33:16];
    smultominf            <= {14'h0,s} * {14'h0,ominf};

    ominsmultominf <= 18'h10000 - smultominf[33:16];

    p_tmp                <= {14'h0,value} * {14'h0,omin_del};
    q_tmp                <= {14'h0,value} * {14'h0,ominsmultf_del};
    t_tmp                <= {14'h0,value} * {14'h0,ominsmultominf};
    v                    <= value;

end

assign rw = r_tmp[16] == 1'h1 ? r_tmp[17:0] - 18'h1 : r_tmp[17:0];
assign gw = g_tmp[16] == 1'h1 ? g_tmp[17:0] - 18'h1 : g_tmp[17:0];

```

```
assign bw = b_tmp[16] == 1'h1 ? b_tmp[17:0] - 18'h1 : b_tmp[17:0];
```

```
always@(posedge CLK)
```

```
begin
```

```
    if( h_del == 3'h0 )
```

```
    begin
```

```
        r_tmp <= v;
```

```
        g_tmp <= t_tmp[33:16];
```

```
        b_tmp <= p_tmp[33:16];
```

```
    end
```

```
    else
```

```
        if( h_del == 3'h1 )
```

```
        begin
```

```
            r_tmp <= q_tmp[33:16];
```

```
            g_tmp <= v;
```

```
            b_tmp <= p_tmp[33:16];
```

```
        end
```

```
    else
```

```
        if( h_del == 3'h2 )
```

```
        begin
```

```
            r_tmp <= p_tmp[33:16];
```

```
            g_tmp <= v;
```

```
            b_tmp <= t_tmp[33:16];
```

```
        end
```

```
    else
```

```
        if( h_del == 3'h3 )
```

```
        begin
```

```
            r_tmp <= p_tmp[33:16];
```

```
            g_tmp <= q_tmp[33:16];
```

```
            b_tmp <= v;
```

```
        end
```

```
    else
```

```
        if( h_del == 3'h4 )
```

```
        begin
```

```
            r_tmp <= t_tmp[33:16];
```

```
            g_tmp <= p_tmp[33:16];
```

```

    b_tmp <= v;
    end

                                else
                                if( h_del == 3'h5 )

    begin
    r_tmp <= v;
    g_tmp <= p_tmp[33:16];
    b_tmp <= q_tmp[33:16];
    end

    R <= {rw[15:8],2'h0};
    G <= {gw[15:8],2'h0};
    B <= {bw[15:8],2'h0};
end
endmodule

```

Література

1. K.N. Plataniotis and A.N. Venetsanopoulos, “Color Image Processing and Applications”, SpringerVerlag, February 18, 2000.
2. https://haralick.org/DV/understanding_color_models.pdf - The RYB, CMYK, RGB, HSV and Hexadecimal Color Models
3. https://www.ferrum.edu/downloads/academics/programs/school_of_art_divisions/dept_visual/art-color-systems-handbook.pdf

Інформаційні ресурси в Інтернеті

1. <https://www.rapidtables.com/convert/color/rgb-to-hsv.html> RGB to HSV color conversion

2. <https://www.rapidtables.com/convert/color/hsv-to-rgb.html> HSV to RGB color conversion

3.

https://learn.ztu.edu.ua/pluginfile.php/233175/mod_resource/content/0/%D0%9C3%D0%9B5%20-%D0%9F%D1%80%D0%B5%D0%B4%D1%81%D1%82%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%BD%D1%8F%20%D0%BA%D0%BE%D0%BB%D1%8C%D0%BE%D1%80%D1%83.%20RGB%20%D1%82%D0%B0%20HSV.pdf

Вивчення основних параметрів діодного екрана

Мета заняття:

- 1 Отримати практичні навички розрахунку параметрів світлодіодного екрану.
- 2 Навчитись створювати файл налаштувань світлодіодного екрану.
- 3 Засвоєння на практиці регулювання параметрів відображення світлодіодного екрану.

Основні питання заняття:

- 1 Основні параметри світлодіодного екрану.
- 2 Структура файлу налаштувань світлодіодного екрану.
- 3 Налаштування регістрів драйверу світлодіодного екрану.

Короткі теоретичні відомості

Світлодіодний екран являє собою площину для відтворення динамічного або статичного зображення за допомогою перетворення струму, що протікає через світлодіод в світлове випромінювання з відповідною довжиною хвилі.

LED екран складається з трьох основних частин:

Джерела живлення – для створення потрібної напруги або напруг живлення. Як правило блоки працюють з вихідною напругою 3,8-5В при вихідній потужності 200-350Вт.

Плати керування на основі FPGA – для розподілення інформації і керування відображенням.

LED модулів – котрі служать для відтворення зображення шляхом перетворення двійкового коду кольору в світлове випромінювання LED.

Відображення на LED екрані відбувається методом динамічного сканування, рисунок 10.1.

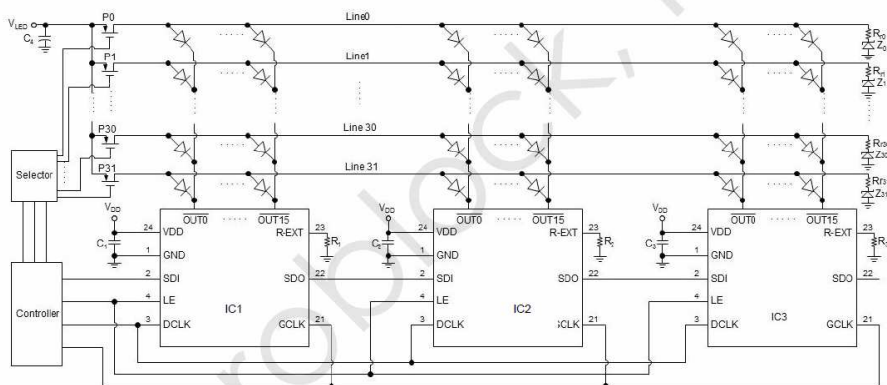


Рисунок 10.1 – Схема LED модуля динамічного сканування з включенням LED зі спільним анодом.

В світлодіодному екрані можуть використовуватися драйвери і ключі як окремі мікросхеми, так і драйвер із вбудованими ключами, рисунок 10.2.

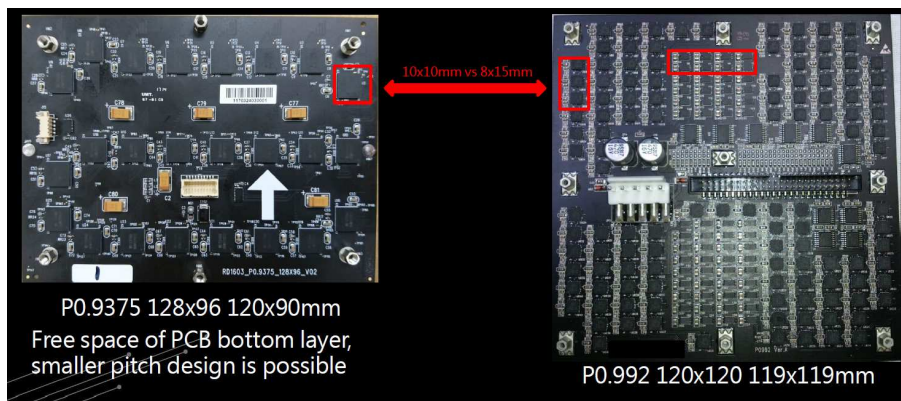


Рисунок 10.2 – LED модулі з окремими драйверами та ключами та з вбудованими ключами в драйвер (зліва). Ключі та драйвери виділено червоним.

Частота оновлення екрану (Refresh rate) визначається як:

$$R_r = F_f \times N_s$$

Де

R_r – частота оновлення екрану, Refresh rate;

F_f – кадрова частота, Frame Frequency;

N_s – кількість сегментів сканування;

Глибина кольору, бітність екрану визначається як:

$$G_s = \log_2 (N_{GCK} \times N_s)$$

Де

G_s – бітність екрану, Gray Scale;

N_{GCK} – кількість імпульсів $GCLK$ в сегменті сканування;

N_s – кількість сегментів сканування;

Частота запису даних $DCLK$ залежить від схемотехніки самого модуля. Впливає кількість драйверів поєднаних в послідовний ланцюг, адже дані записуються послідовно за допомогою зсувних регістрів. Враховується кількість ліній сканування, під'єднаних до виходу драйвера.

$$F_{DCLK} = \frac{1}{(256 + (16 \times Nd_{out})) \times N_{dr} \times L_n \times F_f}$$

Де

Nd_{out} – кількість задіяних виходів драйвера світлодіодів;

N_{dr} – кількість драйверів в ланцюгу даних;

L_n – кількість ліній сканування (Scan);

F_f – кадрова частота, Frame Frequency;

Частота відтворення $GCLK$ – GrayScale CLK, - буде залежати від заданих режимів відображення глибини кольору, частоти оновлення екрану та кадрової частоти екрану.

$$F_{GCLK} = \frac{1}{(((T_{dl} + T_{dh} + N_{GCK} + N_{GCKadd}) \times L_n \times N_s)) \times F_f}$$

Де

T_{dl} – час розряду;

T_{dl} – час заряду;

N_{GCLK} – кількість імпульсів $GCLK$ в сегменті сканування;

$N_{GCLKadd}$ – кількість додаткових імпульсів $GCLK$ в сегменті сканування;

L_n – кількість ліній сканування (Scan);

N_s – кількість сегментів сканування;

F_f – кадрова частота, Frame Frequency;

Для кожного діодного екрану, створюється індивідуальний файл налаштувань, який містить набір даних необхідний для коректної роботи як плати керування, так і драйвера екрану. Файл налаштувань складається з наступних полів:

[Manufacturer] – містить інформацію про виробника екрану

[Additional] – додаткова інформація про екран (для внутрішнього/зовнішнього використання, наявність температурного давача, додаткового охолодження тощо).

[Panel] – дані про особливості будови екрану (фізичний розмір, кількість модулів, кількість пікселів в модулі, глибину кольору, частоту оновлення тощо).

[Module] – дані описують будову модуля та основні параметри налаштувань (кількість ліній сканування L_n , кількість сегментів сканування N_s , кількість імпульсів $GCLK$ в сегменті сканування N_{GCLK} , частоти $GCLK$ $DCLK$).

[Driver] – дані які стосуються мікросхеми драйвера екрану.

[LEDAddressMap] – дані адресації пікселів зображення, відповідним виводам мікросхеми драйвера, створюється відповідно до схемотехніки модуля.

[DefaultCIE], [DefaultLuma], [ThermalGradient], [PrefixGamma], [ILED] – додаткові, необов'язкові поля.

Розбиття файлу налаштувань на окремі поля – умовне, дані з кожного поля взаємопов'язані і використовуються програмою керування, платою керування і драйвером екрану.

Приклад 1. Визначити частоту оновлення екрану R_r та бітність, якщо екрана працює з частотою оновлення $F_f = 60 \text{ Гц}$, кількість сегментів у кадрі вибрано $N_s = 64$, довжина кожного сегмента $N_{GCK} = 512$

Розв'язок: частота оновлення екрану визначається, як добуток кадрової частоти на кількість сегментів сканування в одному кадрі. Як правило кількість сегментів в кожному кадрі одна і та ж. Тоді

$$R_r = F_f \times N_s = 60 \text{ Гц} \times 64 = 3840 \text{ Гц}$$

$$G_s = \log_2 (N_{GCK} \times N_s) = \log_2 (512 \times 64) = 15 \text{ біт}$$

Приклад 2. Визначити частоти $GCLK$ та $DCLK$, якщо екрана працює з частотою оновлення $F_f = 120 \text{ Гц}$. Глибина кольору 16 біт - кількість сегментів у кадрі вибрано $N_s = 64$, кількість імпульсів $GCLK$ в сегменті сканування $N_{GCK} = 1024$, $N_{GCKadd} = 32$, кількість ліній сканування, підключених до виходу драйвера $L_n = 27$, час заряду T_{dl} та час розряду T_{dl} рівний 64 імпульси $GCLK$, драйвер має 16 виходів Nd_{out} .

Розв'язок: за наведеними вище формулами розраховуємо потрібні частоти $GCLK$ та $DCLK$.

$$F_{DCLK} = \frac{1}{(256 + (16 \times Nd_{out})) \times N_{dr} \times L_n \times F_f} =$$

$$= \frac{1}{(256 + (16 \times 16)) \times 8 \times 27 \times 120} = 13\,271\,040 = 13,27104 \text{ МГц}$$

$$F_{GCLK} = \frac{1}{(((T_{dl} + T_{dh} + N_{GCK} + N_{GCKadd}) \times L_n \times N_s)) \times F_f} =$$

$$= \frac{1}{(((64 + 64 + 1024 + 32) \times 27 \times 64)) \times 120} = 245\,514\,240 = 245,5$$

Рисунок 9.1

Хід заняття

1. Опрацювати лекційний матеріал за темою заняття.
2. Ознайомитися з описом драйвера MBI5754 (Додаток 2) файли:
- MBI5754 Application Note V2.00 - EN_0325.pdf
- MBI5754 Advance Information_V0.03. pdf.
3. Вивчити структуру файлу налаштувань – test.ini (Додаток 1), порівняти з файлом опису драйвера.
4. Розрахувати параметри екрану і драйвера (задачі 1,2,3.).
5. Виконати контроль параметрів (задача 4).
6. Результати роботи представити у вигляді звіту.

Задача 1. На прикладі драйвера MBI5754, визначити бітність G_s та частоту оновлення екрану R_r використовуючи данні таблиці 1, де вказано апаратні режими роботи драйвера. Варіанти завдань наведено у таблиці 10.1.

Таблиця 10.1.

Номер варіанта	F_f , Гц	$N_{\text{ГСК}}$	N_s
1	30	1024	64
2	48	512	128
3	50	1024	32

4	60	512	64
5	100	512	32
6	120	256	64
7	144	128	64
8	240	256	32

Задача 2. Визначити частоту DCLK використовуючи данні таблиці 2, Варіанти завдань наведено у таблиці 10.2.

Таблиця 10.2.

Номер варіанта	$N_{d_{out\ r}}$	N_{dr}	L_n	F_f
1	16	12	60	30
2	16	16	54	48
3	12	16	30	50
4	16	12	27	60
5	15	8	24	100
6	12	6	24	120
7	15	5	16	144
8	16	4	12	240

Задача 3. На прикладі драйвера MBI5754 визначити частоту GCLK використовуючи данні таблиці 3, де вказано налаштування драйвера. Варіанти завдань наведено у таблиці 10.3.

Таблиця 3.

Номер варіанта	T_{dl}	T_{dl}	N_{GCLK}	N_s	$N_{GCLKadd}$	L_n	F_f
1	8	8	1024	64	32	60	30
2	32	32	512	128	28	54	48
3	16	16	1024	32	8	30	50
4	4	4	512	64	16	27	60
5	8	8	512	32	24	24	100
6	24	24	256	64	2	24	120
7	12	12	128	64	8	16	144
8	5	5	256	32	10	12	240

Задача 4. Програмно налаштуйте драйвер, використовуючи отримані результати відповідно варіантів по таблицям 1-3.

Для чого, в файлі test.ini (Додаток 1) заповнити відповідні поля по наведеному нижче прикладу:

P_FrameRate - частота оновлення екрану, Refresh rate;

P_BitProcessing - бітність екрану, Gray Scale;

M_PhysDriverNumber - кількість драйверів в ланцюгу даних;

M_LineScanNumber - кількість ліній сканування (Scan);

M_SegmentNumber - кількість сегментів сканування;

M_SegmentLength - кількість імпульсів GCLK в сегменті сканування;

M_GCLKSet – частота $GCLK \geq GCLK$ розрахункова;

M_DCLKSet – частота $DCLK \geq DCLK$ розрахункова;

За допомогою осцилографа стати на контрольну точку TP2 контролера відео модуля, регулюванням параметру M_LineScanDead60 для 30/60/120/144/240 Гц та M_LineScanDead50 для 48/50/100 Гц встановити довжину імпульса від 10 до 50 мкс. Переконатися що зображення відображається правильно, без артефактів.

Контрольні питання

1. Які з основних параметрів драйвера, визначають глибину кольору та частоту оновлення екрану?
2. Як зміниться частота оновлення екрану R_f при зміні кадрової частоти f з 60 Гц на 120 Гц?
3. Що, окрім схемотехніки діодного модуля, впливає на значення частоти запису даних $DCLK$?

Додаток 1. Test.ini

;12_04_2024

; 1.588 mm Pixel Pitch

;V0.1_15bit_3840_60Hz_Test

;-----

[Manufacturer]

Manufacturer_Text = TruePerformance

[Additional]

A_PossibilityTrunc = 0

A_Name = 1 ; DP1.588

A_Outdoor = 1

A_NoFan = 1
A_TemperatureSensor = 0
A_ExtendedMakeweight = 1

[Panel]

P_Type = 2 ; 2- DP1.58
P_ModulesInX = 2
P_ModulesInY = 2
P_PixelsInModuleX = 192 ; 96
P_PixelsInModuleY = 108 ; 108
P_SizeInXmm = 610
P_SizeInYmm = 343
P_FrameRate = 3840
P_BitProcessing = 16
P_BitProcessingGrowth = 128 ; 128 - Disable, 0 - 15 Bit, 1 - 14 Bit, 2 - 13 Bit, 3 - 12 Bit
P_ColorSwap = RGB
P_ROMLocation = NONE;
0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15
P_GroupSwap =
1/4,2/12,3/5,4/13,5/6,6/14,7/7,8/15,9/0,10/8,11/1,12/9,13/2,14/10,15/3,16/1
1,17/20,18/28,19/21,20/29,21/22,22/30,23/23,24/31,25/16,26/24,27/17,28/2
5,29/18,30/26,31/19,32/27
P_GroupZigzag = 0
P_AnodeSwitchType = 1 ; 0 - MBI5908/ 1- RT5958
P_Autonomous0 = FF01
P_Autonomous1 = 0x0001
;P_MicroControllerInfo = 0033 ; 0011 - , 0021 - only EQ ,
0031 - lock
P_VCalOnModule = 0
P_TCalOnModule = 0
P_BCalOnModule = 0
P_DeliveryLock = 1
P_CalibrationLock = 1
P_NumbOfTables = 1

P_PossibilityTrunc = 0
P_Text = DP1.588

[Module]

M_Type = 2 ; 2
M_DataDriverMask = 0000 ; GrayScale Shift
M_PhysDriverNumber = 6 ; Complementary Code
M_LineScanNumber = 27 ; Value-1
M_SegmentNumber = 64 ; Value-1
M_SegmentLength = 512 ; Value*8-1 (GCLK Number per Segment; /2 for D_GCLKMultiplier = 1)
M_LineScanDead60 = 449 ; 632=9mks 295 562/2 for D_DblRefreshRate = 1
M_LineScanDead50 = 449 ; 632=10mks 562/2 for D_DblRefreshRate = 1
M_FirstGCLKExpander = 122 ;
M_FirstGCLKExpander18= 0
M_GCLKSet = 12 ; 0 - 8.9MHz, 1 - 9.9MHz, 2 - 10.8MHz, 3 - 12.9MHz, 4 - 13.9MHz, 5 - 14.9MHz, 6 - 16MHz, 7 - 16.9MHz, 8 - 17.5MHz, 9 - 18.5MHz, 10 - 19.5MHz, 11 - 95MHz, 12 - 100MHz
M_DCLKSet = 9 ; 0 - 4.5MHz, 1 - 5.5MHz, 2 - 6.5MHz, 3 - 7MHz, 4 - 7.625MHz, 5 - 8MHz, 6 - 8.714MHz, 7 - 9.5MHz, 8 - 19(9.5)MHz, 9 - 10(5)MHz

M_Text = _V0.1

[Driver]

;D_CurrentRed18 = 30 ; Hex (21)
;D_CurrentGreen18 = 30 ; Hex (17)
;D_CurrentBlue18 = 28 ; Hex (11)

D_Type = 6 ; 6 (MBI5754)
D_DefaultReg1Red = 9BBF ; 9F8E MBI5754 (Application Note Recomendend)

D_DefaultReg1Green Application Note Recomend)	= 9BB2	; 9F8C MBI5754 (
D_DefaultReg1Blue Application Note Recomend)	= 9BA6	; 9F87 MBI5754 (
D_DefaultReg2Red Application Note Recomend) 0601	= 0601	; MBI5754 (
D_DefaultReg2Green Application Note Recomend) 0601	= 0601	; MBI5754 (
D_DefaultReg2Blue Application Note Recomend) 0601	= 0601	; MBI5754 (
D_DefaultReg3Red Programming Guide Recomend)	= 8420	; MBI5754 (
D_DefaultReg3Green Programming Guide Recomend)	= 90A5	; MBI5754 (
D_DefaultReg3Blue Programming Guide Recomend)A561	= 90A1	; MBI5754 (
D_DefaultReg4Red Programming Guide Recomend)	= A800	; MBI5754 (
D_DefaultReg4Green Programming Guide Recomend)	= BC00	; MBI5754 (
D_DefaultReg4Blue Programming Guide Recomend)	= D800	; MBI5754 (
D_DefaultReg5Red Programming Guide Recomend)	= 7A01	; MBI5754 (
D_DefaultReg5Green Programming Guide Recomend)7A01	= 7A01	; MBI5754 (
D_DefaultReg5Blue Programming Guide Recomend)	= 7A01	; MBI5754 (
D_DefaultReg6Red Programming Guide Recomend) 4000	= 4000	; MBI5754 (
D_DefaultReg6Green Programming Guide Recomend)	= 4000	; MBI5754 (

D_DefaultReg6Blue Programming Guide Recomendend)	= 4000	; MBI5754 (
D_DefaultReg7Red Programming Guide Recomendend)	= 8800	; MBI5754 (
D_DefaultReg7Green Programming Guide Recomendend)	= 8400	; MBI5754 (
D_DefaultReg7Blue Programming Guide Recomendend)	= 8E00	; MBI5754 (
D_DefaultReg8Red Programming Guide Recomendend) 0040	= 0022	; MBI5754 (
D_DefaultReg8Green Programming Guide Recomendend) 0040	= 0022	; MBI5754 (
D_DefaultReg8Blue Programming Guide Recomendend) 0040	= 0022	; MBI5754 (
D_DefaultReg9Red Programming Guide Recomendend)	= 0131	; MBI5754 (
D_DefaultReg9Green Programming Guide Recomendend)	= 0131	; MBI5754 (
D_DefaultReg9Blue Programming Guide Recomendend)	= 0131	; MBI5754 (
D_DefaultReg10Red Programming Guide Recomendend)	= 0000	; MBI5754 (
D_DefaultReg10Green Programming Guide Recomendend)	= 0000	; MBI5754 (
D_DefaultReg10Blue Programming Guide Recomendend)	= 0000	; MBI5754 (
D_DefaultReg11Red MBI5754 (Programming Guide Recomendend) 045F	= 047D	; 045F 1250ns
D_DefaultReg11Green MBI5754 (Programming Guide Recomendend)	= 047D	; 045F 1250ns
D_DefaultReg11Blue MBI5754 (Programming Guide Recomendend)	= 047D	; 045F 1250ns

D_DefaultReg12Red	= 08AF	; 0097 1750ns
MBI5754 (Programming Guide Recomend)		
D_DefaultReg12Green	= 08AF	; 0097 1750ns
MBI5754 (Programming Guide Recomend)		
D_DefaultReg12Blue	= 08AF	; 0097 1750ns
MBI5754 (Programming Guide Recomend)		
D_DefaultReg13Red	= 301B	; MBI5754 (
Programming Guide Recomend)		
D_DefaultReg13Green	= 301B	; MBI5754 (
Programming Guide Recomend)		
D_DefaultReg13Blue	= 301B	; MBI5754 (
Programming Guide Recomend)		
D_DefaultReg14Red	= 8051	; MBI5754 (
Programming Guide Recomend) 8050		
D_DefaultReg14Green	= 8051	; MBI5754 (
Programming Guide Recomend)		
D_DefaultReg14Blue	= 8051	; MBI5754 (
Programming Guide Recomend)		
D_DefaultReg15Red	= 0000	; MBI5754 (
Programming Guide Recomend)		
D_DefaultReg15Green	= 0000	; MBI5754 (
Programming Guide Recomend)		
D_DefaultReg15Blue	= 0000	; MBI5754 (
Programming Guide Recomend)		
D_DefaultReg16Red	= 0000	; MBI5754 (
Programming Guide Recomend)		
D_DefaultReg16Green	= 0000	; MBI5754 (
Programming Guide Recomend)		
D_DefaultReg16Blue	= 0000	; MBI5754 (
Programming Guide Recomend)		

D_CurrentRed = 3F ; Hex (22)
 Ir=0,003934 A 6bit in Reg1[5:0] L=990 cd/m², T=6590K, x=0.3134,
 y=0.3142

D_CurrentGreen = 32 ; Hex (B)
 Ir=0,003919 A

D_CurrentBlue = 26 ; Hex (F)
 Ir=0,002443 A

D_GCLKChange = 2 ;

D_GCLKMultiplier = 0 ;

D_DblRefreshRate = 1 ;

D_GrayScaleMode = 1 ;

D_StripRemovalLowGreyRed = 0 ;

D_StripRemovalLowGreyGreen = 0 ;

D_StripRemovalLowGreyBlue = 0 ;

D_CrossEffectRemovalRed = 0 ;

D_CrossEffectRemovalGreen = 0 ;

D_CrossEffectRemovalBlue = 0 ;

D_CrossEffectRemovalLevelRed = 0 ;

D_CrossEffectRemovalLevelGreen = 0 ;

D_CrossEffectRemovalLevelBlue = 0 ;

D_LowGreyCompensRed = 0 ;

D_LowGreyCompensGreen = 0 ;

D_LowGreyCompensBlue = 0 ;

D_LowGreyCompensAdvance2Red = 0 ;

D_LowGreyCompensAdvance2Green = 0 ;

D_LowGreyCompensAdvance2Blue = 0 ;

D_LowGreyCompensAdvance1Red = 0 ;

D_LowGreyCompensAdvance1Green = 0 ;

D_LowGreyCompensAdvance1Blue = 0 ;

D_LowGreyCompensAdvance3Red = 0 ;
D_LowGreyCompensAdvance3Green = 0 ;
D_LowGreyCompensAdvance3Blue = 0 ;

D_DarkScreenCompens1Red = 1 ;
D_DarkScreenCompens1Green = 4 ;
D_DarkScreenCompens1Blue = 4 ;

D_DarkScreenCompens2Red = 1 ;
D_DarkScreenCompens2Green = 5 ;
D_DarkScreenCompens2Blue = 5 ;

D_DarkScreenCompens3Red = 0 ;
D_DarkScreenCompens3Green = 5 ;
D_DarkScreenCompens3Blue = 1 ;

D_DimLineCompensRed = A ;
D_DimLineCompensGreen = F ;
D_DimLineCompensBlue = 16 ;
D_LowerGhostEliminationRed = 1 ;
D_LowerGhostEliminationGreen = 1 ;
D_LowerGhostEliminationBlue = 1 ;

D_GradientCompensRed = 0 ;
D_GradientCompensGreen = 0 ;
D_GradientCompensBlue = 0 ;

D_GradientCompens1Red = 0 ;
D_GradientCompens1Green = 0 ;
D_GradientCompens1Blue = 0 ;

[LEDAAddressMap]
Begin

32	33	34	35	36	37	38
39	40	41	42	43	44	45
			46	47		
48	49	50	51	52	53	54
55	56	57	58	59	60	61
			62	63		
64	65	66	67	68	69	70
71	72	73	74	75	76	77
			78	79		
80	81	82	83	84	85	86
87	88	89	90	91	92	93
			94	95		
96	97	98	99	100	101	102
103	104	105	106	107	108	
		109	110	111		
112	113	114	115	116	117	1
18	119	120	121	122	123	124
		125	126	127		
160	161	162	163	164	165	1
66	167	168	169	170	171	172
		173	174	175		
176	177	178	179	180	181	1
82	183	184	185	186	187	188
		189	190	191		
192	193	194	195	196	197	1
98	199	200	201	202	203	204
		205	206	207		
208	209	210	211	212	213	2
14	215	216	217	218	219	220
		221	222	223		
224	225	226	227	228	229	2
30	231	232	233	234	235	236
		237	238	239		
240	241	242	243	244	245	2
46	247	248	249	250	251	252
		253	254	255		
288	289	290	291	292	293	2
94	295	296	297	298	299	300
		301	302	303		

	304	305	306	307	308	309	3
10	311	312	313	314	315	316	
		317	318	319			
	320	321	322	323	324	325	3
26	327	328	329	330	331	332	
		333	334	335			
	336	337	338	339	340	341	3
42	343	344	345	346	347	348	
		349	350	351			
	352	353	354	355	356	357	3
58	359	360	361	362	363	364	
		365	366	367			
	368	369	370	371	372	373	3
74	375	376	377	378	379	380	
		381	382	383			
	416	417	418	419	420	421	4
22	423	424	425	426	427	428	
		429	430	431			
	432	433	434	435	436	437	4
38	439	440	441	442	443	444	
		445	446	447			
	448	449	450	451	452	453	4
54	455	456	457	458	459	460	
		461	462	463			
	464	465	466	467	468	469	4
70	471	472	473	474	475	476	
		477	478	479			
	480	481	482	483	484	485	4
86	487	488	489	490	491	492	
		493	494	495			
	496	497	498	499	500	501	5
02	503	504	505	506	507	508	
		509	510	511			
	544	545	546	547	548	549	5
50	551	552	553	554	555	556	
		557	558	559			
	560	561	562	563	564	565	5
66	567	568	569	570	571	572	
		573	574	575			

	576	577	578	579	580	581	5
82	583	584	585	586	587	588	
		589	590	591			
	592	593	594	595	596	597	5
98	599	600	601	602	603	604	
		605	606	607			
	608	609	610	611	612	613	6
14	615	616	617	618	619	620	
		621	622	623			
	624	625	626	627	628	629	6
30	631	632	633	634	635	636	
		637	638	639			
	672	673	674	675	676	677	6
78	679	680	681	682	683	684	
		685	686	687			
	688	689	690	691	692	693	6
94	695	696	697	698	699	700	
		701	702	703			
	704	705	706	707	708	709	7
10	711	712	713	714	715	716	
		717	718	719			
	720	721	722	723	724	725	7
26	727	728	729	730	731	732	
		733	734	735			
	736	737	738	739	740	741	7
42	743	744	745	746	747	748	
		749	750	751			
	752	753	754	755	756	757	7
58	759	760	761	762	763	764	
		765	766	767			
	800	801	802	803	804	805	8
06	807	808	809	810	811	812	
		813	814	815			
	816	817	818	819	820	821	8
22	823	824	825	826	827	828	
		829	830	831			
	832	833	834	835	836	837	8
38	839	840	841	842	843	844	
		845	846	847			

	848	849	850	851	852	853	8
54	855	856	857	858	859	860	
		861	862	863			
	864	865	866	867	868	869	8
70	871	872	873	874	875	876	
		877	878	879			
	880	881	882	883	884	885	8
86	887	888	889	890	891	892	
		893	894	895			
	928	929	930	931	932	933	9
34	935	936	937	938	939	940	
		941	942	943			
	944	945	946	947	948	949	9
50	951	952	953	954	955	956	
		957	958	959			
	960	961	962	963	964	965	9
66	967	968	969	970	971	972	
		973	974	975			
	976	977	978	979	980	981	9
82	983	984	985	986	987	988	
		989	990	991			
	992	993	994	995	996	997	9
98	999	1000	1001	1002	1003		1
		004	1005	1006	1007		
			1008				
1009	1010	1011	1012	1013	1014		
1015	1016	1017	1018	1019	1020		
1021	1022	1023			End		

[DefaultCIE]

RX = 0.6922

RY = 0.3078

GX = 0.1628

GY = 0.7520

BX = 0.1296

BY = 0.0672

[DefaultLuma]

PrimR = 292

PrimG = 607

PrimB = 92

CaliR = 218

CaliG = 482

CaliB = 74

[ThermalGradient]

GradientMapR = Thermo_1.588_384x216-a.bmp

SpreadingR = 7.5

SpreadingG = 0.0

SpreadingB = 0.0

[PrefixGamma]

InStop = 32

Gamma = 1.51

DispersGamma = 0.20

Increase = 0

BT1886= 0

BlackLevel= 0

DynamicExp=1

[ILED]

I_ResistorRed = 1047

I_ResistorGreen = 1165

I_ResistorBlue = 1102

I_IoutRed = 20.63 mA

I_IoutGreen = 15.98 mA

I_IoutBlue = 14.39 mA

Додаток 2.

MBI5754 Application Note V2.00 - EN_0325.pdf -

https://drive.google.com/file/d/1noZMsRFRPC4z6pzGsy5hC_hy7GmIBexi/view?usp=drive_link

MBI5754 Advance Information_V0.03. pdf –

https://drive.google.com/file/d/10vNwxWgrvmzxCfu20Qm8KzvessXC7jIB/view?usp=drive_link

Лабораторна робота №11
**Дослідження основних параметрів відеосигналу в системі обробки
та відтворення зображень**

Мета заняття:

- 1 Вивчити основні параметри відеосигналу в системі обробки та відтворення зображень.
- 2 Ознайомитися з форматом стандарту даних EDID (Extended Display Identification Data).
- 3 Розрахувати та виміряти основні параметри відеосигналу.

Основні питання заняття:

- 1 Основні параметри відеосигналу.
- 2 Формат стандарту даних - Extended Display Identification Data (EDID)
- 3 Зміна параметрів файлу EDID.
- 4 Розрахунок та вимірювання параметрів відеосигналів згідно мети заняття

Короткі теоретичні відомості

Відео, також відеографія (від лат. video — дивлюся, бачу) — під цим терміном розуміють широкий спектр технологій запису, обробки, передачі, зберігання й відтворення візуального і аудіовізуального матеріалу на пристроях відображення (монітори, відео-проектори).

Відео характеризується такими параметрами:

Частота кадрів, кількість кадрів на секунду — це число нерухомих зображень, що послідовно змінюються одне за одним впродовж однієї секунди відеоматеріалу (рисунок 11.1), створюючи ефект руху об'єктів на екрані. Чим більша частота кадрів на секунду, тим плавнішим і природнішим буде здаватися рух (рисунок 11.2). Вимірюється у кадрах за секунду (fps) або герцах (Гц).

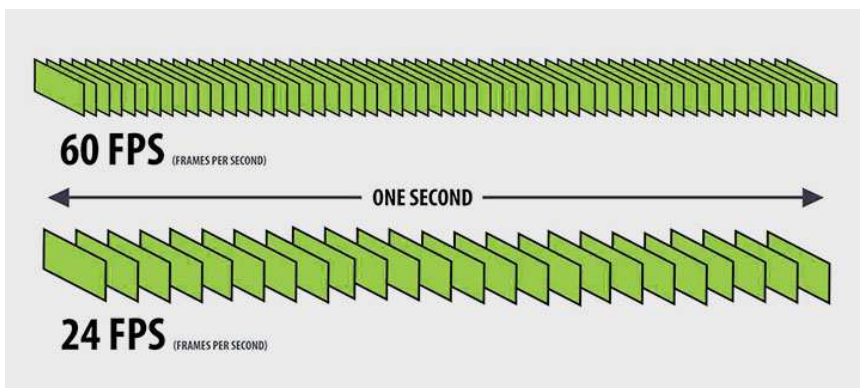


Рисунок 11.1 – Послідовність кадрів з частотою 60 Гц і 24 Гц

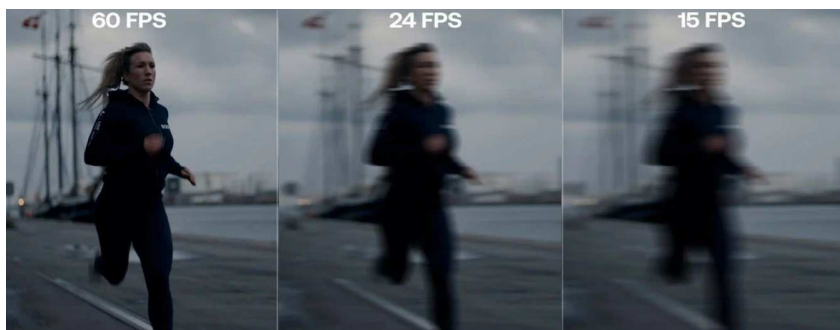


Рисунок 11.2 – Вплив кадрової частоти на плавність відео

Розгортка зображення – процес поелементного перетворення зображення у послідовність електричних сигналів за встановленим порядком з метою відтворення, запису або збереження розгорнутого зображення на спеціальних пристроях. Розгортка відеоматеріалу може бути прогресивною (англ. Progressive) або черезрядковою (англ. Interlaced, інтерлейс). При прогресивній розгортці всі горизонтальні лінії зображення (рядки) відображаються одночасно. При черезрядковій розгортці показуються по чергові парні й непарні рядки (поля кадру).

Роздільна здатність зображення – величина, що визначає кількість пікселів на одиницю площі або довжини (рис. 3). Будь-який

відеосигнал також має роздільну здатність (англ. Resolution, роздільність), горизонтальну і вертикальну, що вимірюється у пікселях.

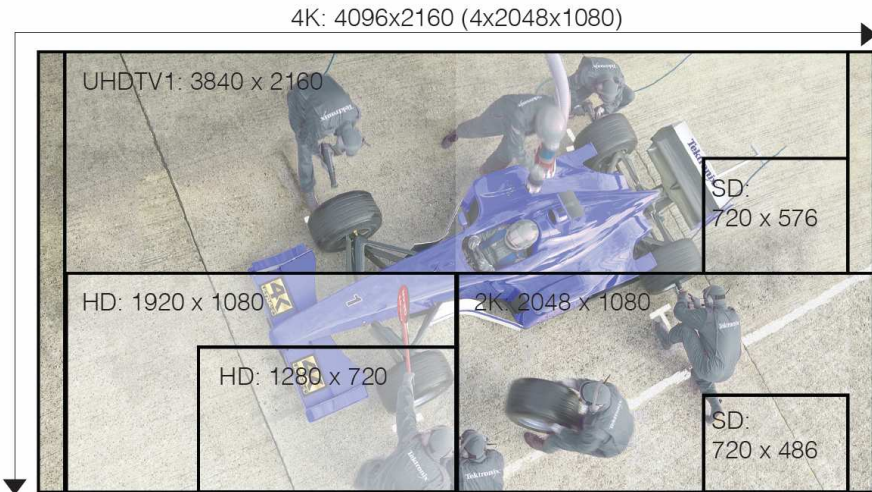


Рисунок 11.3 – Роздільна здатність зображення

Кількість кольорів і кольорова розрядність (глибина кольору)

– описується кольоровими моделями (YUV, YCbCr, RGB, HSV, CMYK). Кількість кольорів, що може бути відображена залежить від якості пристрою відображення. Людське око може сприйняти, по різних підрахунках, від 5 до 10 мільйонів відтінків кольорів.

Кількість кольорів у відеоматеріалі визначається числом бітів, відведеним для кодування кольору кожного пікселя (англ. *bitsperpixel*, *bpp*). 1 біт дозволяє закодувати 2 кольори (як правило, чорний і білий), 2 біти — 4 кольори, 3 біти — 8 кольорів, ..., 8 бітів — 256 кольорів ($2^8 = 256$), 16 бітів — 65 536 кольорів (2^{16}), 24 біти — 16 777 216 кольорів (2^{24}).

Ширина відеопотоку, або бітова швидкість (англ. *bitrate*) — це кількість оброблюваних бітів відеоінформації за секунду часу, і вимірюється у бітах на секунду (біт/с). Похідні від цієї одиниці —

кілобіт на секунду (кбіт/с) та мегабіт на секунду (мбіт/с). Чим вище ширина відеопотоку, тимкраща якістьвідео.

Для зв'язку, між пристроєм відображення і джерелом відеосигналу існує набір протоколів **DDC- Display data channel** (канал даних відображення), створений Асоціацією стандартів відеоелектроніки (VESA). Зокрема, такі інтерфейси для з'єднання цифрових пристроїв як DVI, HDMI, DisplayPort використовують розширені ідентифікаційні дані дисплея (англ. Extended Display Identification Data, EDID).

EDID - визначає компактний двійковий формат файлу, що описує можливості монітора (пристрою відображення) та підтримувані графічні режими, що зберігається в мікросхемі пам'яті (EEPROM), запрограмованому виробником монітора. Формат використовує блок опису, що містить 128 байт даних, з додатковими блоками розширення для надання додаткової інформації. Найновішою версією є Enhanced EDID (E-EDID) Release A v2.0. Він містить інформацію про виробника, максимальний розмір зображення, колірні характеристики, заводські встановлені таймінги, межі частотного діапазону, а також рядки, що містять назву монітора і серійний номер.

У даній лабораторній роботі, дослідження параметрів відеосигналу проводяться наобладнанні – система обробки і відображення ERMAC16 (рисунок 11.4). Характеристики даного відеоконтролера наведені в додатку 1.

Для керування відеоконтролером ERMAC16 використовується програмне забезпечення “EKTA Screen Console”. Інтерфейс програми зображено на рисунку 11.5, в вкладці “Input Switch” (вікно контролю вхідних сигналів) в полі “Info” відображається інформація про вставні картки (тип картки, дата проекту, версія), наявність відесигналів та їх параметри (роздільна здатність та частота кадрів).



Рисунок 11.4 –Відеоконтролер ERMAC16

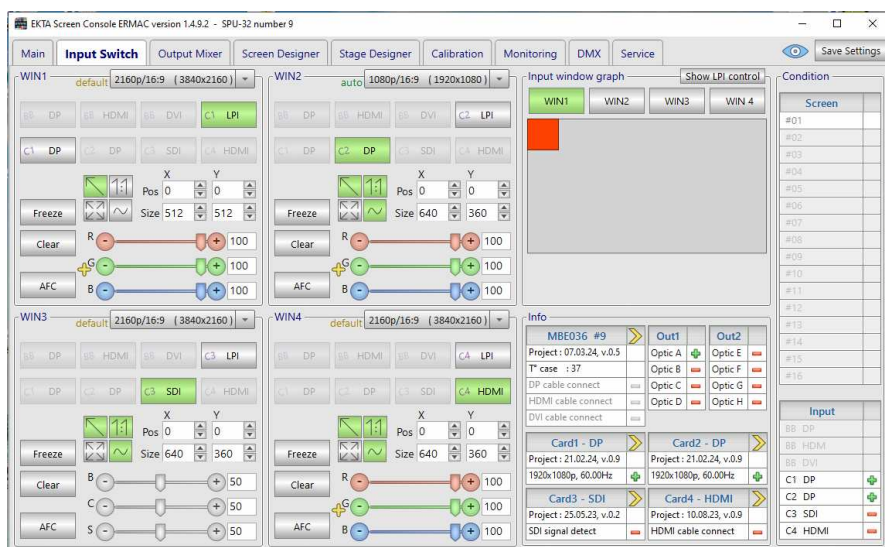


Рисунок 11.5 – Зовнішній вигляд програми «EKTA Screen Console».

При наведенні курсору миші на поле параметрів вставної картки, в випадаючому вікні, відображається повний список характеристик відеосигналу (рис.6) - роздільна здатність (resolution), колірний простір

(color space), глибина кольору (color depth), кількість кадрів за секунду в Гц (frame freq), повна кількість рядків зображення (total lines), частота пікселів(pixel clk).

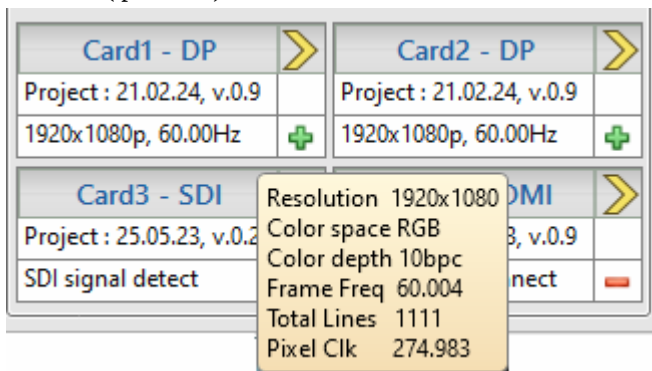


Рисунок 11.6 – Вікно розширених параметрів відеосигналу.

У програмі «EKTA Screen Console», передбачена можливість запису файлу EDID. Для цього необхідно перейти на вкладку «Service» і в полі «EDID settings» завантажити попередньо створений файл для обраної вставної картки (рисунок 11.7).

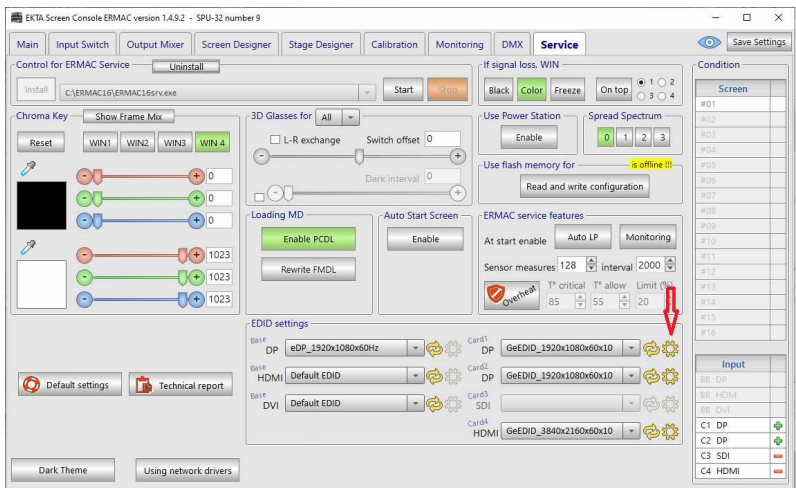


Рисунок 11.7 –Вікно «Service»

Також, в програмному забезпеченні вбудований генератор файлу EDID (рисунок 11.8), який дає можливість, задати наступні параметри відеоконтролера – максимальну роздільну здатність, кількість кадрів за секунду, глибину кольору, таймінги.

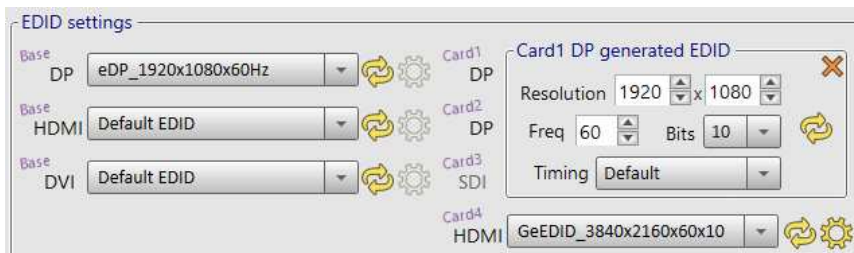


Рисунок 11.8 – Поля «EDID settings»

Для виклику підпрограми генератора файлу EDID, необхідно натиснути на значок налаштувань “шестерня”  в полі “EDID settings”. Після зміни параметрів, натиснути значок “перезавантаження” . Після чого, картка інтерфейсу DisplayPort (рисунок 11.10) згенерує сигнал, за яким джерело відеосигналу, прочитає оновлені дані файлу EDID.

В даній лабораторній роботі, джерелом відеосигналу є відеокарта NVIDIA. В програмі “панель керування NVIDIA” в розділі “Дисплей” відображається інформація про підключений відео контролер E16 eDP1(рисунок 11.9) та є можливість змінити налаштування зі списку (всі можливі параметри задані в файлі EDID) .

Рисунок 11.9 – Панель керування NVIDIA

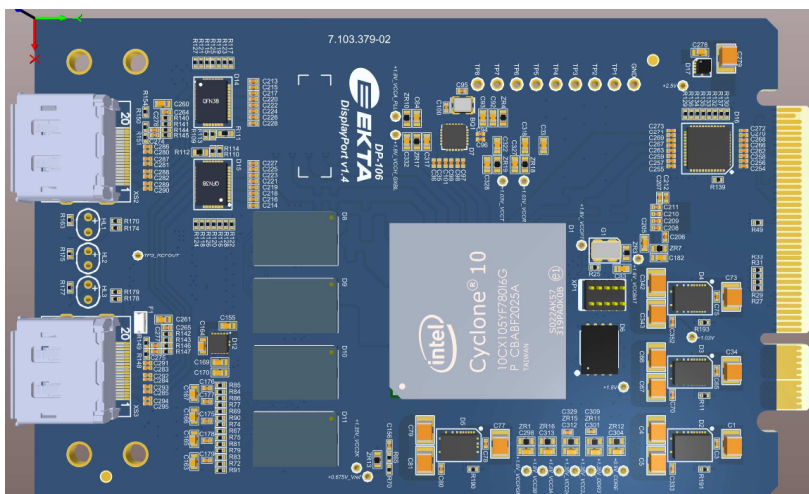


Рисунок 11.10 – Вставна картка відео контролера ERMAC16 з підтримкою інтерфейсу DisplayPort

Приклад 1. Відео тривалістю 30 хвилин, розмір кадру 720x576 пікселів з глибиною кольору 24 біт і кадровою частотою 30 fps - кадрів за секунду . Потрібно розрахувати бітову швидкість відеопотоку, та розмір відео.

Розв'язок:

$760 * 576 = 437760$ - пікселів на кадр

$437760 * 24 = 10\,506\,240$ = 10,50624 Мбіт – кількість біт на кадр

$10,50624 * 30 = 315,1872$ Мбіт/с – бітова швидкість

$315,1872 \text{ Мбіт/с} * 1800\text{с} = 567336,96 \text{ Мбіт} = 70,91 \text{ Гібайт}$

Приклад 2. Зображення передається по чотирьом лініям з бітовою швидкість 10,0 Гбіт/с на лінію і схемою кодування 8b/10b. Розрахувати максимальний розмір кадру в пікселях для відеопотоку з кадровою частотою $F_s = 120$ Гц і глибиною кольору 30 bpp.

Розв'язок:

Схема кодування 8b/10b передбачає пакети довжиною 10 біт, з яких 8 біт відведено під дані, тому для передачі даних доступно лише 80% ($8/10 = 0,8$) смуги пропускання.

$10.0 * 4 = 40.0$ Гбіт/с = 40 000 Мбіт/с – загальна бітова швидкість

$40\,000 * 0,8 = 32000$ Мбіт/с – бітова швидкість з врахуванням втрат через кодування

$32000 / 120 = 266,666$ Мбіт – кількість біт на кадр

$266,666 / 30 = 8\,888\,866$ пікселів

Хід заняття

1. Опрацювати лекційний матеріал за темою заняття.
2. Ознайомитися з описом Enhanced EDID (E-EDID) Release A v2.0.
3. Ознайомитися з програмним забезпеченням “ЕКТА Screen Console” для управління відеоконтролером ERMAC16.
4. Виконати задачі 1, 2 відповідно до варіанту.
5. Виконати завдання 3. Записати результати вимірювань.
6. Результати вимірювань оформити та представити до захисту у звіті.

Задача 1.

Розрахувати бітову швидкість відеопотоку, та розмір відео використовуючи дані таблиці 11.1, де вказано характеристики відеопотоку.

Таблиця 11.1

Номер варіанту	F_f , Гц	Розмір кадру, пікселів	Глибина кольору, bpp	Тривалість відео, хвилини
1	30	2560x1600	48	30
2	48	3840x2400	24	25
3	50	2560x1600	18	60
4	60	1920x1200	30	45

5	100	2560x1440	18	30
6	120	1920x1080	24	20
7	144	1024x768	30	25
8	240	720x576	18	15
9	24	3840x2160	48	40
10	25	2560x1600	36	60
11	30	2048x1536	30	90
12	48	4096x2160	24	35
13	50	1920x1200	18	50
14	60	1280x1024	30	25
15	100	1024x768	36	15
16	120	800x600	48	20

Задача 2.

Розрахувати максимальний розмір кадру в пікселях який можна передати через канал зв'язку. Характеристики каналу зв'язку та відеопотоку наведені у таблиці 11.2.

Таблиця 11.2

Номер варіанту	Бітова швидкість на лінію, Гбіт/с	Число ліній	$F_{\text{Гц}}$	Глибина кольору, bpp	Схема кодування
1	1,62	4	30	48	8b/10b
2	2,7	4	48	24	8b/10b

3	5,4	2	50	18	128b/132b
4	8,1	2	60	30	128b/132b
5	10,0	4	100	18	8b/10b
6	13,5	4	120	24	8b/10b
7	20,0	2	144	30	128b/132b
8	20,0	4	240	18	128b/132b
9	1,62	2	24	48	8b/10b
10	2,7	2	25	36	8b/10b
11	5,4	2	30	30	8b/10b
12	8,1	2	48	24	8b/10b
13	10,0	4	50	18	128b/132b
14	10,0	4	60	30	128b/132b
15	13,5	4	100	36	128b/132b
16	20,0	4	120	48	128b/132b

Задача 3.

На вхідній картці з підтримкою інтерфейсу DisplayPort (рис. 10) на тестові точки виведені контрольні сигнали:

TP1 – сигнал вертикальної синхронізації - VS (vertical synchronization)

TP2 – сигнал горизонтальної синхронізації -HS (horizontal synchronization)

TP3 – сигнал наявності даних (піксель зображення) – DQ(data qualifier)

Відео сигнал має наступні параметри:

Частота кадрів - $F_f = 60$ Гц

Роздільна здатність - 1920x1080 пікселів

Розгортка зображення – прогресивна

Глибина кольору – 8bpc (24bpp)

Щупом першого каналу осцилографа стати на тестову точку TP1 - сигнал вертикальної синхронізації. Встановити синхронізацію від першого каналу. Щупом другого каналу стати на точку TP2 – горизонтальна синхронізація. Щуп третього каналу розмістити на точці TP3 - сигнал наявності даних. Записати значення частот вертикальної і горизонтальної синхронізації. Оцінити тривалість передачі одного рядка зображення.

В програмі “EKTA Screen Console” на вкладці “Service” і в полі “EDID settings” запустити підпрограму редактора файлу EDID. Змінити максимальну роздільну здатність та кількість кадрів за секунду на значення наведені в Таблиця 1, відповідно до варіанту. Натиснути значок перезавантаження EDID. Далі в налаштуваннях відеокarti, вибрати нове максимальне розширення та частоту кадрів. Після застосування змін, виміряти та записати значення частот вертикальної і горизонтальної синхронізації. Оцінити тривалість передачі одного рядка зображення.

Контрольні питання

1. Як впливає глибина кольору зображення на роздільну здатність кадру при незміній частоті кадрів та ширині каналу?
2. Який з параметрів відеозображення впливає на «плавність» відео?
3. Для чого необхідний стандарт даних **EDID**?

Інформаційні ресурси в інтернеті:

1. <https://glenwing.github.io/docs/VESA-EEDID-A2.pdf>- VESA ENHANCED EXTENDED DISPLAY IDENTIFICATION DATA
2. <https://uk.wikipedia.org/wiki/DisplayPort>

Додаток1.

SYSTEMERMAC16

Embedded PC	+
Embedded OS	+
ImageChannel (simultaneous)	2-6

CONTROL

ControllInterface (simultaneous)	System, USB 2.0, Ethernet, DMX
ConfigurableandControlSoftware	ESConsole, ScreenBuilder (Embedded PC Location)

INPUT

DisplayPort	3840 x 2160 x 120 Hz
-------------	----------------------

HDMI	3840 x 2160 x 120 Hz
CVBS	+
3G SDI (Copper)	1x 4x (6G)
3G SDI (Fiber)	2x 4x (6G)
NDI	+
AudioInput	3G SDI, DP, HDMI
Genlock	+

OUTPUT

Fiberopticdisplaylink	8 x 2.3 Mpx
GigabitEthernet	+
HDMI Loop	+
SDI Loop	(6G)
AudioOutput	AES / EBU HDMI Outputaudiopackettransferove r FO

FEATURES

OutputResolution, 60 Hz	16,6 Mpx
HDR /10bit / 3D	+
Signaldelay, frames	1
VideoProcessing	Scaling, PIP, Cropping, Switching

Знайомство з виробничими процесами пайки друкованих плат

Мета заняття:

- 1 Практичне знайомство з реальним процесом пайки на виробництві ТОВ “Екта-Пром”
- 2 Вивчення і освоєння методів оцінки працездатності та способів пошуку несправностей.
- 3 Розрахунок витрат (на прикладі витрат паяльної пасти для 1000 одиниць).

Основні питання заняття:

- 1 Методика розрахунку.
- 2 Огляд онлайн перетворювачів та їх налаштувань.
- 3 Відпрацювання алгоритму перетворення, знаходження помилки перетворення.

Короткі теоретичні відомості

Виготовлення електронного пристрою це складний процес, що часто включає такі складові:

- Підбір та закупівля комплектуючих
- Розробка дизайну виробу
- Підготовка виробництва до монтажу
- Поставка плат та трафаретів для SMD-монтажу
- SMD-монтаж
- DIP-монтаж (при великих обсягах виробництва для пайки вивідних компонентів використовується установка пайки хвилию припою)
- Збірка виробу за викрутковою технологією
- Виготовлення жгутів та кабелів
- Кліматичне тестування виробів
- Виготовлення деталей механічного виробництва
- Налагодження блоків
- Селективне нанесення вологозахисних покриттів

- Доробка корпусів, збірка, упакування

Пайка компонентів на друкованих платах може виконуватися як вручну так і на автоматичних лініях.

Поверхневий (SMT) монтаж друкованих плат на виробництві ТОВ “Екта-Пром” здійснюється на трьох автоматичних лініях, до складу яких входять:

1. Автомат трафаретного друку DEK HORIZON 03I, що дозволяє наносити паяльну пасту або клей на заготовки розміром

50 – 400 x 40,5 – 400 мм. Система технічного зору автомата дозволяє поєднувати зображення трафарету з малюнком друкованої плати з точністю 25 мікрон, а також проводити перевірку якості нанесення паяльної пасти в автоматичному режимі.

Автомат установки поверхнево монтуюємих (SMD) компонентів Samsung 421, дозволяє встановлювати широку номенклатуру компонентів з максимальною продуктивністю до 21 000 компонентів на годину. Система дозволяє встановлювати чіп компоненти від 0201 (0,6x0,3мм) до мікросхем розміром 45 x 45 мм, а також корпуси BGA з кроком виводу від 0,5 мм й діаметром шарикових виводів від 0,3 мм, та планарні з кроком від 0,45 мм.

Дев'ятизонна конвекційна піч MR933, призначена для пайки складних електронних вузлів з використанням безсвинцевих паяльних паст та паст, що містять свинець. Установка оптичної інспекції SJ5000 дозволяє проводити автоматичний контроль якості паяних з'єднань, наявності перемичок припою, правильності установки полярних компонентів тощо.

1. Автомат трафаретного друку DEK 265GS, що дозволяє наносити паяльну пасту або клей на заготовки розміром 50 – 400 x 38 – 400 мм.

Автомат установки поверхово монтуюємих (SMD) компонентів Samsung 411, дозволяє встановлювати широку номенклатуру компонентів з максимальною продуктивністю до 42 000 компонентів на годину.

П'ятизонна конвекційна піч M520 призначена для пайки складних електронних вузлів з використанням безсвинцевих паяльних паст та

паст, що містять свинець. Дев'ятизонна конвекційна піч MR933 +, призначена для пайки складних електронних вузлів з використанням безсвинцевих паяльних паст та паст, що містять свинець.

Установка оптичної інспекції SJ5000 дозволяє проводити автоматичний контроль якості паяних з'єднань, наявності перемичок припою, правильності установки полярних компонентів тощо.

1. Автомат трафаретного друку DEK 265 (рис. 3.), що дозволяє наносити паяльну пасту або клей на заготовки розміром 50 – 400 x 38 – 400 мм. Автомат установки поверхово монтуємих (SMD) компонентів Toraz XII (рис. 3.) дозволяє встановлювати широку номенклатуру компонентів з максимальною продуктивністю до 20 000 компонентів на годину.

Монтаж вивідних (DIP) компонентів, при великому обсязі замовлення, може здійснюватися на високошвидкісний автоматизований установці Radial 8xt з продуктивністю до 25 000 компонентів на годину. Пайка вузлів проводиться на установці пайки хвилею припою виробництва

компанії SOLTEC (рис. 4.).

При необхідності відмивання вузлів використовуються ультразвукові ванни.

Пристрої після запаювання проходять контроль:

- 1) оптичний (машина та людина)
- 2) функціональну перевірку працездатності
- 3) напрацювання

Якщо в пристрої виявлено дефект — пристрій повертається для усунення на відповідну ділянку і на повторний контроль.

Найпоширенішими причинами несправності вузлів, що виявляються в процесі контролю є:

- відсутність якісного паяного з'єднання
- наявність паяного з'єднання близько розміщених контактів не передбаченого схемою
- неналежне позиціонування компонента
- відсутність компонента, що вказаний у специфікації
- встановлення компонента, що не відповідає специфікації

- встановлення неякісного компонента, або такого що мав неналежне зберігання

- механічне пошкодження компоненту або друкованих провідників

- брак друкованої плати

Дефекти пайки

В результаті пайки можуть виникати дефекти, які проявляють себе відразу або з часом використання пристрою. В стандарті IPC-J-STD-001 вказано норми паяних з'єднань.

Загалом комітети IPC розробили такі стандарти для проектування, виготовлення, складання та тестування друкованих плат (рисунок 12.1):

IPC STANDARDS — EVERYTHING YOU NEED FROM START TO FINISH

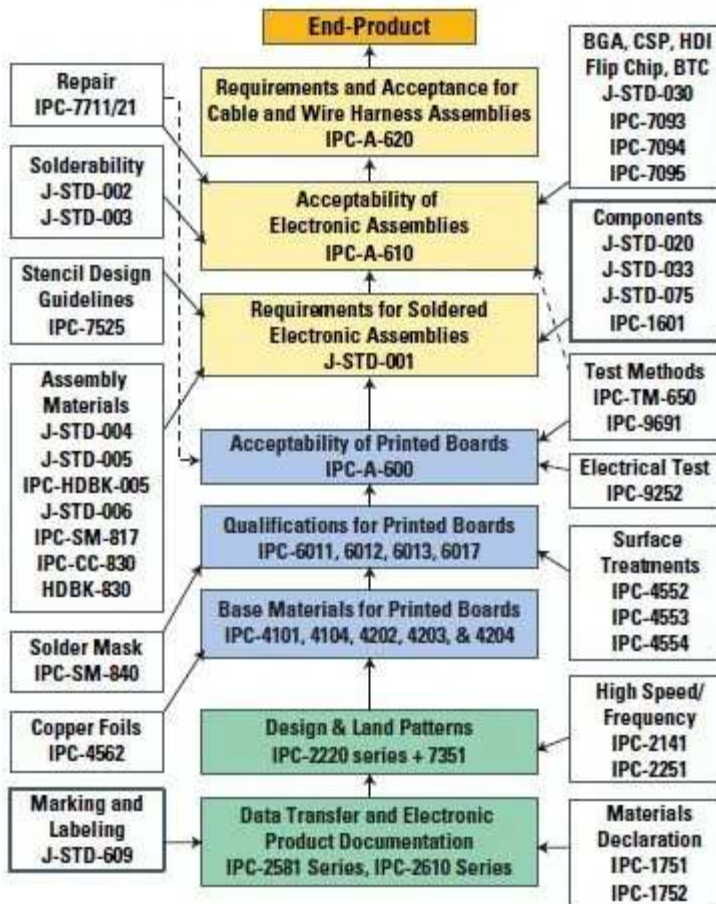


Рисунок 12.1 – Стандарти проектування, виготовлення, складання та тестування друкованих плат

Нижче наведено фото деяких проблем пайки:

З'єднання припосм (рисунок 12.2) — як правило пов'язані з надмірною кількістю паяльної пасти або неправильним термопрофілем.

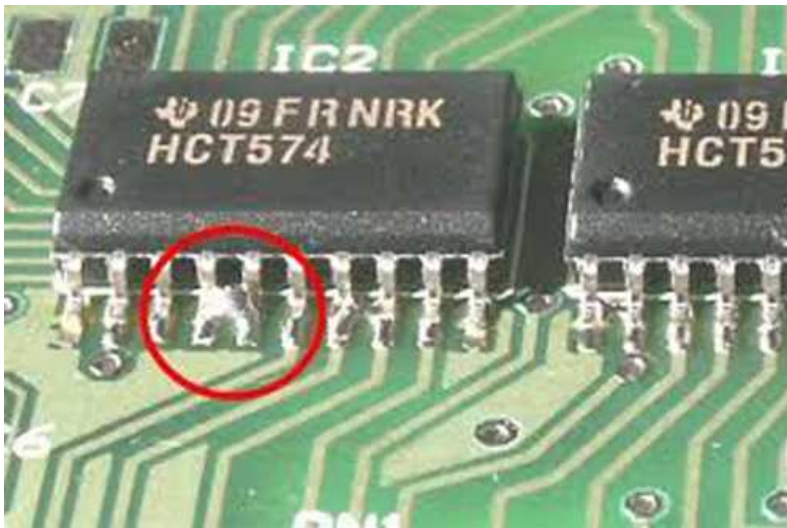


Рисунок 12.2 – З'єднання припосм

Щоб запобігти цій несправності, слід розплавити ці ділянки за допомогою паяльника до такої міри, щоб він повністю перетворився на рідину, після чого ви зможете легко розірвати шлейф.

Надмірна кількість припою – рисунок 12.3.



Рисунок 12.3 – Надмірна кількість припою

Не дозволяє оцінити надійність з'єднання, та може бути причиною дефектів, що проявляються з часом чи нагрівом (як от окислення виводу).

Згортання припою в кульки – рисунок 12.4.



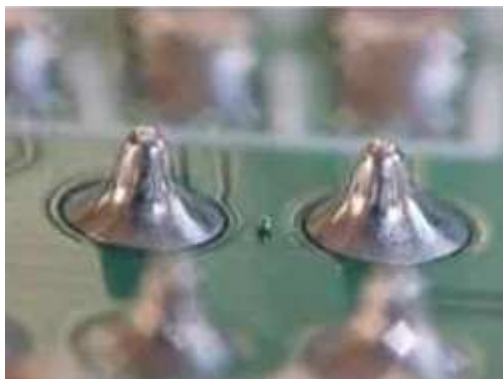


Рисунок 12.4 – Згортання припою в кульки

Ефект “надгробного каменю” – рисунок 12.5

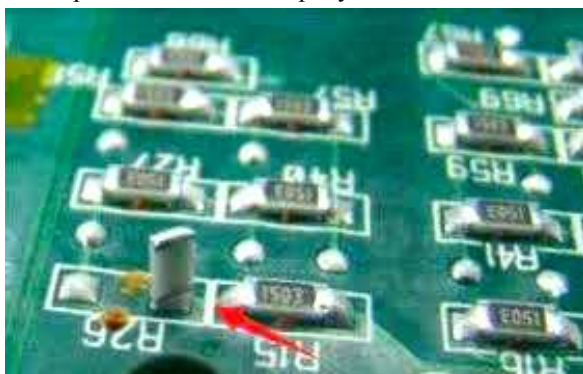


Рисунок 12.5 – Ефект “надгробного каменю”

Основними причинами є відсутність теплового рельєфу конструкції та неоднакова товщина доріжки, що з'єднує колодки.



Рисунок 12.6 – Недостатнє змочування

Виводи компонента не контактують належним чином із контактом на друкованій платі – рисунок 12.7.

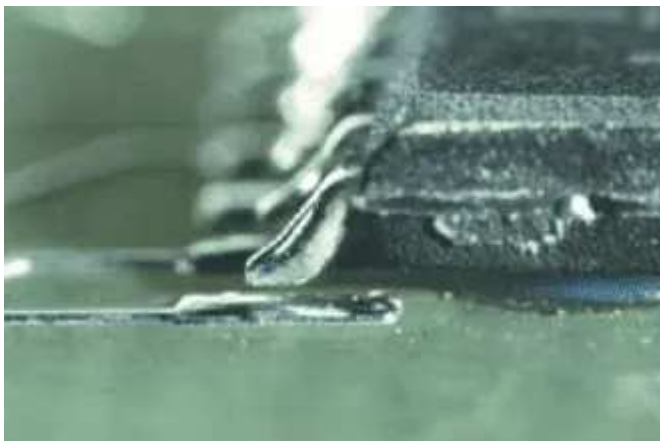


Рисунок 12.7 – Відсутність контакту

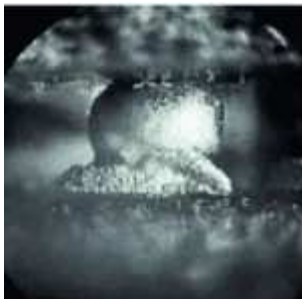


Рисунок 12.8 – Кулька припою та паяльна паста не зливаються разом

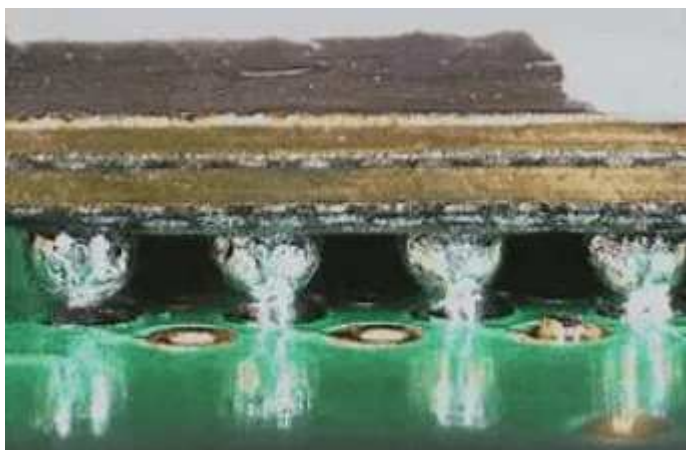


Рисунок 12.9 – Оплавлення паяльної пасти на кульках припою BGA не завершено

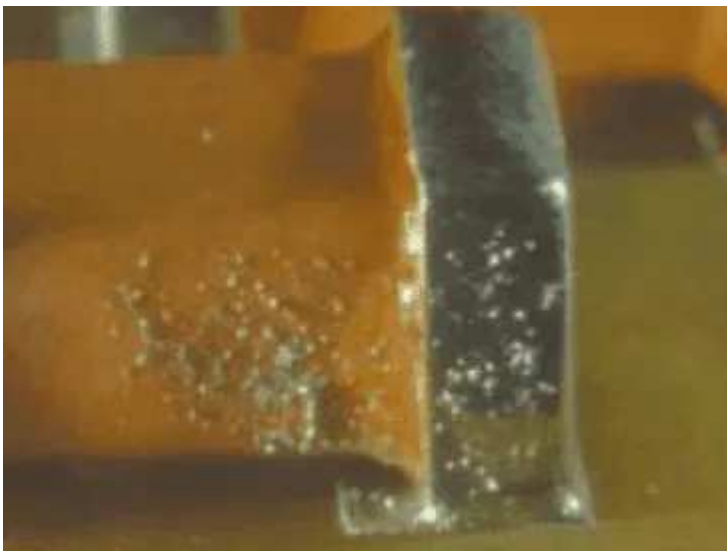


Рисунок 12.10 – Бризки кульок припою на компоненті

Розрахунок витрат

Одним із аспектів виробництва є розрахунок витрат матеріалів (для пайки, очистки, вологозахисту...). Так розрахунок витрат пасти на пристрій і підбір часто виконується безпосередньо зважуванням на отриманих зразках друкованих плат. Якщо ж потрібно заздалегідь прорахувати витрати то нижче зразок розрахунку витрат припою для запаювання одного виводу. Прийнято, що для міцного з'єднання необхідна відстань між виводом компонента та друкованою платою 75мкм.

Приклад 1

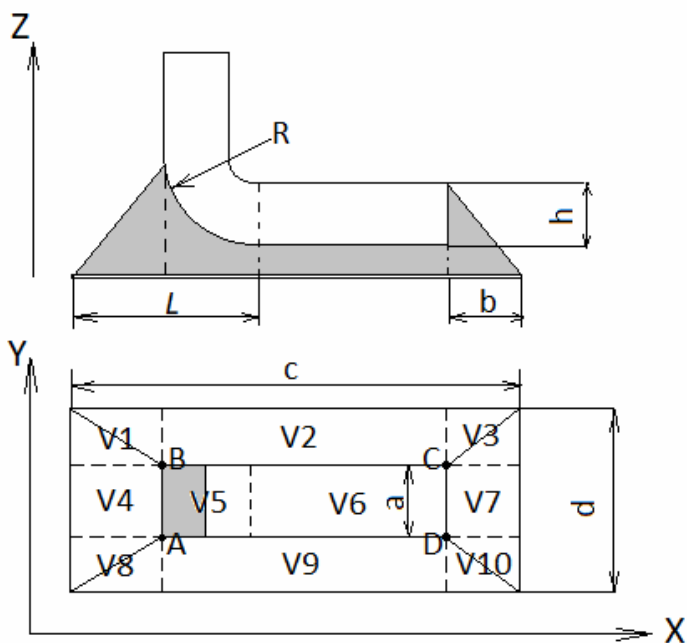


Рисунок 12.11 – Бризки кульок припою на компоненті

h- товщина виводу,

d та c – розміри контактної площадки,

сірим позначено припій

Розв'язок.

Нижче таблиця для розрахунку зон припою

Зона об'єму припою	Формула для обчислення об'єму	Значення X	Значення Y
V1=V8	$xyz/3$	$l - \sqrt{2Rh - h^2}$	$(d-a)/2$

V2=V9	xyz/2	$c - l - \sqrt{2Rh - h^2}$	(d-a)/2
V3=V10	xyz/3	b	(d-a)/2
V4	xyz/2	$l - \sqrt{2Rh - h^2}$	a
V5	$y \left[xz + 0.5x(R - z) - \frac{\pi R^2 \sin^{-1} \frac{x}{R}}{360} \right]$	$\sqrt{2Rh - h^2}$	a
V6	xyz	$c - l - \sqrt{2Rh - h^2}$	a
V7	xyz/2	b	a

Наприклад, для паяльної пасти з вмістом флюсу 11% необхідний об'єм пасти буде розрахований наступним чином:

$$V = \frac{1}{1 - 0.11} * \sum_{i=1}^{10} V_i$$

Хід заняття

1. Опрацювати лекційний матеріал за темою заняття.
2. Прослухати розповідь про лінію виробництва
3. Вирішити завдання, які наведено нижче.
4. Оформити звіт про хід виробництва електронних пристроїв.

На реальних зразках продукції виявити невідповідності.
Зафіксувати дефект фотографією та запропонувати спосіб вирішення.

Варіант 1”Дефект - не програмується”

Варіант2” Дефект - нема відображення температури”

Варіант 3“ Дефект - нема індикації”

Задача 2.

Розрахувати орієнтовні витрати паяльної пасти M8
(Aimsolder)

(Sn 3.0Ag 0.5Cu) для пайки 1000 плат, що містять наступний склад
(базуючись на даних даташит):

Варіант 1

6шт - транзисторів DMN10H220LK3;

230 шт – резисторів RC0603-0.1Вт

50шт - 0603-X7R-16B-0.47 мкФ ±10% Конденсатор

(Walsin)

1 шт - ATmega168-20AI Мікросхема (Atmel)

Варіант 2

100 шт - мікросхем TLC5944PWP

800 шт – резисторів RC0603-0.1Вт

100 шт -0805-X5R-25B-4.7 мкФ ±10% Конденсатор

(Samsung)

Варіант3

10 шт - 74VHC245PW

20 шт – резисторів RC0603-0.1Вт

2шт - SL2-010-S116/01-11 вилка (E-tec Interconnect)

Завдання 3.Оформити звіт.

Література

Основна література

1. Ralf Rudersdorfer Radio receiver technology : principles, architectures, and applications / Ralf Rudersdorfer, Ulrich Graf, Hans Zahnd. – John Wiley & Sons Ltd, 2014. – 317 p.
2. Cornell Drentea Modern Communications Receiver Design and Technology. – Artech House, 2010. – 485 p.
3. Kevin McClaning Wireless Receiver Design for Digital Communications. – 2nd ed. – SciTech Publishing, Inc, 2012. – 775 p.
4. Kevin McClaning, Tom Vito Radio receiver design. – Noble Publishing Corporation, 2000. – 790 p.
5. PaulClark, David Clark Introduction to Software Defined Radio. Volume 1. – Meadow Registry Press, 2015. – 173 p.
6. John Bard Software defined radio : the software communications architecture / John Bard, Vincent J. Kovarik Jr. – John Wiley & Sons Ltd, 2007. – 465 p.

Допоміжна література

1. Peter B. Kenington RF and Baseband Techniques for Software Defined Radio. – Artech House, 2005. – 352 p.
2. Marcel Dekker Radio receiver design – Marcel Dekker, Inc, 1989. – 489 p.
3. Heinrich Meyr Digital communication receivers: synchronization, channel estimation and signal processing / Heinrich Meyr, Marc Moeneclaey, Stefan A. Fechtel.– John Wiley & Sons, 1998. – 843 p.
4. Теорія і практика управління використанням радіочастотного ресурсу / П. В. Слободянюк, Т. М. Наритник, В. Г. Благодарний та ін.; за ред. В. Г. Кривуци. – К.: ДУІКТ, 2012. – 596 с.