

**МІНІСТЕРСТВО ВНУТРІШНІХ СПРАВ УКРАЇНИ  
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
ВНУТРІШНІХ СПРАВ**

**Кафедра інформаційних технологій та кібербезпеки факультету № 4**

**МЕТОДИЧНІ МАТЕРІАЛИ  
ДО ПРАКТИЧНИХ ЗАНЯТЬ**

навчальної дисципліни  
«Електроніка та схемотехніка»  
обов'язкових компонент освітньої програми  
першого (бакалаврського) рівня вищої освіти

**125 «Кібербезпека» (поліцейські)**

**Харків 2020**

**ЗАТВЕРДЖЕНО**

Науково-методичною радою  
Харківського національного  
університету внутрішніх справ  
Протокол від 23.09.2020 № 9

**СХВАЛЕНО**

Вченою радою факультету № 4  
Протокол від 16.09.2020 № 5

**ПОГОДЖЕНО**

Секцією Науково-методичної ради  
ХНУВС з технічних дисциплін  
Протокол від 18.09.2020 № 5

Розглянуто на засіданні кафедри інформаційних технологій та кібербезпеки  
(протокол від 15.09.2020 № 16)

**Розробники:**

1. Доцент кафедри інформаційних технологій та кібербезпеки факультету № 4,  
кандидат технічних наук, доцент Клімушин П.С.

**Рецензенти:**

1. Завідувач кафедри інформаційних управляючих систем ХНУРЕ, д.т.н., професор  
Петров К. Е.

2. Провідний науковий співробітник Науково-дослідної лабораторії з проблем  
розвитку інформаційних технологій ХНУВС, к.т.н., доцент Мордвинцев М.В.

# Зміст

<b>ПРАКТИЧНЕ ЗАНЯТТЯ №1. МАТЕРІАЛИ ТА КОМПОНЕНТИ ЕЛЕКТРОННОЇ ТЕХНІКИ.....</b>	<b>4</b>
1. Класифікація матеріалів та компонентів електронної техніки .....	5
2. Пасивні елементи: резистор, індуктивність та ємність як елементи електричного кола .....	5
3. Активні елементи: діод, світлодіод, фотодіод, оптрон, транзистор .....	8
4. Аналогова та цифрова електроніка .....	10
5. Контрольні питання .....	13
<b>ПРАКТИЧНЕ ЗАНЯТТЯ №2. РОЗРАХУНОК ЛОГІЧНИХ СХЕМ .....</b>	<b>14</b>
1. Розв'язання завдань щодо мінімізації логічних функцій.....	14
2. Індивідуальні завдання на тему «Логічні основи цифрової схемотехніки» .....	18
3. Контрольні питання .....	25
<b>ПРАКТИЧНЕ ЗАНЯТТЯ №3. ДОСЛІДЖЕННЯ БІПОЛЯРНИХ І ПОЛЬОВИХ ТРАНЗИСТОРІВ .....</b>	<b>25</b>
1. Загальні відомості про біполярні та польові транзистори .....	26
2. Визначення залежності вихідної напруги від вхідної для схеми із загальним емітером.....	27
3. Вимірювання коефіцієнта підсилення схеми із загальним емітером і негативним зворотним зв'язком по струму .....	27
4. Визначення залежності вихідної напруги від вхідної для схем із загальною базою і загальним колектором .....	27
5. Транзистор як джерело стабільного струму.....	28
6. Визначення залежності вихідної напруги від вхідної для схеми із загальним истоком та истокового повторювача .....	28
7. Керований дільник напруги.....	28
8. Контрольні питання .....	29
<b>ПРАКТИЧНЕ ЗАНЯТТЯ №4. ДОСЛІДЖЕННЯ ПОСТІЙНОЇ ТА ОПЕРАТИВНОЇ ПАМ'ЯТІ КОМП'ЮТЕРА .....</b>	<b>29</b>
1. Дослідження функціонування модуля оперативної пам'яті.....	30
2. Дослідження функціонування модуля постійної пам'яті.....	32
3. Контрольні питання .....	32
<b>ПРАКТИЧНЕ ЗАНЯТТЯ №5. ПРОГРАМУВАННЯ МІКРОКОНТРОЛЕРІВ .....</b>	<b>33</b>
1. Вивчення алгоритмів додавання та віднімання багатобайтових чисел.....	33
2. Програмування операцій додавання та віднімання операндів.....	37
3. Порядок роботи в інтегрованому середовищі EdSIM51 SIMULATOR.....	39
4. Контрольні питання .....	42
<b>ПРАКТИЧНЕ ЗАНЯТТЯ №6. ОРГАНІЗАЦІЯ СИСТЕМИ ПЕРЕРИВАНЬ ТА ТАЙМЕРІВ-ЛІЧИЛЬНИКІВ У МІКРОКОНТРОЛЕРАХ MCS-51 .....</b>	<b>43</b>
1. Організація системи переривань в MCS-51 .....	43
2. Організація таймерів/лічильників в MCS-51 .....	45
3. Формування часової затримки таймером .....	47
4. Контрольні питання .....	50
<b>ПРАКТИЧНЕ ЗАНЯТТЯ №7. ДОСЛІДЖЕННЯ ЦИФРО-АНАЛОГОВИХ ТА АНАЛОГО-ЦИФРОВИХ ПЕРЕТВОРЮВАЧІВ .....</b>	<b>51</b>
1. Основні характеристики ЦАП і АЦП .....	51
2. Дослідження цифро-аналогового перетворювача із двійково-зваженими опорамми .....	53
3. Дослідження ЦАП на основі матриці R-2R.....	54
4. Дослідження роботи мікросхем АЦП.....	55
5. Контрольні питання .....	56

## 1. Розподіл часу навчальної дисципліни за темами

Номер та назва навчальної теми	Кількість годин відведених на вивчення навчальної дисципліни						Вид контролю
	Всього	з них:					
		лекції	Семінарські заняття	Практичні заняття	Лабораторні заняття	Самостійна робота	
Семестр № 4							
Тема № 1. Матеріали та компоненти електронної техніки	12	4		2		6	
Тема № 2. Диференціюючі та інтегруючі кола імпульсної електроніки	12	2			4	6	
Тема № 3. Логічні основи цифрової схемотехніки	18	2		4	4	8	
Тема № 4. Схемотехніка логічних елементів	16	4		4		8	
Тема № 5. Схемотехніка цифрових елементів	14	4			2	8	
Тема № 6. Схемотехніка комбінаційних вузлів	16	4			4	8	
Всього за семестр № 4:	88	20		10	14	44	залік
Семестр № 5							
Тема № 6. Схемотехніка комбінаційних вузлів	16	4			4	8	
Тема № 7. Схемотехніка цифрових вузлів	16	4			4	8	
Тема № 8. Напівпровідникові запам'ятовуючі пристрої	16	4		4		8	
Тема № 9. Основи мікропроцесорної техніки	32	6		8	2	16	
Тема № 10. Цифро-аналогові і аналого-цифрові перетворювачі	12	2		4		6	
Всього за семестр № 5:	92	20		16	10	46	екзамен
Всього по дисципліні:	180	40		26	24	90	

## 2. Методичні вказівки до практичних занять

### Тема 1. Матеріали та компоненти електронної техніки

#### Практичне заняття №1. Матеріали та компоненти електронної техніки

##### Навчальна мета заняття:

1. Ознайомитися із принципом дії і основними характеристиками матеріалів та компоненти електронної техніки.

**Кількість годин:** 2 год.

##### Література:

1. Крилик Л.В., Селецька О.О. Матеріали електронної техніки: навчальний посібник. Вінниця: ВНТУ, 2017. 120 с

2. Чешко І.В. Вступ до спеціальності «Електроніка» : навчальний посібник. Суми : Сумський державний університет, 2017. 148 с.

3. Квітка С.О., Яковлев В.Ф., Нікітіна О.В. Електроніка та мікросхемотехніка: Навчальний посібник / За ред. проф. В.Ф. Яковлева. Київ: Аграрна освіта, 2010. 329с.

4. Болюх В. Ф., Данько В. Г. Основи електроніки і мікропроцесорної техніки: Навч. посібник. Харків: НТУ «ХПІ», 2011. 257 с.

**Матеріально-технічне забезпечення:** комп'ютерна мережа із системою модулювання електронних схем Electronics Workbench.

### **Навчальні питання**

1. КЛАСИФІКАЦІЯ МАТЕРІАЛІВ ТА КОМПОНЕНТІВ ЕЛЕКТРОННОЇ ТЕХНІКИ .....	5
2. ПАСИВНІ ЕЛЕМЕНТИ: РЕЗИСТОР, ІНДУКТИВНІСТЬ ТА ЄМНІСТЬ ЯК ЕЛЕМЕНТИ ЕЛЕКТРИЧНОГО КОЛА .....	5
3. АКТИВНІ ЕЛЕМЕНТИ: ДІОД, СВІТЛОДІОД, ФОТОДІОД, ОПТРОН, ТРАНЗИСТОР .....	8
4. АНАЛОГОВА ТА ЦИФРОВА ЕЛЕКТРОНІКА .....	10
5. КОНТРОЛЬНІ ПИТАННЯ .....	13

### **1. Класифікація матеріалів та компонентів електронної техніки**

*Електроніка* - науково-технічна галузь дослідження законів взаємодії електронів та інших носіїв електричного заряду з електромагнітними полями та створення електронних приладів, у яких ця взаємодія використовується для передавання, оброблення та зберігання інформації.

Компоненти електронної техніки можна поділити на два класи:

1) *Пасивні елементи*, які функціонують без додаткових джерел напруги живлення й не збільшують енергії вхідних електричних сигналів. До них відносяться резистори (R), конденсатори (C), котушки індуктивності (L), діоди (VD) та інші.

2) *Активні компоненти* - це компоненти, для функціонування яких потрібне зовнішнє джерело напруги живлення. Вони здатні підсилювати електричні сигнали. До них відносяться: електровакуумні прилади, газорозрядні прилади, напівпровідникові прилади, інтегральні мікросхеми.

Всі матеріали поділяються на провідники, напівпровідники та діелектрики.

*Провідники* - це матеріали, в яких при нормальних умовах є вільні електричні заряди, котрі забезпечують проходження електричного струму при докладанні напруги.

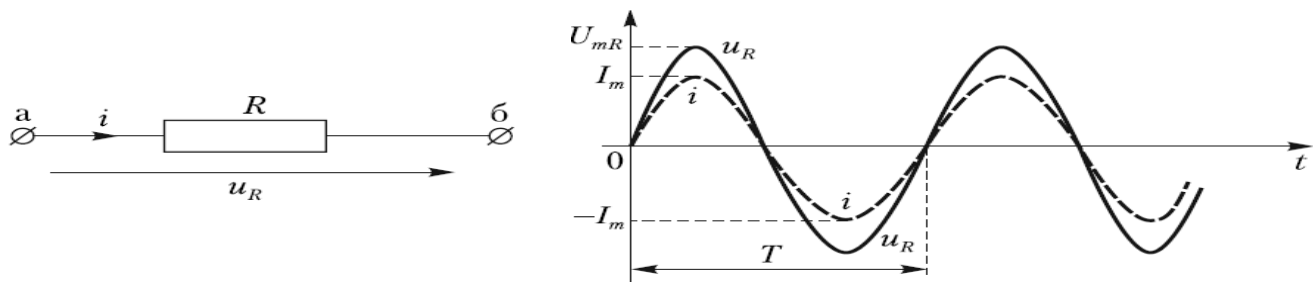
*Напівпровідникові матеріали* (германій *Ge*, кремній *Si*, селен *Se*, бар *Ba*, телур *Te*, арсенід галію *GaAs* та інші) використовуються для створення напівпровідникових пристроїв (діодів, транзисторів, тиристорів, резисторів та ін.).

*Діелектрики* - матеріали, в яких нема вільних електричних зарядів, і тому вони практично не проводять електричний струм.

### **2. Пасивні елементи: резистор, індуктивність та ємність як елементи електричного кола**

*Резистор* - елемент, в якому відбувається незворотний процес перетворення електричної енергії в теплову (іноді - в механічну або хімічну). Резистор - завжди споживач енергії. Коли струм в резисторі постійний, напруга також постійно.

У сталому синусоїдальному режимі напруга і струм в резисторі по фазі збігаються (одночасно проходять через нуль, одночасно досягають максимуму).



Амплітудні значення напруги і струму в резисторі пов'язані співвідношенням  $U_{mR} = RI_m$ , яке можна вважати законом Ома для амплітуд.

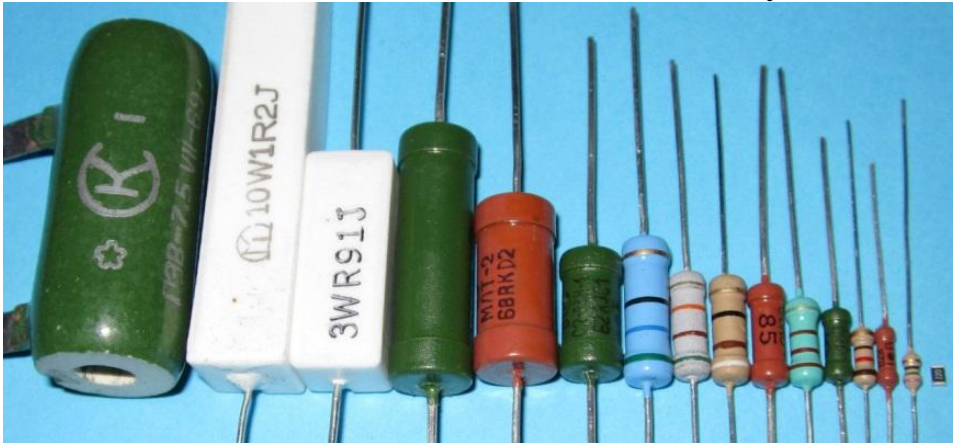
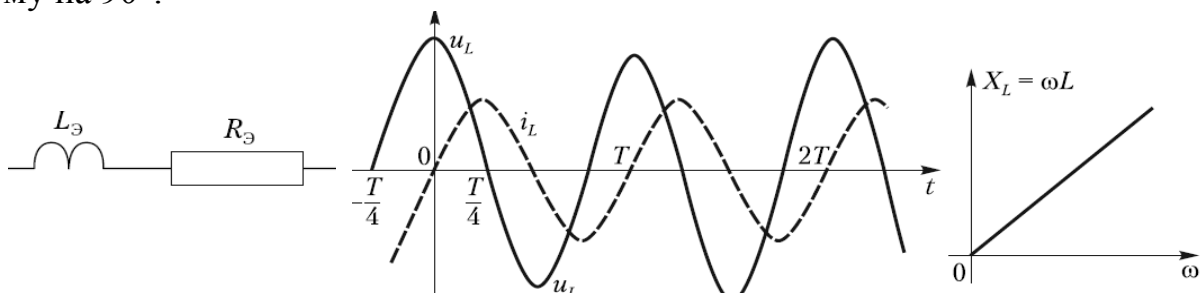


Рис. 1. Зовнішній вигляд резисторів різного номіналу

*Індуктивність* - елемент, запасуючий електричну енергію в магнітному полі. Запасена енергія при відповідних умовах може бути повністю повернута джерелу. Ідеальна індуктивність - тільки накопичувач енергії, властивостями незворотного споживання енергії не володіє.

$$u_L = \varphi_a - \varphi_b = L \frac{di(t)}{dt},$$

Тобто різницю потенціалів на індуктивному елементі пропорційна швидкості зміни струму в часі. Напруга на індуктивності – гармонійна функція тієї ж частоти, що і струм. Але фаза синусоїдальної напруги на індуктивності перевищує фазу струму на  $90^\circ$ .



Амплітуди напруги і струму в індуктивності пов'язані співвідношенням  $U_{mL} = \omega LI_m$ , його можна трактувати як закон Ома, при чому роль опору виконує величина  $X_L = \omega L$ , звана *індуктивним опором*.

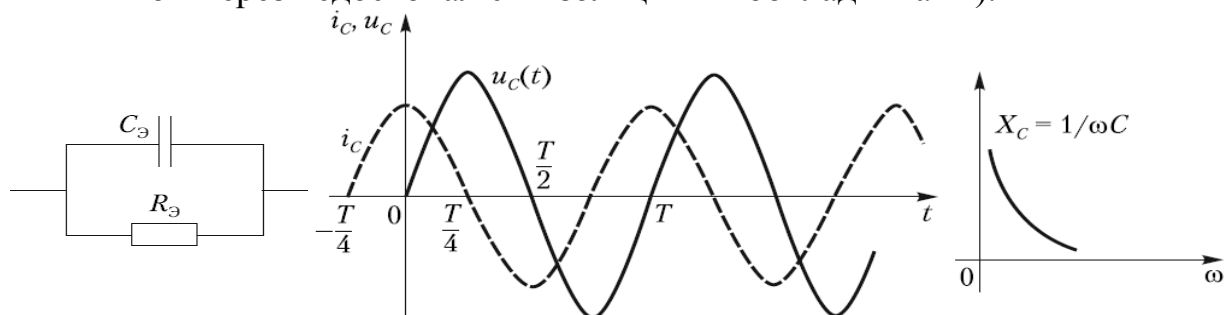


Рис. 2. Зовнішній вигляд котушок індуктивності різного типу

*Ємність* - елемент електричного кола, запасаючи енергію в електричному полі. Ідеальна ємність тільки накопичує енергію і при певних умовах може повернути її повністю.

В синусоїдальному режимі напруга і струм в ємності - гармоніки однієї частоти, що відрізняються по фазі. Фаза струму в ємності перевищує фазу напруги на  $90^\circ$ . Амплітуди напруги і струму в ємності пов'язані співвідношенням  $U_{mC} = I_m / (\omega C)$ , яке так само можна трактувати як закон Ома, роль опору виконує величина  $X_C = 1 / (\omega C)$ , звана *ємнісним опором*.

Реальний конденсатор проявляє не тільки ємнісні властивості, але і резистивні (головним чином через недосконалість ізоляції між обкладинками).



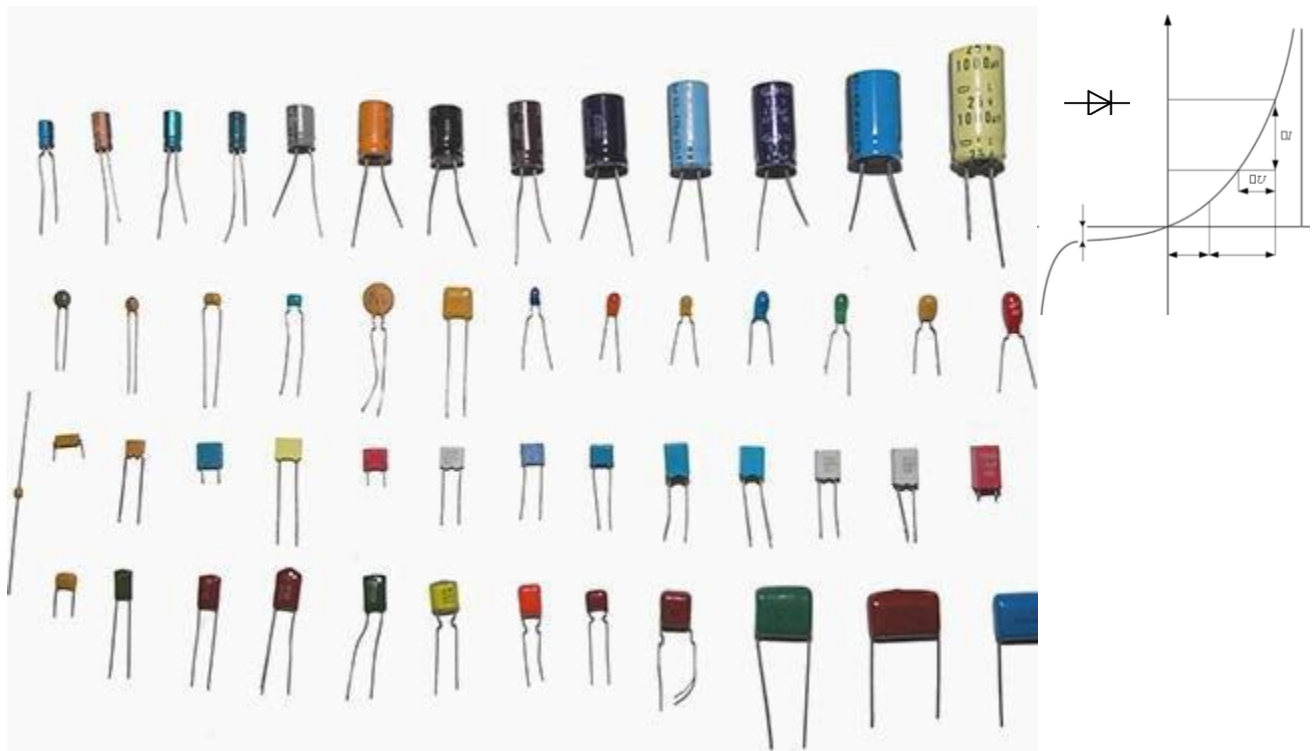


Рис. 3. Зовнішній вигляд конденсаторів різного типу

### 3. Активні елементи: діод, світлодіод, фотодіод, оптрон, транзистор

Активні електронні елементи на відміну від пасивних здатні керувати електричною енергією та підсилювати її. Приклади активних елементів : діод, світлодіод, фотодіод, напівпровідниковий лазер, діодний міст або випрямний діод, тріод або транзистор, мікросхема, мікропроцесор.

*Напівпровідниковим діодом* називається прилад який має один (або декілька) р-п переходів та два виводи. Всі напівпровідникові діоди можливо розділити на дві групи: випрямляючі та спеціальні. *Випрямляючі* призначені для випрямлення змінного струму. *Спеціальні* напівпровідникові діоди використовують різні властивості р-п переходів - явище пробою, бар'єрної ємності, наявність ланок з негативним опором та ін. Найбільш поширене використання отримали діоди які мають один р-п перехід. Електрод підключений до р області називається анодом, а до п - катодом.

*Діоди* мають односторонню провідність та використовуються для випрямлення змінного струму, стабілізації струму та напруги, формування імпульсів регулювання потужності та ін.

Для більшої ефективності в електроніці зазвичай діоди об'єднані в так звані діодні мости (рис. 4), що представляють собою послідовне підключення чотирьох діодів так щоб два діода із них були включені зустрічно. Діодні мости застосовують для живлення апаратури, в блоках живлення та зарядних

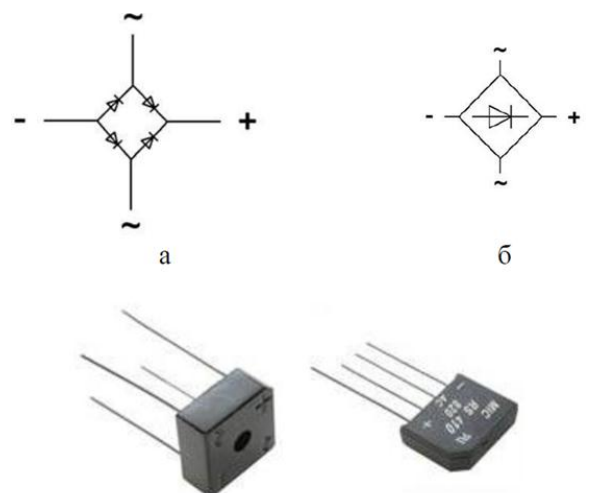


Рис. 4. Схема (а), умовне графічне позначення (б) та приклади зовнішнього вигляду (в) діодних мостів



пристроях.

Інший поширений вид діодів – *стабілітрон*. Він перешкоджає перевищенню напруги вище певного порогу на конкретній ділянці схеми. Цей активний елемент може виконувати як захисні так і обмежувальні функції. Стабілітрони працюють тільки в колах постійного струму. Однотипні стабілітрони можна з'єднувати послідовно для підвищення напруги, що стабілізується. Основним параметром стабілітроном є напруга стабілізації.

В сучасній електроніці поширюється використання оптоелектронних діодів: світлодіодів та фотодіодів.

Світлодіод випромінює світло при пропусканні через нього електричного струму (рис. 5). Вони застосовуються в пристроях індикації, телефонах для підсвічування дисплея і клавіатури, як джерело світла в ліхтарях і системах освітлення тощо. Світлодіод уявляє собою випромінюючий р-п перехід, свідчення в якому виникає внаслідок рекомбінації носіїв заряду (електронів і дірок). Воно спостерігається при змищенні переходу в прямому напрямі. Колір випромінювання світлодіодів визначається матеріалом, з якого вони виготовлені. Найчастіше зустрічаються світлодіоди червоного, зеленого, жовтого, помаранчевого свідчення. Існують світлодіоди змінного кольору свідчення з двома светоізлучаючими переходами, колір свідчення залежить від співвідношення струмів через переходи. Існують комбіновані прилади, в одному корпусі яких конструктивно поєднано два і більше світлодіодів різних кольорів.

*Цифро-буквений світлодіодний індикатор* зроблений у вигляді інтегральної мікросхеми з декількох світлодіодних структур, розташованих так, щоб при відповідних комбінаціях сегментів, що світяться, виходило зображення букви або цифри. Однорозрядні індикатори дозволяють відтворити одну цифру від 0 до 9 або деякі букви. Багаторозрядні індикатори відтворюють одночасно декілька знаків. У більшості індикаторів сегменти мають вигляд смужок (зазвичай 7 для кожного розряду).

*Фотодіоди* - напівпровідниковими діодами, в яких використовується залежність зворотного струму від світлового потоку. Такий режим роботи називається фотодіодним.

*Оптрон* - це напівпровідниковий прилад, в якому конструктивно об'єднано джерело і приймач випромінювання, що мають між собою оптичний зв'язок.

*Транзистор* – напівпровідниковий елемент з трьома виводами, на один з яких (колектор) подається сильний струм, а на інший (база) подається слабкий (керуючий) струм (рис. 6). При певній силі керуючого струму, струм з колектора починає текти на

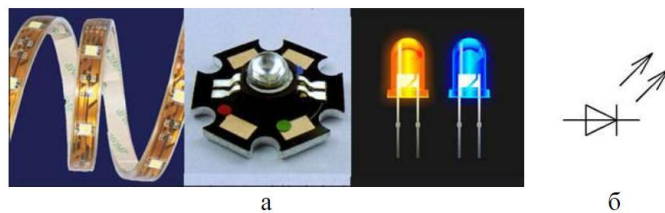


Рис. 5. Приклади зовнішнього вигляду (а) та умовне графічне позначення (б) світлодіодів

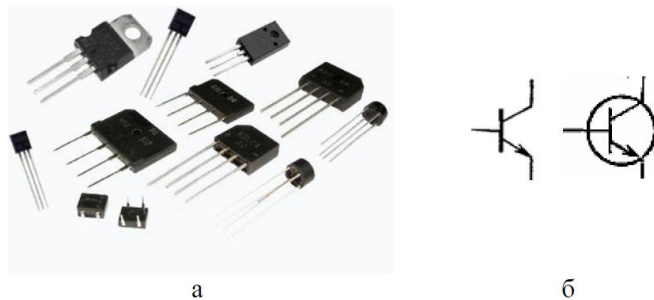


Рис. 6. Приклади зовнішнього вигляду (а) та умовне графічне позначення (б) транзисторів

третій вивід (емітер). Тобто транзистор – це своєрідний клапан, який при певній силі струму, різко зменшує опір і пропускає струм з колектора на емітер. Якщо до бази не прикласти електричний струм, то транзистор буде знаходитися в урівноваженому стані і не пропускати струм на емітер. У більшості схем транзистори використовуються як змінний резистор, опір якого змінюється під впливом слабкого вхідного сигналу. Однак основне призначення транзистора – підсилення електричного сигналу для швидкої зміни сили струму. Якщо сили струму в електричному ланцюзі недостатньо для включення транзистора, то використовуються декілька транзисторів з більшою чутливістю, що з'єднані каскадним способом. Потужні транзистори з'єднані в один або кілька корпусів використовуються в цифрових підсилювачах. Часто їм потрібне додаткове охолодження. У більшості схем, вони працюють в режимі ключа (в режимі перемикача).

На даний час найпоширенішим активним електронним елементом стали мікросхеми. *Мікросхема* – це мікроелектронний твердотільний пристрій, що представляє собою електронну інтегральну схему довільної складності виготовлену на напівпровідниковому кристалі (або плівці) і поміщена в нерозбірний корпус (рис. 7). В професійній літературі, мікросхеми часто називають *чипами* або *мікрочіпами*. Але все ж таки прийнято розділяти ці поняття. Під чипом розуміють власне кристал або плівку з електронною схемою, а під мікросхемою – чип вкладений в корпус. Компоненти, що входять до складу мікросхеми, не можуть бути виділені з неї як самостійний виріб, крім того, вони характеризуються деякими особливостями в порівнянні з дискретними наприклад транзисторами або діодами. Особливістю мікросхем є висока складність функцій, що ними виконуються, тому кількість компонентів в одній мікросхемі може обчислюватися сотнями тисяч і навіть мільйонами. За своїм функціональним призначенням мікросхеми поділяються на дві основні групи: аналогові, або лінійно-імпульсні, і логічні, або цифрові. За складністю електронних схем, які в них реалізовані, та кількістю елементів мікросхеми поділяються за ступенем інтеграції.

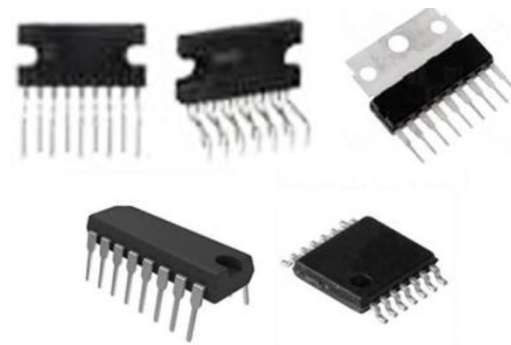


Рис. 7. Зовнішній вигляд мікросхем різних типів

#### 4. Аналогова та цифрова електроніка

*Сигнал* – це фізичний процес, який відображає повідомлення. У технічних системах найчастіше використовуються електричні сигнали. Вони, як правило, є функціями часу. Всі електронні сигнали можна розділити на *аналогові* (а), *дискретні* (б) і *цифрові* (в).

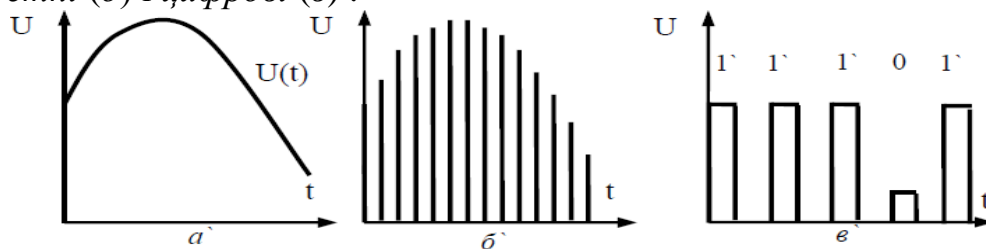


Рис. 8. Електронні сигнали: на аналогові (а), дискретні (б) і цифрові (в).

Можна сказати, що в природі практично всі сигнали – аналогові, тобто вони змінюються безперервно в певних межах. Саме тому перші електронні пристрої були аналоговими. Вони перетворювали фізичні величини в пропорційні їм напругу або струм, виробляли над ними якісь операції і потім виконували зворотні перетворення в фізичні величини. Наприклад, голос людини (коливання повітря) за допомогою мікрофона перетворюється в електричні коливання, потім ці електричні сигнали посилюються електронним підсилювачем і за допомогою акустичної системи знову перетворюються в коливання повітря – в сильніший звук.

У загальному розумінні аналогові електронні пристрої призначені для прийому, перетворення і передачі сигналів, які змінюються за законом безупинної (аналогової) функції. Аналогові пристрої відрізняються простотою, швидкістю, однак мають низьку завадостійкість і нестабільність параметрів при впливі зовнішніх дестабілізуючих факторів, наприклад температури, вологості, часу і т.д.

Однак аналогові сигнали і працюючи з ними аналогова електроніка мають великі недоліки, пов'язані саме з природою аналогових сигналів. Справа в тому, що аналогові сигнали чутливі до дії всіляких паразитних сигналів – шумів, наведень, перешкод.

*Шум* – це внутрішні хаотичні слабкі сигнали будь-якого електронного пристрою (мікрофона, транзистора, резистора).

*Наведення і перешкоди* – це сигнали, що приходять на електронну систему ззовні і спотворюють корисний сигнал (наприклад, електромагнітні випромінювання від радіопередавачів або від трансформаторів).

Для переходу від однієї форми сигналу і іншу існують спеціальні перетворювачі – цифро-аналогові (ЦАП) та аналогово-цифрові (АЦП). На рис. 9 приведена функціональна схема електронної системи з використанням обох перетворювачів для фіксації, обробки та відтворення аудіосигналу.

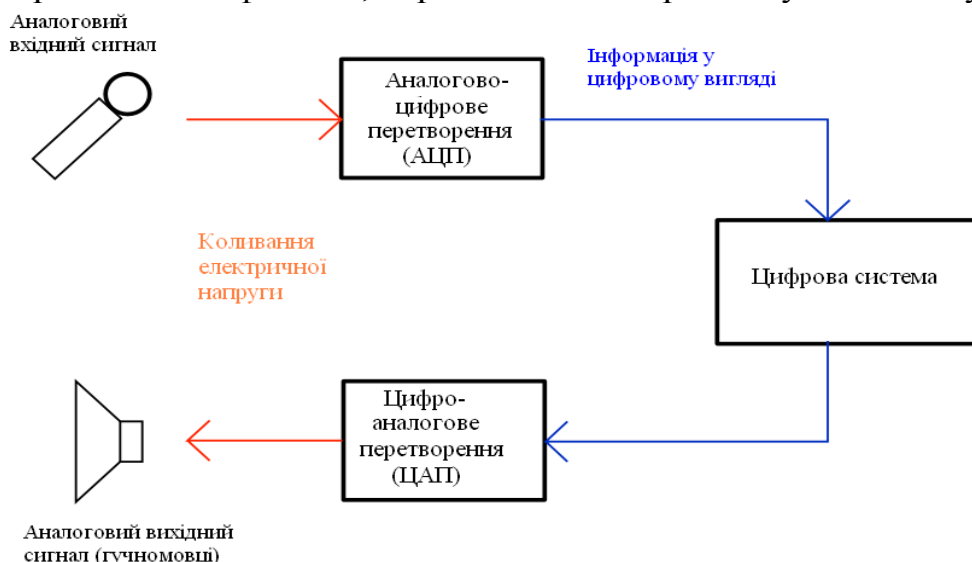


Рис. 9. Приклад цифро-аналогової системи передачі аудіосигналу

Процес дискретизації (кодування, «оцифровки») сигналу АЦП може відбуватися різними способами. На рис. 10 представлені способи кодування безперервного сигналу прямокутними імпульсами, що отримав назву модуляції. Розрізняють три основні види модуляції :

– амплітудно-імпульсна модуляція (АІМ), коли амплітуда імпульсів пропорційна вхідному сигналу (б);

– широтно-імпульсна модуляція (ШІМ), коли ширина імпульсів пропорційна вхідному сигналу, амплітуда і частота імпульсів постійні (в);

– частотно-імпульсна модуляція (ЧІМ), коли вхідний сигнал визначає частоту проходження імпульсів, які мають постійну тривалість і амплітуду (г).

У самому загальному випадку цифровий режим роботи – це коли інформація передається у вигляді числа, якому відповідає певний набір імпульсів (цифровий код) при цьому важлива тільки наявність або відсутність імпульсу.

Цифрові пристрої найчастіше працюють тільки з двома значеннями сигналів – нулем «0» (зазвичай низький рівень напруги або відсутність імпульсу) і «1» (зазвичай високий рівень напруги або наявність прямокутного імпульсу) тобто інформація представляється в двійковій системі числення.

Це обумовлено зручністю створення, обробки, зберігання та передачі сигналів, представлених у двійковій системі: ключ замкнутий – розімкнутий, транзистор відкритий – закритий, конденсатор заряджений – розряджений, магнітний матеріал намагнічений – розмагнічений і т.д.

За виглядом електричних сигналів на діючих виходах логічних елементів у різних дискретних пристроях застосовуються три способи подання інформації електричними сигналами: *потенційний або статичний* (а, двом значенням логічної величини «0» і «1» відповідають низький і високий рівні потенціалу (напруги)); *імпульсний* (б, одиничному і нульовому значенням двійкової змінної відповідають наявність (відсутність) електричного імпульсів в тактові моменти часу); *динамічний або потенціально-імпульсний* (в, двом можливим значенням змінної відповідає наявність або відсутність періодичної серії сигналів відносно тактового моменту).

*Кодування сигналів.* На практиці допустимими є значні зміни рівнів одиничного і нульового сигналів, при яких не виникає небезпеки помилкової оцінки значень змінної.

У зв'язку з цим залежно від кодування сигналів розрізняють позитивну і негативну логіку. При позитивній логіці високому рівню напруг ставлять у відповідність «1», при негативній – навпаки.

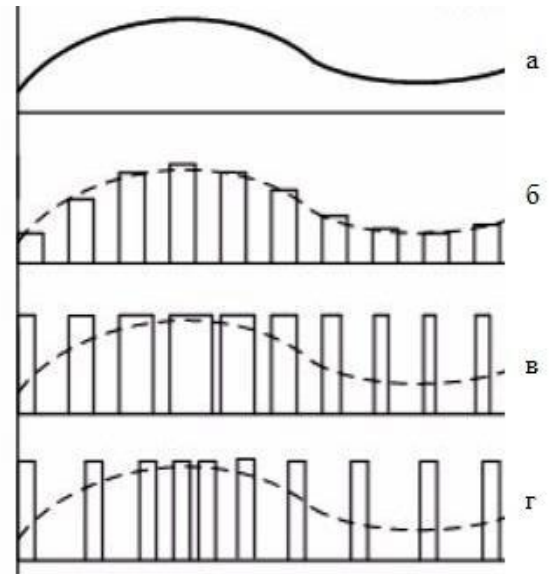
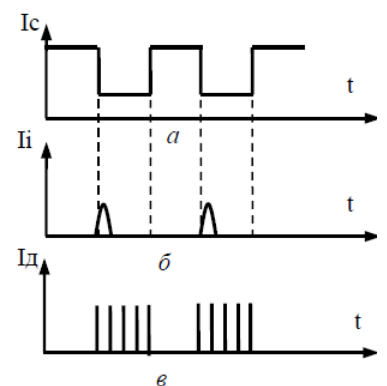
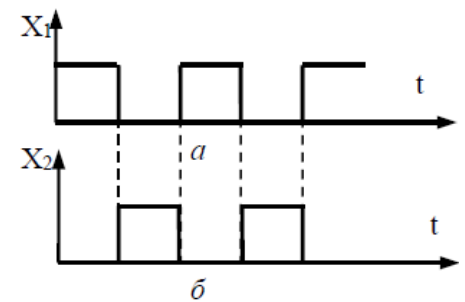


Рис. 10. Способи модуляції сигналу : а – без модуляції. б – АІМ. в – ШІМ. г – ЧІМ



Вибір логіки при кодуванні логічної інформації має велике значення, оскільки це приводить до різного тлумачення логічної функції, яку може реалізувати логічний елемент.

Часто при реалізації різних логічних схем з декількома входами і виходами (особливо схем, тригерів) зустрічається так зване парафазне подання сигналу, яке є одним з видів потенційного способу подання інформації. На рис. показаний спосіб завдання парафазного сигналу для входів  $x_1$  та  $x_2$ .



На практиці допустимими є значні зміни рівнів одиничного і нульового сигналів, при яких не виникає небезпеки помилкової оцінки значень змінної.

У зв'язку з цим залежно від кодування сигналів розрізняють позитивну і негативну логіку. При позитивній логіці високому рівню напруг ставлять у відповідність "1", при негативній – навпаки.

Вибір логіки при кодуванні логічної інформації має велике значення, оскільки це приводить до різного тлумачення логічної функції, яку може реалізувати логічний елемент.

*Кодування* – це процес перетворення повідомлення у форму, зручну для передачі по даному каналу. Наприклад, передачу повідомлення у вигляді телеграми за допомогою телеграфного коду.

*Декодування* – операція відновлення прийнятого повідомлення. У систему зв'язку необхідно ввести пристрої для кодування і декодування інформації.

*Вимірювання параметрів імпульсів* є процесом визначення числових значень ряду параметрів – показників:

$A$  – амплітуда (найбільше значення напруги або струму);

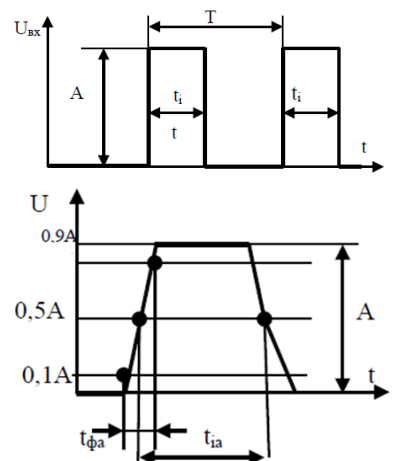
$t_{ia}$  – активна тривалість імпульсу (вимірювана на рівні половини амплітуди);

$t_{фа}$  – активна тривалість фронту (вимірювана між рівнями 0,1 і 0,9 амплітуди);

$T$  – період проходження імпульсів (час, відлічений між моментами появи фронтів або максимальних значень);

$f$  – частота проходження імпульсів ( $f = 1 / T$ );

$\Theta$  – шпаруватість (скважність – рос.) імпульсного процесу ( $\Theta = T / t_{ia}$ ), для імпульсу меандр  $\Theta=2$ .



## 5. Контрольні питання

1. Що таке випрямлячі і для чого вони потрібні?
2. Які види випрямлячів ви знаєте?
3. У чому відмінність між двопівперіодної із середньою точкою і мостовою схемами?
4. Як залежить напруга пульсацій випрямляча з фільтром від струму навантаження: напруга пульсацій падає, росте, залишається незмінною?
5. Чи однакові частоти вхідного і вихідного сигналів у схемах одно і двопівперіодного випрямлячів?



6. Які розходження між вхідним і вихідним сигналами одно і двопівперіодних випрямлячів?

7. Які фактори впливають на величину коефіцієнта пульсацій випрямляча з ємнісним фільтром на виході?

### Тема 3. Логічні основи цифрової схемотехніки

#### Практичне заняття №2. Розрахунок логічних схем

##### Навчальна мета заняття:

1. Закріплення теоретичного матеріалу, набуття навиків аналізу та синтезу простих логічних пристроїв.

2. Закріплення теоретичного матеріалу, набуття навиків створення і моделювання схем цифрової схемотехніки, робота з різними вимірювальними приладами.

**Кількість годин:** 4 год.

##### Література:

1. Рябенський В.М. Жуйков В.Я. Ямненко Ю.С. Заграничний А.В. Схемотехніка: Пристрої цифрової електроніки. Київ: 2016, КПІ. 399 с.

2. Кравець В.О., Сокол Є.І., Рисований О.М. Комп'ютерна схемотехніка. Підручник. Харків: НТУ «ХПІ», 2007. 480 с.

**Матеріально-технічне забезпечення:** комп'ютерна мережа із системою модулювання електронних схем Electronics Workbench.

##### Навчальні питання:

1. Розв'язання завдань щодо мінімізації логічних функцій..... 14
2. Індивідуальні завдання на тему «ЛОГІЧНІ ОСНОВИ ЦИФРОВОЇ СХЕМОТЕХНІКИ»..... 18
3. Контрольні питання ..... 25

#### 1. Розв'язання завдань щодо мінімізації логічних функцій

**Завдання 1.** Перетворимо до ВДНФ логічну функцію:  $f(x) = \overline{x_1}x_2 \vee x_2\overline{x_3}$ . Оскільки функція залежить від трьох змінних  $x_1, x_2, x_3$ , першу кон'юнкцію помножимо на вираз  $(x_3 \vee \overline{x_3})$ , а другу – на  $(x_1 \vee \overline{x_1})$ :

$$f(x) = \overline{x_1}x_2(x_3 \vee \overline{x_3}) \vee (x_1 \vee \overline{x_1})x_2\overline{x_3} = \overline{x_1}x_2x_3 \vee \overline{x_1}x_2\overline{x_3} \vee x_1x_2\overline{x_3} \vee \overline{x_1}x_2\overline{x_3} = \overline{x_1}x_2x_3 \vee \overline{x_1}x_2\overline{x_3} \vee x_1x_2\overline{x_3}$$

Характерною властивістю ВДНФ є те, що представлення в ній логічної функції єдині. Звідси виходить, що кожна ВДНФ містить стільки конститuent одиниці, скільки одиничних наборів має логічна функція. Так, функція, розглянута в прикладі, задана на трьох одиничних наборах, отже, її ВДНФ має три конституенти одиниці. Логічна функція константи одиниці у ВДНФ

представляється диз'юнкцією  $2^n$  конститuent одиниці.

**Завдання 2.** Логічна функція трьох змінних задана табл. 1.  
 $f(x_1, x_2, x_3) = K_1(1) + K_3(1) + K_6(1) + K_7(1) = \overline{x_1}\overline{x_2}x_3 \vee \overline{x_1}x_2\overline{x_3} \vee x_1x_2\overline{x_3} \vee x_1x_2x_3$ .

Кон'юнктивна нормальна форма називається вдосконаленою, якщо кожна елементарна диз'юнкція, що входить до неї, має в прямому або інверсійному вигляді всі змінні, від яких залежить функція.

Характерною властивістю вдосконаленої КНФ є те, що

Таблиця 1

$x_1$	$x_2$	$x_3$	$f(x_1x_2x_3)$
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

представлення в ній логічної функції єдині. Елементарні диз'юнкції, які входять у вдосконалену КНФ функції, носять назву конституент нуля. Кожна конституента нуля, що входить у вдосконалену КНФ, набуває нульового значення на єдиному наборі значень змінних, який є нульовим набором функції. Отже, кількість нульових наборів логічної функції збігається з кількістю конституент нуля, які входять в її вдосконалену КНФ.

Для кожного рядка таблиці істинності, у якій функція дорівнює нулю, складається елементарна диз'юнкція всіх змінних. При цьому в диз'юнкцію входить сама змінна, якщо її значення дорівнює нулю, або заперечення, якщо його значення дорівнює одиниці. Одержані елементарні диз'юнкції об'єднуються знаком кон'юнкції.

$$f(x_1, x_2, x_3) = K_0(0) + K_2(0) + K_4(0) + K_5(0) = (x_1 \vee x_2 \vee x_3)(x_1 \vee \bar{x}_2 \vee x_3)(\bar{x}_1 \vee x_2 \vee x_3)(\bar{x}_1 \vee x_2 \vee \bar{x}_3)$$

Слід зазначити, що для функцій, кількість одиничних наборів яких перевищує кількість нульових наборів, більш компактним є запис у вигляді ВКНФ і навпаки.

**Завдання 3.** Представити логічну функцію  $f(x_1, x_2, x_3, x_4) = \overline{x_1 x_2 x_3 x_4}$  у ДНФ.

Послідовно виконуючи теореми де Моргана перетворимо функцію в ДНФ:

$$f(x_1, x_2, x_3, x_4) = \overline{x_1 x_2 x_3 x_4} = \overline{x_1} + \overline{x_2 x_3 x_4} = \overline{x_1} + \overline{x_2} \overline{x_3 x_4} = \overline{x_1} + \overline{x_2}(\overline{x_3} + \overline{x_4}) = \overline{x_1} + \overline{x_2} \overline{x_3} + \overline{x_2} \overline{x_4}$$

**Завдання 4.** Логічну функцію, яка подана табл.2 істинності виразити у ВДНФ та ВКНФ, мінімізувати їх методом безпосередніх перетворень. Побудувати функціональні схеми, які реалізують ці функції у загальному базисі та базисах І-НІ та АБО-НІ.

Таблиця 2

$x_1$	$x_2$	$x_3$	$f(x_1, x_2, x_3)$	$x_1$	$x_2$	$x_3$	$f(x_1, x_2, x_3)$
0	0	0	0	1	0	0	1
0	0	1	1	1	0	1	1
0	1	0	0	1	1	0	0
0	1	1	1	1	1	1	0

Представимо спочатку функцію у ВДНФ, тобто визначимо її одиничні значення за допомогою конституент одиниць.

$$F(x_1, x_2, x_3) = \Sigma(1, 3, 4, 5) = \overline{x_1} \overline{x_2} \overline{x_3} + \overline{x_1} x_2 \overline{x_3} + x_1 \overline{x_2} \overline{x_3} + x_1 \overline{x_2} x_3$$

Мінімізуємо цю функцію методом безпосередніх перетворень:

$$F(x_1, x_2, x_3) = \overline{x_1} x_3 (\overline{x_2} + x_2) + x_1 \overline{x_2} (x_3 + \overline{x_3}) = \overline{x_1} x_3 + x_1 \overline{x_2}$$

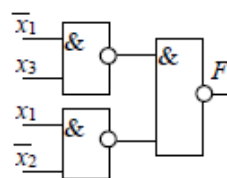
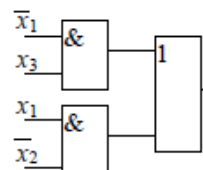
Далі побудуємо функціональну схему, яка реалізує цю функцію в основному базисі.

Щоб реалізувати цю ж функцію в базисі І-НІ, необхідно ДНФ функції надати до такої форми, яка б не вміщала операції логічного додавання. Для цього необхідно від функції взяти подвійне заперечення (що не змінює його значення) та за теоремою де Моргана виконати перетворення:

$$F(x_1, x_2, x_3) = \overline{\overline{\overline{x_1} x_3 + x_1 \overline{x_2}}} = \overline{\overline{x_1} x_3} \overline{x_1 \overline{x_2}} \quad \text{Цей логічний вираз вміщує тільки операції І-НІ.}$$

Побудуємо його функціональну схему.

Представимо цю ж функцію у ВКНФ. Це означає, що потрібно визначити її за нульовими наборами за допомогою конституент нуля:



$$F(x_1, x_2, x_3) = \Pi_0(0, 2, 6, 7) = (x_1 + x_2 + x_3)(x_1 + \bar{x}_2 + x_3)(\bar{x}_1 + \bar{x}_2 + x_3)(\bar{x}_1 + \bar{x}_2 + \bar{x}_3).$$

Для мінімізації отриманої функції використаємо закон склеювання:

$$(x_1 + x_2 + x_3)(x_1 + \bar{x}_2 + x_3) = (x_1 + x_3);$$

$$(\bar{x}_1 + \bar{x}_2 + x_3)(\bar{x}_1 + \bar{x}_2 + \bar{x}_3) = (\bar{x}_1 + \bar{x}_2).$$

Побудуємо функціональну схему, яка реалізує цю функцію в основному базисі.

Для побудови цієї функції у базисі АБО-НИ необхідно в КНФ позбавитись операції кон'юнкції. Ця операція відбувається при використанні подвійного заперечення від функції за допомогою теореми де-Моргана:

$$F(x_1, x_2, x_3) = \overline{\overline{(x_1 + x_3)(\bar{x}_1 + \bar{x}_2)}} = \overline{\overline{x_1 + x_3} + \overline{\bar{x}_1 + \bar{x}_2}}.$$

Будуємо функціональну схему в базисі АБО-НИ:

Завдання 5. Навести у мінімальній ДНФ та мінімальній КНФ логічну функцію трьох змінних, які подані у таблиці істинності.

Таблиця 3

x <sub>1</sub>	x <sub>2</sub>	x <sub>3</sub>	f(x <sub>1</sub> , x <sub>2</sub> , x <sub>3</sub> )	x <sub>1</sub>	x <sub>2</sub>	x <sub>3</sub>	f(x <sub>1</sub> , x <sub>2</sub> , x <sub>3</sub> )
0	0	0	0	1	0	0	1
0	0	1	0	1	0	1	1
0	1	0	0	1	1	0	0
0	1	1	0	1	1	1	0

Запишемо функцію у ВДНФ та мінімізуємо її:

$$F(x_1, x_2, x_3) = x_1 \bar{x}_2 \bar{x}_3 + x_1 \bar{x}_2 x_3 = x_1 \bar{x}_2.$$

Представимо цю ж функцію у ВКНФ і також мінімізуємо її:

$$F(x_1, x_2, x_3) = (x_1 + x_2 + x_3)(x_1 + x_2 + \bar{x}_3)(x_1 + \bar{x}_2 + x_3)(x_1 + \bar{x}_2 + \bar{x}_3) + (x_1 + \bar{x}_2 + x_3)(\bar{x}_1 + \bar{x}_2 + x_3) = (x_1 + x_2)(x_1 + x_2)(x_1 + \bar{x}_2) = x_1 \bar{x}_2.$$

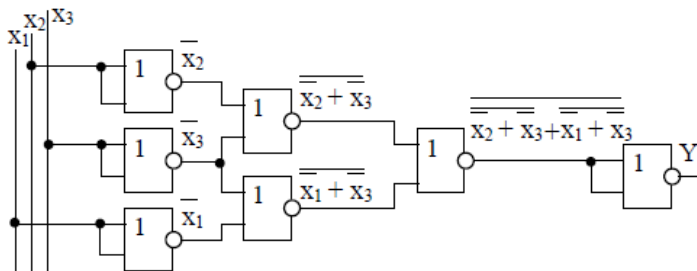
Наведені приклади дозволяють ще раз переконатися, що функції у ДНФ та КНФ дозволяють отримати одні й ті ж результати.

Завдання 6. Мінімізувати логічну функцію та привести до базису АБО-НИ. Побудувати функціональну схему на двовходових елементах.

$$Y = \bar{x}_1 x_2 x_3 + x_1 \bar{x}_2 x_3 + x_1 x_2 \bar{x}_3 = \bar{x}_1 x_2 x_3 + x_1 \bar{x}_2 x_3 + x_1 x_2 \bar{x}_3 + x_1 x_2 x_3 =$$

$$= \overline{\overline{x_2 x_3} + \overline{x_1 x_3}} = \overline{\overline{x_2 + x_3}(\bar{x}_1 + \bar{x}_3)} = \overline{\overline{x_2 + x_3} + \overline{\bar{x}_1 + \bar{x}_3}}.$$

Функціональна схема, яка реалізує функцію, наведена на рисунку:



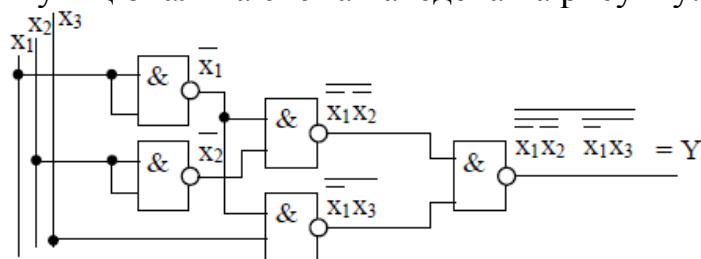
Завдання 7. Мінімізувати логічну функцію та привести до базису І-НИ. Побудувати функціональну схему на двовходових елементах.



$$Y = \overline{x_1}\overline{x_2}\overline{x_3} + \overline{x_1}x_2x_3 + \overline{x_1}\overline{x_2}x_3 = \overline{x_1}\overline{x_2}\overline{x_3} + \overline{x_1}x_2x_3 + \overline{x_1}\overline{x_2}x_3 =$$

$$= \overline{x_1}\overline{x_2}(\overline{x_3} + x_3) + \overline{x_1}x_2(\overline{x_3} + x_3) = \overline{x_1}\overline{x_2} + \overline{x_1}x_2 = \overline{x_1}(\overline{x_2} + x_2) = \overline{x_1} = \overline{x_1x_2} \quad \overline{x_1x_3}.$$

Функціональна схема наведена на рисунку:



Завдання 8. Мінімізувати логічну функцію та привести до базису І-НІ.

$$Y = \overline{x_1}\overline{x_2}x_3 + \overline{x_1}x_2x_3 + x_1\overline{x_2}x_3 = x_2x_3 + \overline{x_1}x_2x_3 = x_2(x_3 + \overline{x_1}x_3) =$$

$$= x_2(\overline{x_3} + \overline{x_1}) = x_2\overline{x_3} + x_2\overline{x_1} = \overline{x_2x_3} \quad \overline{x_2x_1}.$$

Завдання 9. Мінімізувати логічну функцію та привести до базису АБО-НІ.

$$Y = x_1x_3x_4 + x_2x_3x_4 + x_1x_3\overline{x_4} + x_2x_4 = x_1x_3x_4 + x_4(x_2x_3 + x_2) + x_1x_3\overline{x_4} =$$

$$= x_1x_3x_4 + x_4(x_3 + \overline{x_2}) + x_1x_3\overline{x_4} = x_1x_3x_4 + x_3x_4 + \overline{x_2}x_4 + x_1x_3\overline{x_4} =$$

$$= x_3x_4 + \overline{x_2}x_4 + x_1x_3\overline{x_4} = \overline{x_2}x_4 + x_3x_4x_1 = \overline{x_2}x_4 + \overline{x_3x_4x_1} = \overline{x_2} + \overline{x_4} + \overline{x_3} + \overline{x_4} + \overline{x_1}.$$

Завдання 10. Приклади використання карт Карно.

Для визначення мінімальної ДНФ за допомогою карт Карно необхідно поєднати усі клітинки, які вміщують одиницю, у замкнені прямокутні області (контури). У цих контурах кількість клітинок у кожній може дорівнювати  $2^k$ , де  $k = 0, 1, 2, 3, \dots, n$ . Зазначені області можуть перетинатися, і одна й та сама клітинка може належати кільком областям. Але слід пам'ятати, що будь-який контур повинен мати хоча б одну клітинку, яка належить виключно йому.

Кількість таких областей має бути якомога меншою, а кількість клітинок у кожній такій області – якомога більшою. За цих умов кількість аргументів для членів мінімізованої функції буде мінімальною. При охопленні усіх точок карти Карно слід пам'ятати, що права та ліва границі карти – це сусідні стовпчики, так само, як верхній та нижній рядки карти Карно – це сусідні рядки (наприклад, чотири одиниці, розташовані в кутах карти, утворюють один контур).

Слід також зазначити, що якщо для карти Карно кількість нулів значно менша за кількість одиниць, то зручніше склеювати саме їх. Не варто тільки забувати, що у цьому випадку ми отримаємо не саму мінімальну функцію, а її інверсію, яку потім необхідно проінвертувати.

Дві змінних

$$f(x_1, x_2) = (\overline{x_1} + \overline{x_2})(x_1 + \overline{x_2})(x_1 + x_2).$$

$x_1 \backslash x_2$	0	1
0	0	0
1	1	0

$$f(x_1, x_2) = \overline{x_1x_2}.$$

Три змінних

$x_1 \backslash x_2x_3$	00	01	11	10
0	0	0	1	1
1	1	0	0	1

$$f(x_1, x_2) = \overline{x_1}x_2 + x_1x_3.$$

## Чотири змінних

$x_3x_4$	00	01	11	10
$x_1x_2$ 00	1	0	0	1
01	1	0	1	0
11	1	0	1	0
10	1	0	0	1

$$f(x_1, x_2) = \bar{x}_3\bar{x}_4 + \bar{x}_2\bar{x}_4 + x_2x_3x_4.$$

Відомо, що деякі комбінації вхідних змінних логічних функцій з'явитися не можуть (їх значення неістотно). У таких випадках говорять про невизначені набори.

На картах Карно невизначені набори позначаються яким-небудь символом, наприклад, “\*”. Такі елементи можуть довільним чином входити до груп як одиничних, так й нульових об'єднань.

$$f(x) = \Pi_0(2, 4, 5, 6, 7, 10, 11) + \Pi_{\#}(14, 15).$$

$x_2x_1$	00	01	11	10
$x_4x_3$ 00	1	1	1	0
01	0	0	0	0
11	1	1	*	*
10	1	1	0	0

$$f(x) = (x_3 + \bar{x}_4)(x_2 + \bar{x}_1)(x_4 + x_2).$$

## 2. Індивідуальні завдання на тему «Логічні основи цифрової схемотехніки»

Контрольна робота є важливою складовою навчального процесу. Вона доповнює знання курсантів, отриманні на лекційних та лабораторних заняттях.

Синтез логічних схем передбачає побудову структурної схеми пристрою. Визначення складу логічних елементів і з'єднань між ними, при яких забезпечується перетворення вхідних цифрових сигналів у вихідні. Під час синтезу зазвичай виконується мінімізація апаратних витрат на реалізацію пристрою. Синтез можна розділити на такі етапи:

1. Визначення умов функціонування цифрового пристрою. Ці умови можуть бути задані за допомогою таблиць істинності, аналітичних виразів, або словами.
2. Визначення виразу логічної функції. Мінімізація виразу з допомогою метода безпосередніх перетворень або карт Карно (діаграм Вейча).
3. Перетворення мінімізованого виразу логічної функції в заданий базис її схемної реалізації.
4. Складання структурної схеми, тобто зображення потрібних елементів і зв'язків між ними.

Аналіз логічних схем передбачає оцінювання деяких характеристик структури цифрових пристроїв.

Таблиця 6. Вибір варіанту

	$x_1$	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
	$x_2$	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
	$x_3$	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
	$x_4$	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
1		0	0	0	0	0	0	1	0	0	1	0	1	1	1	1	1
2		1	0	1	0	0	0	1	1	0	0	1	0	0	0	1	1
3		1	1	0	1	0	1	0	1	0	1	0	1	0	0	0	0
4		1	0	0	0	1	1	1	1	0	0	0	0	1	0	1	0
5		0	1	0	1	0	1	0	1	0	0	0	0	0	1	1	1

6		0	1	0	0	0	0	0	0	0	1	0	1	1	1	1	1
7		1	1	1	1	0	0	1	1	0	0	0	0	0	0	0	1
8		1	0	1	0	1	0	1	1	0	0	1	0	0	0	1	0
9		1	1	1	1	0	0	1	1	0	1	0	0	0	0	0	0
10		0	0	0	0	1	0	0	0	0	1	0	1	1	1	1	1
11		0	0	1	0	0	0	1	1	1	0	1	0	0	0	1	1
12		0	1	1	1	0	1	0	1	0	1	0	1	0	0	0	0
13		0	0	1	0	1	1	1	1	0	0	0	0	1	0	1	0
14		0	1	0	1	0	1	1	1	0	0	0	0	0	1	0	1
15		0	0	1	0	0	0	1	1	0	0	1	0	1	0	1	1
16		1	1	1	1	0	0	1	1	0	0	0	0	0	0	1	0
17		1	0	1	0	1	0	1	0	0	0	1	1	0	0	1	0
18		1	1	1	1	0	0	1	1	1	0	0	0	0	0	0	0
19		0	0	0	0	0	0	0	1	0	1	0	1	1	1	1	1
20		0	0	1	0	0	1	1	1	0	0	1	0	0	0	1	1
21		0	1	0	1	0	1	0	1	0	1	1	1	0	0	0	0
22		0	0	0	0	1	1	1	1	0	0	1	0	1	0	1	0
23		0	1	0	1	1	1	0	1	0	0	0	0	0	1	0	1
24		0	1	0	1	0	1	1	1	0	1	0	1	0	0	0	0
25		0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	0

Приклад виконання завдання.

1. Формування варіанту завдання відповідно до таблиці істинності (табл. 7).

Таблиця 7

№	$x_1$	$x_2$	$x_3$	$x_4$	$F$
0	0	0	0	0	0
1	0	0	0	1	0
2	0	0	1	0	1
3	0	0	1	1	0
4	0	1	0	0	0
5	0	1	0	1	1
6	0	1	1	0	1
7	0	1	1	1	1
8	1	0	0	0	0
9	1	0	0	1	0
10	1	0	1	0	1
11	1	0	1	1	0
12	1	1	0	0	0
13	1	1	0	1	0
14	1	1	1	0	1
15	1	1	1	1	1

2. Знаходження ВДНФ або ВКНФ.

Для знаходження ВДНФ формуємо табл. 8. Для цього виписуємо з табл. 7 індивідуального завдання ті рядки, в яких функція приймає значення одиниці.

Таблиця 8

№	$x_1$	$x_2$	$x_3$	$x_4$	$F$
2	0	0	1	0	1
5	0	1	0	1	1
6	0	1	1	0	1
7	0	1	1	1	1
10	1	0	1	0	1

14	1	1	1	0	1
15	1	1	1	1	1

Знаходимо конституенти одиниці для цих наборів. Одержуємо їх як кон'юнкцію  $n$  змінних, що входять до аналітичного виразу у прямому виді, якщо значення даної змінної в наборі  $x_i = 1$ , або в інверсному вигляді, якщо  $x_i = 0$ . Таким чином, отримуємо:

для набору 2 –  $K_2 = (\bar{x}_1 \cdot \bar{x}_2 \cdot x_3 \cdot \bar{x}_4)$ ;

для набору 5 –  $K_5 = (\bar{x}_1 \cdot x_2 \cdot \bar{x}_3 \cdot x_4)$ ;

для набору 6 –  $K_6 = (\bar{x}_1 \cdot x_2 \cdot x_3 \cdot \bar{x}_4)$ ;

для набору 7 –  $K_7 = (\bar{x}_1 \cdot x_2 \cdot x_3 \cdot x_4)$ ;

для набору 10 –  $K_{10} = (x_1 \cdot \bar{x}_2 \cdot x_3 \cdot \bar{x}_4)$ ;

для набору 14 –  $K_{14} = (x_1 \cdot x_2 \cdot x_3 \cdot \bar{x}_4)$ ;

для набору 15 –  $K_{15} = (x_1 \cdot x_2 \cdot x_3 \cdot x_4)$ .

Об'єднуємо конституенти одиниці цих наборів, використовуючи операцію логічної суми  $K_2 \vee K_5 \vee K_6 \vee K_7 \vee K_{10} \vee K_{14} \vee K_{15}$ . Це буде аналітичний вираз для функції  $F$ . Таке подання функції має назву "досконала диз'юнктивна нормальна форма" (ВДНФ). Записуємо отриманий вираз у вигляді

$$F_{\text{ВДНФ}} = (\bar{x}_1 \cdot \bar{x}_2 \cdot x_3 \cdot \bar{x}_4) \vee (\bar{x}_1 \cdot x_2 \cdot \bar{x}_3 \cdot x_4) \vee (\bar{x}_1 \cdot x_2 \cdot x_3 \cdot \bar{x}_4) \vee \\ \vee (\bar{x}_1 \cdot x_2 \cdot x_3 \cdot x_4) \vee (x_1 \cdot \bar{x}_2 \cdot x_3 \cdot \bar{x}_4) \vee (x_1 \cdot x_2 \cdot x_3 \cdot \bar{x}_4) \vee (x_1 \cdot x_2 \cdot x_3 \cdot x_4)$$

Для знаходження ДКНФ формуємо табл. 9. Для цього виписуємо з табл. 1 індивідуального завдання ті рядки, в яких функція приймає значення нуля.

Таблиця 9

№	$x_1$	$x_2$	$x_3$	$x_4$	$F$
0	0	0	0	0	0
1	0	0	0	1	0
3	0	0	1	1	0
4	0	1	0	0	0
8	1	0	0	0	0
9	1	0	0	1	0
11	1	0	1	1	0
12	1	1	0	0	0
13	1	1	0	1	0

Знаходимо конституенти 0 для цих наборів. Одержуємо їх як диз'юнкцію усіх змінних, що входять до аналітичного виразу у прямому вигляді, коли значення даної змінної в наборі  $x_i = 0$ , або в інверсному вигляді, якщо значення  $x_i = 1$ . Таким чином, отримуємо:

для набору 0 –  $M_0 = x_1 \vee x_2 \vee x_3 \vee x_4$ ;

для набору 1 –  $M_1 = x_1 \vee x_2 \vee x_3 \vee \bar{x}_4$ ;

для набору 3 –  $M_3 = x_1 \vee x_2 \vee \bar{x}_3 \vee \bar{x}_4$ ;

для набору 4 –  $M_4 = x_1 \vee \bar{x}_2 \vee x_3 \vee x_4$ ;

для набору 8 –  $M_8 = \bar{x}_1 \vee x_2 \vee x_3 \vee x_4$ ;

для набору 9 –  $M_9 = \bar{x}_1 \vee x_2 \vee x_3 \vee \bar{x}_4$ ;

для набору 11 –  $M_{11} = \bar{x}_1 \vee x_2 \vee \bar{x}_3 \vee \bar{x}_4$ ;

для набору 12 –  $M_{12} = \bar{x}_1 \vee \bar{x}_2 \vee x_3 \vee x_4$ ;

для набору 13 –  $M_{13} = \bar{x}_1 \vee \bar{x}_2 \vee x_3 \vee \bar{x}_4$ .

Об'єднуємо конституенти нуля тих наборів, де функція приймає значення нуля. Для цього використовуємо операцію логічного добутку. В результаті отримуємо досконалу кон'юнктивну нормальну форму функції  $F$  у вигляді

$$F_{\text{ДКНФ}} = (x_1 \vee x_2 \vee x_3 \vee x_4) \cdot (x_1 \vee x_2 \vee x_3 \vee \bar{x}_4) \cdot (x_1 \vee x_2 \vee \bar{x}_3 \vee \bar{x}_4) \cdot (x_1 \vee \bar{x}_2 \vee x_3 \vee x_4) \cdot (\bar{x}_1 \vee x_2 \vee x_3 \vee x_4) \cdot (\bar{x}_1 \vee x_2 \vee x_3 \vee \bar{x}_4) \cdot (\bar{x}_1 \vee x_2 \vee \bar{x}_3 \vee \bar{x}_4) \cdot (\bar{x}_1 \vee \bar{x}_2 \vee x_3 \vee x_4) \cdot (\bar{x}_1 \vee \bar{x}_2 \vee x_3 \vee \bar{x}_4)$$

### 3. Мінімізація булевих функцій.

Знаходження МДНФ. Заповнюємо діаграму Вейча. Для цього в комірках діаграми обраної форми, що зображена на рис. 7а, записуємо одиниці. Це робимо в тих комірках, номери яких збігаються з номерами наборів, де задана функція приймає значення одиниці. В результаті отримуємо робочу діаграму Вейча для знаходження МДНФ, показану на рис. 7б.

	$x_2$				
$x_1$	12	13	9	8	$x_3$
	14	15	11	10	
	6	7	3	2	
	4	5	1	0	
	$x_4$				

а

	$x_2$				
$x_1$					$x_3$
	1	1		1	
	1	1		1	
		1			
	$x_4$				

б

	$x_2$				
$x_1$	0	0	0	0	$x_3$
	1	1	0	1	
	1	1	0	1	
	0	1	0	0	
	$x_4$				

Рис. 7. Діаграма Вейча для знаходження МДНФ (а – форма діаграми Вейча, що використовується при мінімізації, б – заповнена робоча діаграма для заданої функції)

Рис. 8. Формування прямокутних конфігурацій на діаграмі Вейча для мінімізації ДНФ заданої функції

Далі виконуємо операцію склеювання. Склеюванню підлягають прямокутні конфігурації, які заповнені одиницями і містять 2, 4, 8 або 16 комірок. Верхні й нижні рядки, крайні ліві і крайні праві стовпці діаграми ніби склеюються, утворюючи поверхню циліндра. Перша область – поруч розташовані одиниці 6, 7, 14, 15 наборів, утворюють квадрат з чотирьох поруч розташованих елементів. Друга область – одиниці 2, 10, 6, 14 наборів розташовані поруч, якщо розглядати діаграму, як поверхню циліндра. Залишилось розглянути одиницю з 5 набору. Її можливо склеїти лише з одиницею з 7 набору. Тоді утворюється область з двох поруч розташованих елементів (рис.8).

Формула, виходить в результаті мінімізації, містить 3 елементарних кон'юнкцій (за числом прямокутників у покритті). Кожна кон'юнкція містить тільки ті змінні, які не змінюють свого значення в наборах, що склеюються у відповідному прямокутнику. Змінна вибирається у прямому виді, якщо її значення в наборі дорівнює 1, та з запереченням в іншому випадку.

$$F_{\text{МДНФ}} = (x_2 \cdot x_3) \vee (x_3 \cdot \bar{x}_4) \vee (\bar{x}_1 \cdot x_2 \cdot x_4).$$

Знаходження МКНФ. Заповнюємо діаграму Вейча. В ті клітинки номера, яких збігаються з номерами наборів, де функція приймає значення нуля записуємо 0 (рис. 9).

	$x_2$			
$x_1$	12	13	9	8
	14	15	11	10
	6	7	3	2
	4	5	1	0
	$x_4$			

а

	$x_2$			
$x_1$	0	0	0	0
			0	
			0	
	0		0	0
	$x_4$			

б

Рис. 9. Діаграма Вейча для знаходження МКНФ (а – форма діаграми Вейча, що використовується при мінімізації, б – заповнена робоча діаграма для заданої функції)

	$x_2$			
$x_1$	0	0	0	0
	1	1	0	1
	1	1	0	1
	0	1	0	0
	$x_4$			

Рис. 10. Формування прямокутних конфігурацій на діаграмі Вейча для мінімізації КНФ заданої функції

Далі проводимо операцію склеювання. Склеюванню підлягають прямокутні конфігурації, які заповнені одиницями і містять 2, 4, 8 або 16 клітинок. Верхні й нижні рядки, крайні ліві і праві стовпчики діаграми ніби склеюються, створюючи поверхню циліндра (рис. 10). Перша область це верхній рядок, є чотири поруч розташованих нуля, які утворюють прямокутник (набори 8, 9, 12, 13). Друга область – вертикальний рядок з чотирьох нулів (набори 9, 11, 3, 1), які також утворюють прямокутник. Залишається два нуля на 4 та 0 наборах (у кутах) їх можна склеїти з іншими кутами. Тобто в третю область склеюємо кутові комірки (набори 0, 4, 8, 12)

Формула, виходить в результаті мінімізації, містить 3 елементарних диз'юнкції (за числом прямокутників у покритті). Кожна диз'юнкція містить тільки ті змінні, які не змінюють свого значення в наборах, що склеюються у відповідному прямокутнику. Змінна вибирається у прямому виді, якщо її значення в наборі дорівнює 0, та з запереченням в іншому випадку.

$$F_{\text{МКНФ}} = (\bar{x}_1 \vee x_3) \cdot (x_2 \vee \bar{x}_4) \cdot (x_3 \vee x_4).$$

4. Побудова та дослідження схем. Для побудови схеми використаємо отримані вирази для МДНФ та МКНФ. Ці вирази мають операції інверсії, кон'юнкції, диз'юнкції. Відповідно вибираємо логічні елементи, які реалізують ці операції схемотехнічно.

У виразі МДНФ заданої функції  $F_{\text{МДНФ}}$  є три кон'юнкції (операції у дужках). Кожній операції на схемі відповідає логічний елемент «І». Вихід цих елементів з'єднаний з входом елемента «АБО», що відповідає диз'юнкції у виразі для МДНФ заданої функції. На вхід елементів «І» подаємо входні сигнали згідно виразу: на перший  $x_2, x_3$ ; на другий  $\bar{x}_1, x_2, x_4$ ; на третій  $x_3, \bar{x}_4$  (рис. 11).

У виразі МКНФ заданої функції  $F_{\text{МКНФ}}$  є три диз'юнкції (операції у дужках). Кожній операції на схемі відповідає логічний елемент «АБО». Вихід цих елементів з'єднаний з входом елемента «І», що відповідає кон'юнкції у виразі для МКНФ

заданої функції. На вхід елементів «АБО» подаємо вхідні сигнали згідно виразу: на перший  $\bar{x}_1, x_3$ ; на другий  $x_2, \bar{x}_4$ ; на третій  $x_3, x_4$  (рис. 12).

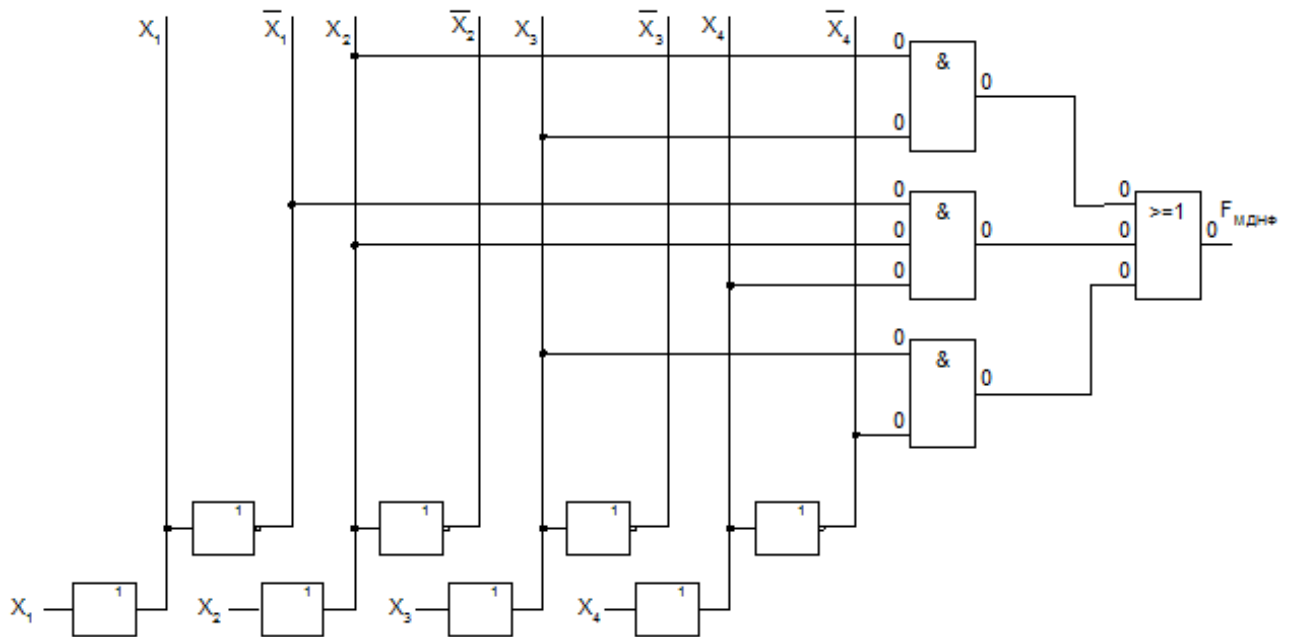


Рис. 11. Схема пристрою реалізації функції МДНФ

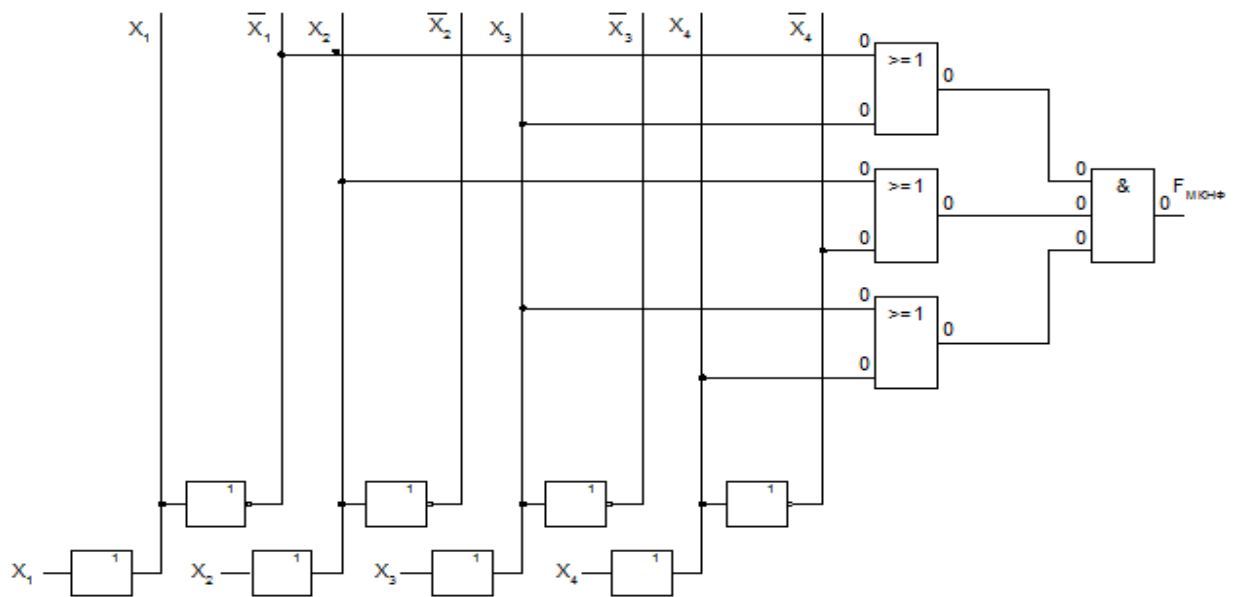


Рис. 12. Схема пристрою реалізації функції МКНФ

5. Побудова та дослідження схеми в базисі І-НІ або АБО-НІ. Для переходу в заданий базис беремо МДНФ. Спочатку ставимо над виразом подвійну інверсію.

$$F_{\text{МДНФ}} = (x_2 \cdot x_3) \vee (x_3 \cdot \bar{x}_4) \vee (\bar{x}_1 \cdot x_2 \cdot x_4) = \overline{\overline{(x_2 \cdot x_3) \vee (x_3 \cdot \bar{x}_4) \vee (\bar{x}_1 \cdot x_2 \cdot x_4)}}$$

Далі користуючись правилом де Моргана ( $\overline{a \vee b} = \bar{a} \cdot \bar{b}$ ), розділяємо одну з операцій інверсії

$$\overline{(x_2 \cdot x_3) \vee (x_3 \cdot \bar{x}_4) \vee (\bar{x}_1 \cdot x_2 \cdot x_4)} = \overline{(x_2 \cdot x_3)} \cdot \overline{(x_3 \cdot \bar{x}_4)} \cdot \overline{(\bar{x}_1 \cdot x_2 \cdot x_4)}$$

Операції між дужками кон'юнкція, над нею є інверсія, тобто це операція Шеффера («І-НІ»)

$$\overline{(x_2 \cdot x_3)} \cdot \overline{(x_3 \cdot \bar{x}_4)} \cdot \overline{(\bar{x}_1 \cdot x_2 \cdot x_4)} = \overline{(x_2 \cdot x_3)} | (x_3 \cdot \bar{x}_4) | (\bar{x}_1 \cdot x_2 \cdot x_4)$$

Операції в дужках кон'юнкції, над кожною дужкою є інверсія, тобто це операції Шеффера («І-НІ»)

$$\overline{(x_2 \cdot x_3)} | \overline{(x_3 \cdot \bar{x}_4)} | \overline{(\bar{x}_1 \cdot x_2 \cdot x_4)} = (x_2 | x_3) | (x_3 | \bar{x}_4) | (\bar{x}_1 | x_2 | x_4)$$

Кожній операції на схемі відповідає логічний елемент «І-НІ». Тому схема складається з чотирьох елементів «І-НІ». На вхід трьох елементів «І-НІ», які відповідають операціям у дужках подаємо вхідні сигнали згідно виразу: на перший  $x_2, x_3$ ; на другий  $\bar{x}_1, x_2, x_4$ ; на третій  $x_3, \bar{x}_4$ . Вихід цих елементів з'єднаний з входами елемента «І-НІ», що відповідає операції між дужками заданої функції (рис. 13).

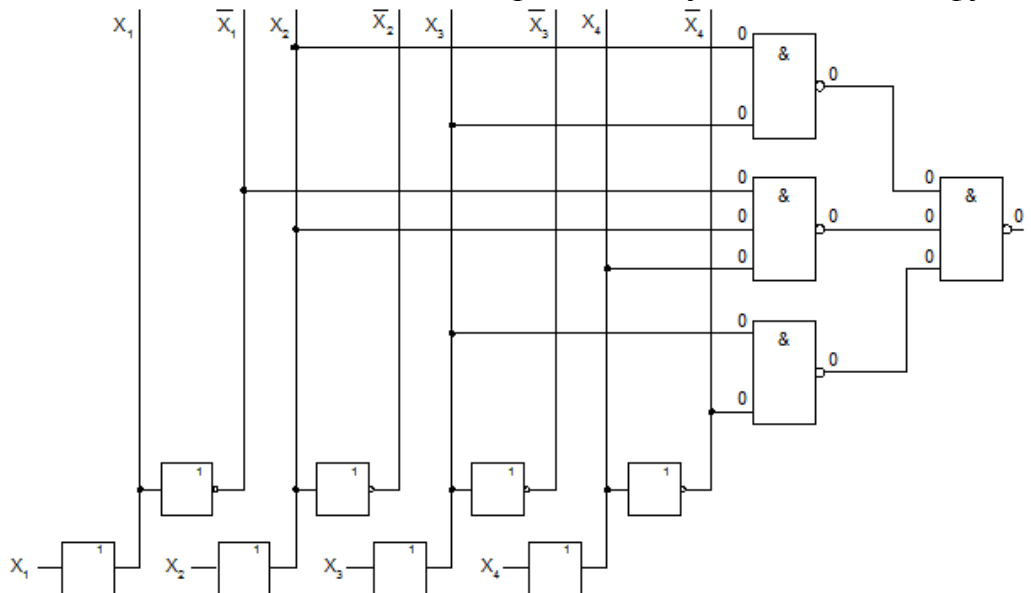


Рис. 13. Схема реалізації пристрою в базисі І-НІ

Для переходу в базис АБО-НІ беремо МКНФ. Спочатку ставимо над виразом подвійну інверсію.

$$F_{\text{МКНФ}} = (\bar{x}_1 \vee x_3) \cdot (x_2 \vee \bar{x}_4) \cdot (x_3 \vee x_4) = \overline{\overline{(\bar{x}_1 \vee x_3)} \cdot \overline{(x_2 \vee \bar{x}_4)} \cdot \overline{(x_3 \vee x_4)}}$$

Далі користуючись правилом де Моргана ( $\overline{a \cdot b} = \bar{a} \vee \bar{b}$ ), розділяємо одну з операцій інверсії

$$\overline{\overline{(\bar{x}_1 \vee x_3)} \cdot \overline{(x_2 \vee \bar{x}_4)} \cdot \overline{(x_3 \vee x_4)}} = \overline{\overline{(\bar{x}_1 \vee x_3)} \vee \overline{(x_2 \vee \bar{x}_4)} \vee \overline{(x_3 \vee x_4)}}$$

Операції між дужками диз'юнкція, над нею є інверсія, тобто це операція Пірса («АБО-НІ»)

$$\overline{\overline{(\bar{x}_1 \vee x_3)} \vee \overline{(x_2 \vee \bar{x}_4)} \vee \overline{(x_3 \vee x_4)}} = \overline{(\bar{x}_1 \vee x_3)} \downarrow \overline{(x_2 \vee \bar{x}_4)} \downarrow \overline{(x_3 \vee x_4)}$$

Операції в дужках диз'юнкції, над кожною дужкою є інверсія, тобто це операції Пірса («АБО-НІ»)

$$= \overline{(\bar{x}_1 \vee x_3)} \downarrow \overline{(x_2 \vee \bar{x}_4)} \downarrow \overline{(x_3 \vee x_4)} = (\bar{x}_1 \downarrow x_3) \downarrow (x_2 \downarrow \bar{x}_4) \downarrow (x_3 \downarrow x_4)$$

Кожній операції на схемі відповідає логічний елемент «АБО-НІ». Тому схема складається з чотирьох елементів «АБО-НІ». На вхід трьох елементів «АБО-НІ», які відповідають операціям у дужках подаємо вхідні сигнали згідно виразу: на перший  $\bar{x}_1, x_3$ ; на другий  $x_2, \bar{x}_4$ ; на третій  $x_3, x_4$ . Вихід цих елементів з'єднаний з входами елемента «АБО-НІ», що відповідає операції між дужками заданої функції (рис. 14).



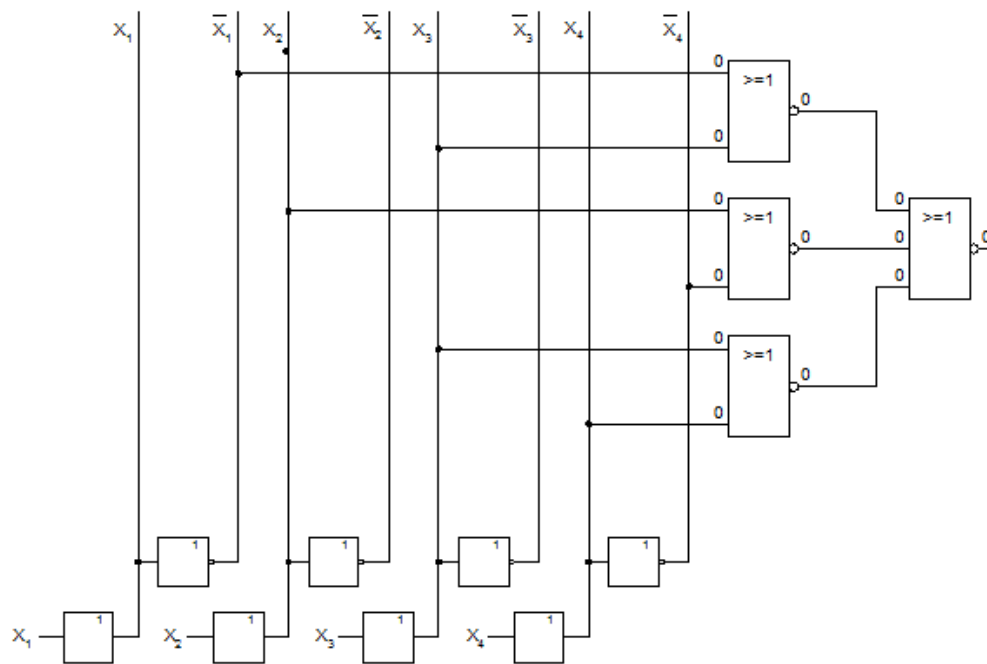


Рис. 14. Схема реалізації пристрою в базисі АБО-НІ

### 3. Контрольні питання

1. Що таке логічна функція?
2. Які існують способи подання логічних функцій?
3. Назвіть основні аксіоми та закони алгебри логіки.
4. Що називають функціонально повною системою логічних функцій?
5. Що таке кон'юнктивна та диз'юнктивна нормальні форми подання логічних функцій?
6. Коли КНФ та ДНФ вважаються досконалими?

### Тема 4. Схемотехніка логічних елементів

#### Практичне заняття №3. Дослідження біполярних і польових транзисторів

##### Навчальна мета заняття:

1. Вивчення принципу роботи, статичних і динамічних характеристик, експериментального дослідження біполярних і польових транзисторів.
2. Закріплення теоретичного матеріалу, набуття навиків створення і моделювання схем аналогових пристроїв, робота з різними вимірювальними приладами.

**Кількість годин:** 4 год.

##### Література:

1. Методичні вказівки до лабораторних робіт з курсу «Моделювання на ЕОМ» для студентів напряму підготовки 6.051001 «Метрологія та інформаційно-вимірювальні технології» денного та заочного навчання / Уклад. Харченко О. Л., Балєв В. М. Харків : НТУ «ХПІ», 2010. 68 с.
2. Крилик Л.В., Селецька О.О. Матеріали електронної техніки: навчальний посібник. Вінниця: ВНТУ, 2017. 120 с
3. Чешко І.В. Вступ до спеціальності «Електроніка» : навчальний посібник. Суми : Сумський державний університет, 2017. 148 с.

4. Квітка С.О., Яковлев В.Ф., Нікітіна О.В. Електроніка та мікросхемотехніка: Навчальний посібник / За ред. проф. В.Ф. Яковлева. Київ: Аграрна освіта, 2010. 329 с.

5. Болюх В. Ф., Данько В. Г. Основи електроніки і мікропроцесорної техніки: Навч. посібник. Харків: НТУ «ХП», 2011. 257 с.

**Матеріально-технічне забезпечення:** комп'ютерна мережа із системою модулювання електронних схем Electronics Workbench.

### Навчальні питання

#### 1. Загальні відомості про біполярні та польові транзистори

*Транзистор* – напівпровідниковий елемент із трьома електродами, що служить для посилення або перемикавання сигналу. Транзистор – активний елемент, що підсилює потужність електричного сигналу. Це посилення відбувається за рахунок енергії зовнішніх джерел живлення. Змінюючи струм у входному ланцюзі за певним законом, можна одержати посилений сигнал на виході тієї ж форми. Транзистори діляться на дві великі групи: біполярні і польові.

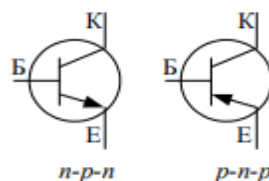


Рис. 1. Позначення біполярних транзисторів

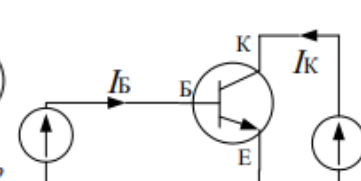


Рис.2. Полярність включення n-p-n транзистора

*Біполярні транзистори.* Розрізняють кремнієві і германієві транзистори. Вони бувають р-п-р і п-р-п типу, на рис. 1 показані їхні позначення. Біполярний транзистор можна розглядати як два протилежно включених діоди, які мають один загальний п- або р-шар. Електрод, пов'язаний з ним, називається базою (Б). Два інших електроди називаються емітером (Е) і колектором (К).

Основна особливість транзистора полягає в тому, що його колекторний струм є кратним базовому струму. Їхнє відношення  $K=I_C/I_B$  називають коефіцієнтом підсилення по струму (рис. 2).

Іншою особливістю є те, що малої зміни входної напруги виявляється досить для того, щоб викликати відносно велику зміну колекторного струму. В електричне коло транзистор включають таким чином, що один з його виводів (електродів) є входним, другий – вихідним, а третій – загальним для входного і вихідного кіл (рис. 3).

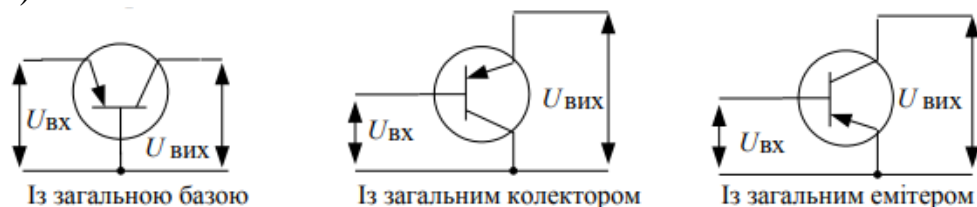


Рис. 3. Схеми включення біполярних транзисторів

Польовий транзистор управляється електричним полем, практично без витрати потужності управляючого сигналу. Серед польових транзисторів розрізняють шість типів, їхні умовні позначки в електричних принципових схемах представлені на рис. 4.

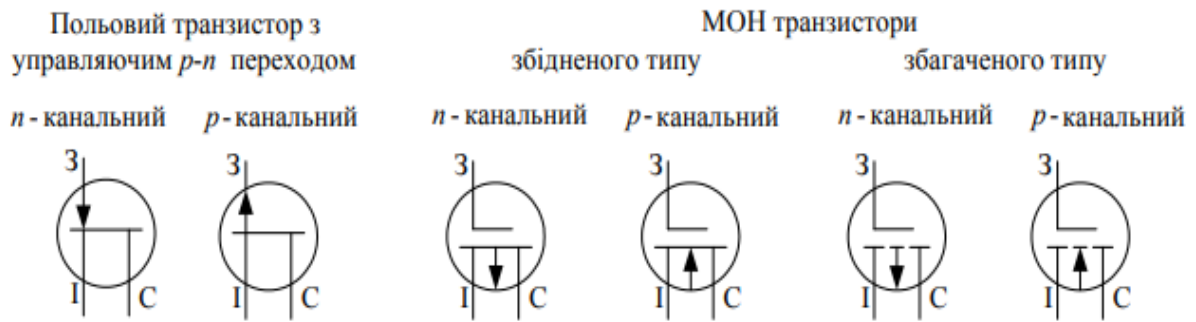


Рис. 4. Позначення польових транзисторів

Керуючим електродом транзистора є затвор 3. Він дозволяє управляти величиною опору між стоком С та істком І. Керуючою напругою є напруга  $U_{з1}$ . Більшість польових транзисторів є симетричними, тобто їхні властивості не змінюються якщо електроди І і С поміняти місцями. Через польовий транзистор з управляючим переходом при напрузі  $U_{з1} = 0$  протікає найбільший струм стоку. Такі транзистори називають нормально відкритими. Аналогічні властивості мають МОН транзистори збідненого типу. Навпаки, МОН транзистори збагаченого типу закриваються при величинах  $U_{з1}$  близьких до нуля. Їх називають нормально закритими.

## 2. Визначення залежності вихідної напруги від вхідної для схеми із загальним емітером

Складіть схему, представлену на рис. 5. Змінюючи вхідну напругу від 0 до 4 В із кроком 0,2 В занесіть у табл. 1 значення вихідної напруги, за отриманими даними побудуйте залежність  $U_{вих} = f(U_{вх})$ .

## 3. Вимірювання коефіцієнта підсилення схеми із загальним емітером і негативним зворотним зв'язком по струму

Складіть схему, представлену на рис. 6. Змінюючи вхідний сигнал від 0 до 10В із кроком 1 В, занесіть у табл. 2 значення вихідної напруги, за отриманими даними побудуйте графік залежності вихідної напруги від вхідної.

Таблиця 2

$U_{вх}, \text{В}$	0	1	...	10
$U_{вих}, \text{В}$				

Таблиця 1

$U_{вх}, \text{В}$	0	0,2	...	4
$U_{вих}, \text{В}$				

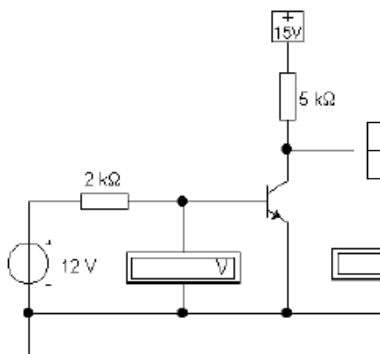


Рис.5. Схема з загальним емітером

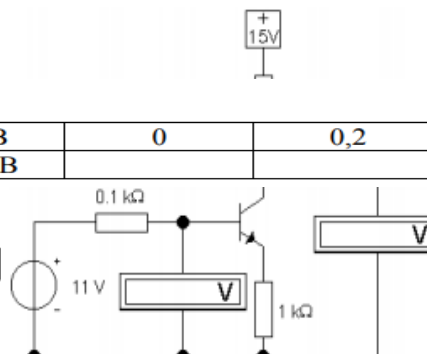


Рис. 6. Схема з загальним емітером та зв. зв'язком

## 4. Визначення залежності вихідної напруги від вхідної для схем із загальною базою і загальним колектором

Складіть схеми, представлені на рис. 7 а, б.

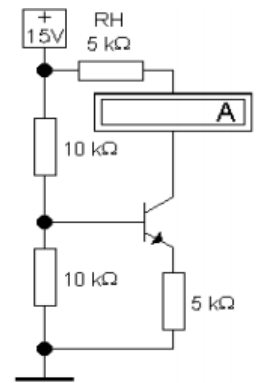
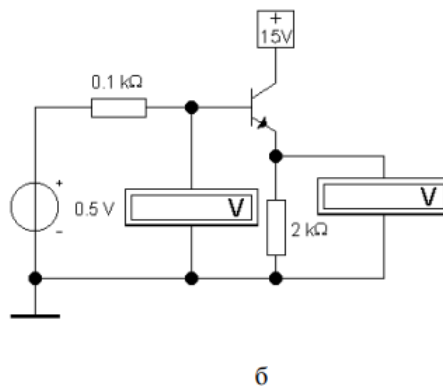
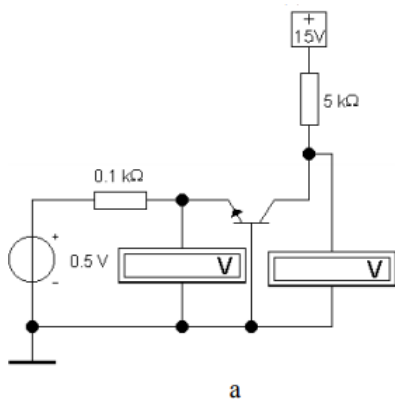


Рис. 7. Схем із загальною базою і загальним колектором

Рис. 8 Джерело струму на транзисторі

Повторіть операції з дослідів 2 для схем із загальною базою і загальним колектором, за результатами досліджень, побудуйте графіки  $U_{вих} = f(U_{вх})$ .

### 5. Транзистор як джерело стабільного струму

Ідеальне джерело струму забезпечує в навантаженні струм, що не залежить від опору навантаження. Схема джерела струму зображеного на рис. 8, побудована на основі схеми із загальним емітером і негативним зворотним зв'язком по струму. Навантаження включається послідовно із транзистором. Вихідний струм залишається незмінним поки транзистор насичений. Складіть схему представлену на рис. 8, змінюючи опір навантаження від 10 Ом до 5 кОм, переконайтеся в незмінності показань амперметра.

### 6. Визначення залежності вихідної напруги від вхідної для схеми із загальним істоком та істокового повторювача

Схема із загальним істоком відповідає схемі із загальним емітером для біполярного транзистора. Схема із загальним стоком (істоковий повторювач) має значно більший опір, чим схема із загальним істоком, крім того, вона істотно зменшує вхідну ємність каскаду (рис. 9).

Змінюючи вхідну напругу від 0 до 14 В із кроком 1. За отриманим даними побудуйте залежність  $U_{вих} = f(U_{вх})$ .

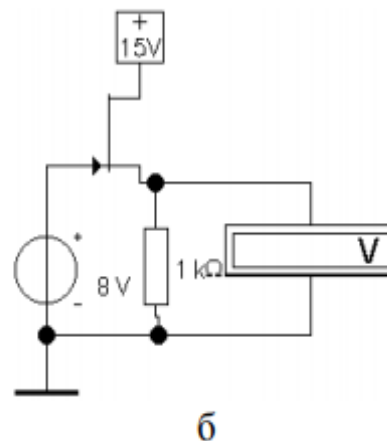
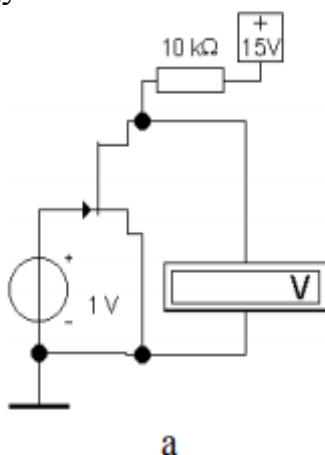


Рис. 9. Схема із загальним істоком (а) та істоковий повторювач (б)

### 7. Керований дільник напруги

При малих напругах стік – істок вольтамперна характеристика така ж, як і в омичного опору, величину якого можна міняти в широких межах шляхом зміни напруги затвор – істок. Мінімальна напруга буде при  $U_{зс} = 0$ . Складіть схему,

представлену на рис. 10. Змінюючи напругу, що подається на затвор транзистора в межах від 0 до 12 В виміряйте вихідну напругу і побудуйте графік отриманої залежності.

### 8. Контрольні питання

1. Назвіть основні типи напівпровідникових приладів та укажіть кількість р-п-переходів у них.

2. Що таке напівпровідниковий діод? Наведіть класифікацію та умовні позначення напівпровідникових діодів.

3. Що таке випрямний діод? Накресліть його умовне позначення і вольт-амперну характеристику. Де застосовуються випрямні діоди?

4. Що таке фото- і світлодіоди? Накресліть їх умовні позначення. Де використовують фото- і світлодіоди?

5. Що таке транзистор? Наведіть класифікацію та умовні позначення транзисторів.

6. Накресліть структуру та умовне позначення біполярних транзисторів р-п-р і п-р-п-типів. Позначте основні елементи.

7. Назвіть і наведіть схеми включення біполярного транзистора.

8. Зобразіть вхідну і вихідну вольт-амперні характеристики біполярного транзистора, включеного за схемою зі спільним емітером.

9. Накресліть структуру, умовне позначення і схему включення польового транзистора з керованими р-п-переходами.

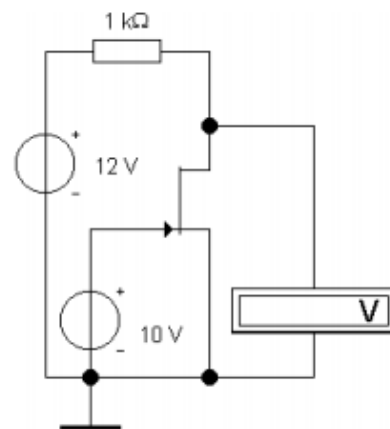


Рис. 10. Керований дільник напруги

## Тема 8. Напівпровідникові запам'ятовуючі пристрої

### Практичне заняття №4. Дослідження постійної та оперативної пам'яті комп'ютера

#### Навчальна мета заняття:

1. Закріплення теоретичного матеріалу, набуття навиків побудови та функціонування модулів пам'яті.

2. Набуття навиків створення і моделювання схем цифрової схемотехніки, робота з різними вимірювальними приладами.

**Кількість годин:** 4 год.

#### Література:

1. Рябенський В.М. Жуйков В.Я. Ямненко Ю.С. Заграничний А.В. Схемотехніка: Пристрої цифрової електроніки. Київ: 2016, КПІ. 399 с.

2. Кравець В.О., Сокол Є.І., Рисований О.М. Комп'ютерна схемотехніка. Підручник. Харків: НТУ «ХПІ», 2007. 480 с.

**Матеріально-технічне забезпечення:** комп'ютерна мережа із системою модулювання електронних схем Electronics Workbench.

#### Навчальні питання

1. Дослідження функціонування модуля оперативної пам'яті..... 30
2. Дослідження функціонування модуля постійної пам'яті..... 32
3. Контрольні питання ..... 32

## 1. Дослідження функціонування модуля оперативної пам'яті

Створити комірку оперативної пам'яті. Підключити на вхід схеми генератор слова, а на вихід – світлодіод чи лампочку. Оформити її у вигляді підсхеми (субблока).

Лабораторна робота виконується по групах. Кожна група повинна побудувати комірки пам'яті статичного та постійного ЗП, а на основі цих субблоків зібрати та дослідити модулі пам'яті згідно з варіантами за табл. 1.

Схема комірки пам'яті з елементами керування наведена на рис. 1. Для зображення підписів елементів треба виділити елемент (клацнути лівою клав'шею миші). Виділений елемент повинен стати червоного кольору. Навести стрілку курсора на елемент та натиснути праву клав'шу миші. Вибрати підменю Component Properties. В ньому вибрати піктограму з назвою Label. Ввести назву елемента. Вікно програми EWB 5.12 зі схемою комірки статичної пам'яті зображене на рис. 2.

Для запису в комірку пам'яті на відповідній розрядній шині даних встановлюється 1 або 0, на вході WR/RD – сигнал 1 та після стробування лічильника або дешифратора адреси сигналом CS включаються елементи 2І: U1, U2. Перепад сигналу з елемента U2 поступає на тактовий вхід D-тригера U4, в результаті чого в нього записується 1 або 0 в залежності від рівня сигналу на його D-вході.



Рис. 1. Схема комірки пам'яті з елементами керування

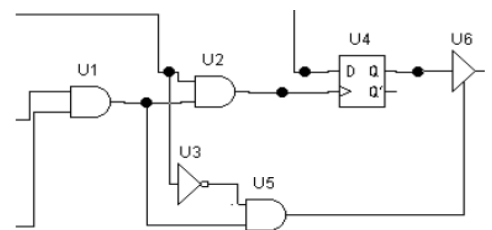


Рис. 2. Вікно програми EWB 5.12 зі схемою комірки статичної пам'яті

При читанні з комірки пам'яті на вході WR/RD встановлюється 0, при цьому спрацьовують елементи U1, U3, U5 та на вхід ДОЗВІЛ ВИХОДУ буферного елемента U6 поступає сигнал дозволу, в результаті чого сигнал Q-виходу D-тригера передається на розрядну шину даних DO0 ... DO3. Для перевірки функціонування комірки пам'яті використовується генератор слова.

Таблиця 2

№ з/п	ОЗП		ПЗП	
	Ряд.	Стовп.	Ряд.	Стовп.
1	2	2	5	2
2	2	3	4	4
3	2	4	4	3
4	3	2	4	2
5	3	3	3	4
6	3	4	3	3
7	4	2	3	2
8	4	3	2	4
9	4	4	2	3
10	5	2	2	2



Відповідно до варіанта завдання побудуйте модуль  $n$ -бітного ОЗП (табл. 1). Для цього потрібно виділити з допомогою лівої клавіші миші таку область схеми, в яку б не потрапили елементи, які не належать до неї. В результаті виконання команди викликається діалогове вікно, в стрічці Name якого треба ввести ім'я підсхеми. Після цього можливі варіанти:

Move from Circuit – виділена частина вирізається з схеми та у вигляді підсхеми копіюється в бібліотеку Custom;

Рис. 3. Комірка статичної пам'яті з використанням підсхеми

The diagram on the left shows a 16-bit OZP matrix, which is a complex interconnection of 16 input lines and 16 output lines. The matrix is composed of 16 small rectangular blocks, each labeled '12', arranged in a 4x4 grid. These blocks are interconnected by a dense network of lines, representing the internal logic of the OZP matrix. The diagram on the right is a schematic detail of the internal structure of one of these blocks, showing a complex arrangement of logic gates and interconnections, including a triangular gate symbol.

Рис. 4. Матриця 16-бітового ОЗП

Для перегляду редагування підсхеми потрібно двічі клацнути мишею по її значку. Редагування підсхеми здійснюється за загальними правилами редагування схем. При створенні додаткового виводу необхідно з відповідної точки підсхеми курсором миші протягнути провідник до краю її вікна до появи незафарбованої

прямокутної контактної площадки, після чого відпустити ліву кнопку миші. Для видалення виводу необхідно курсором миші схопитися за його прямокутну площадку в краю вікна підсхеми і винести її за межі вікна.

Конструктивно будь-яке ОЗП складається з двох блоків – матриці запам'ятовуючих елементів і дешифратора адреси. З технологічних міркувань матриця найчастіше має двокоординатну дешифрацію адреси – за рядками і за стовпцями.

## 2. Дослідження функціонування модуля постійної пам'яті

Створити, в залежності від варіанта, матрицю запам'ятовуючого масиву  $n$ -бітного ПЗП. Для дослідження функціонування матриці підключити відповідні елементи.

Порядок побудови модуля ПЗП аналогічний до порядку побудови модуля ОЗП. Схема нагромаджувача ПЗП наведена на рис. 5.

На першому етапі створення схеми ПЗП необхідно створити схему дешифратора, та налагодити його роботу. Один із варіантів реалізації схеми дешифратора наведено на рис. 6.

Створення субблока дешифратора аналогічне до створення субблока комірки статичної пам'яті та наведене на рис. 7.

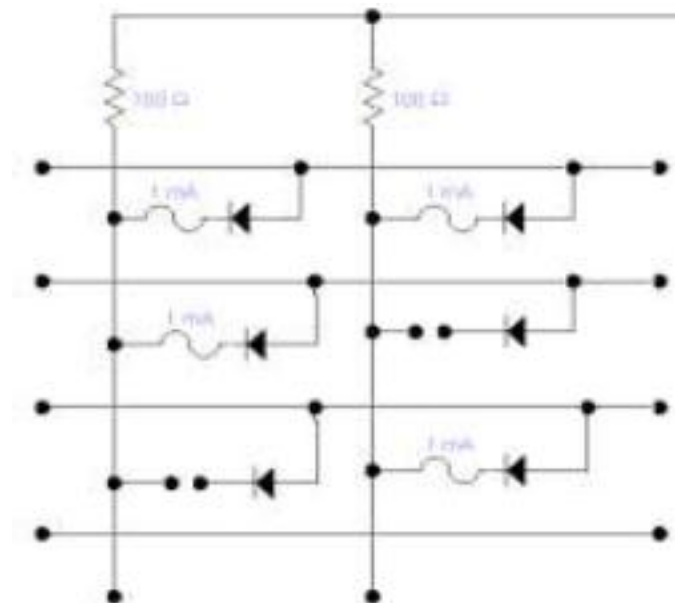


Рис. 5. Схема нагромаджувача ПЗП

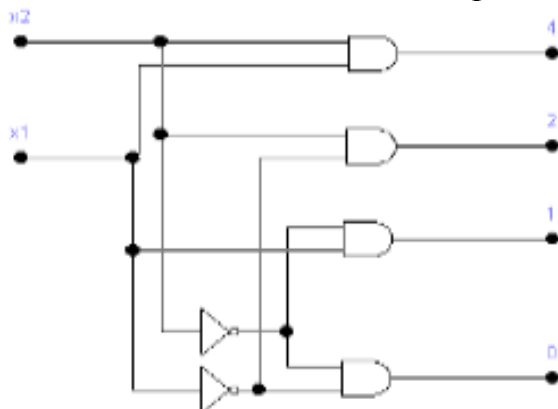


Рис. 6. Схема дешифратора з організацією 2x4

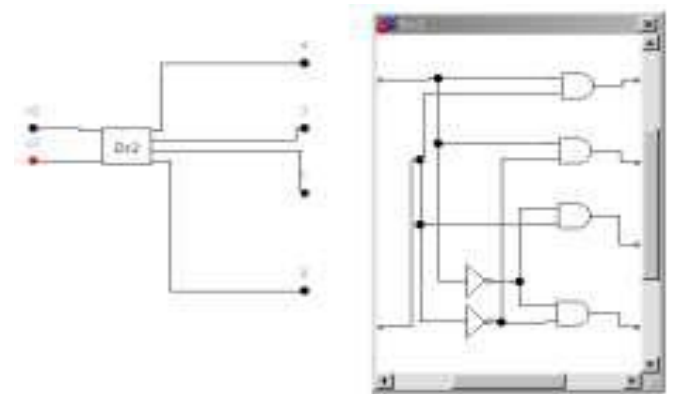


Рис. 7. Схема субблока дешифратора з організацією 2x4

## 3. Контрольні питання

1. Які бувають типи пам'яті?
2. Де в сучасних комп'ютерах використовується пам'ять статичного типу?
3. Чим відрізняється динамічна пам'ять від статичної?
4. Які типи динамічної пам'яті використовуються в сучасних комп'ютерах?
5. Що таке відеопам'ять та як вона пов'язана з характеристиками відображуваної на дисплеї інформації?



## Тема 9. Основи мікропроцесорної техніки

### Практичне заняття №5. Програмування мікроконтролерів

#### Навчальна мета заняття:

1. Закріплення теоретичного матеріалу, набуття навиків програмування мікроконтролерів.

2. Набуття навиків створення і моделювання функціонування мікроконтролерів.

**Кількість годин:** 4 год.

#### Література:

1. Круліковський Б.Б., Николайчук Я.М., Шатний С.В. Мікропроцесорні системи. Практикум. Навчальний посібник. Рівне : НУВГП, 2016. 191 с.

2. Кравець В.О., Сокол Є.І., Рисований О.М. Комп'ютерна схемотехніка. Підручник. Харків: НТУ "ХПІ", 2007. 480 с.

3. Методичні вказівки до лабораторних робіт з дисципліни «Мікропроцесорна техніка» Укл.: В.І. Рева. Запоріжжя: ЗНТУ, 2019. 114 с.

4. Плахтєєв А.П., Бабешко Є.В, Ткаченко В.А., Здоровець Ю.В. Архітектури та розроблення систем Інтернету / Вебу Речей на основі вбудованих платформ. Лабораторні роботи / За ред. В.С. Харченка. Міністерство освіти і науки України, Національний аерокосмічний університет, ХАІ, 2019. 147 с.

5. Мікроконтролери сімейства MCS-51 в задачах обробки інформації та керування. Методичні вказівки до виконання лабораторних робіт з дисципліни «Мікропроцесорні пристрої керування та обробки інформації». Укл.: Войтенко В.П., Хоменко М.А. Чернігів: ЧНТУ, 2014. 71 с.

**Матеріально-технічне забезпечення:** комп'ютерна мережа з інтегрованим середовищем EdSim51 Simulator.

#### Навчальні питання

1. Вивчення алгоритмів додавання та віднімання багатобайтових чисел.....33
2. Програмування операцій додавання та віднімання операндів.....37
3. Порядок роботи в інтегрованому середовищі EdSim51 Simulator .....39
4. Контрольні питання .....42

#### 1. Вивчення алгоритмів додавання та віднімання багатобайтових чисел

Усі команди арифметичних операцій призначені для роботи з додатними цілими числами байтового формату, хоча команди додавання і віднімання у випадку відсутності переповнення забезпечують одержання коректного результату при спеціальному способі кодування від'ємних чисел. При необхідності роботи з числами, що не можуть бути подані в форматі байта, необхідно розробляти відповідні підпрограми. Виконання операцій множення і ділення з числами, що мають довільний знак, можливе, якщо подати їх через знак і модуль та при використанні відповідних підпрограм власної розробки чи запозичені з бібліотек інших розробників.

*Команда додавання ADD працює з даними у форматі байта, при цьому як приймач завжди використовується тільки акумулятор:*

ADD A, #dat

ADD A, Rn

ADD A, @Ri

ADD A, src

Мнемоніка цієї команди відповідає слову ADDition (додавання). Для роботи з числами, що не можуть бути подані у вигляді одного байта, використовується команда додавання, яка враховує біт переносу, отриманий при додаванні попередньої пари байтів – ADDC:

ADDC A, #dat

ADDC A, Rn

ADDC A, @Ri

ADDC A, src

Літера C у позначенні команди вказує на використання біта переносу (ADDition with Carrier).

Набір команд для віднімання набагато менший. Команда обчислення різниці – SUBB існує тільки у варіанті з відніманням вмісту біта переносу:

SUBB A, #dat

SUBB A, Rn

SUBB A, @Ri

SUBB A, src

Мнемоніка цієї команди відповідає словам SUBtraction with Borrow (тобто віднімання з урахуванням позики, тому що при відніманні утворюється позика, а не перенос). З цієї причини перед обчисленням різниці молодших байтів потрібно обов'язково очищати біт переносу, якщо немає впевненості в його вмісті. При обчисленні різниці старших байтів цього робити не потрібно.

Таблиця 1

Додавання двійкових чисел здійснюється у відповідності з табл. 1, де для кожної комбінації вхідних сигналів доданку X, доданку Y та переносу з молодшого розряду  $C_{in}$  вказана відповідна комбінація вихідних сигналів суми S та переносу з поточного в наступний розряд  $C_{out}$ .

$C_{in}$	X	Y	$C_{out}$	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

MCS-51 має команду однобайтового додавання акумулятора A з вказаним операндом з врахуванням переносу ADDC. ADDC використовує для сигналу  $C_{in}$  переносу в нульовий розряд поточне значення прапора переносу CY (або C) з регістру PSW мікроконтролера. Після виконання інструкції ADDC в CY записується  $C_{out7}$  - прапор переносу з старшого 7-го розряду, який може бути використаний в якості  $C_{in0}$  для наступного додавання старших байтів і фактично рахуватись  $C_{in8}$ . Таким чином, повторенням ADDC можна додавати числа з одного, двох і т.д. байтів. Для цього необхідно виконати ADDC для кожних 8 біт доданків починаючи з молодшого, не втрачаючи при цьому значень біту CY між сусідніми командами ADDC. Слід пам'ятати, що перед додаванням молодших байтів чисел, біт CY повинен бути встановленим в 0 за допомогою команди CLR C. Але є більш зручне рішення: команда ADD, в якій значення CY завжди береться рівним 0. Тому при додаванні молодших байтів багатобайтових чисел слід використовувати саме команду ADD.

Необхідно також пам'ятати наступне. При додаванні двох n- розрядних чисел сума може виявитись n+1 -розрядною. Тому, якщо при додаванні двох беззнакових

чисел встановлюється прапор переносу CY, це означає, що для збереження правильної суми необхідний ще один додатковий біт для зберігання цього прапора переносу, що дорівнює 1.

Таким чином, алгоритм add16 додавання двох цілих 16-розрядних чисел можна записати так:

- 1) Завантажити в акумулятор молодший байт доданку А.
- 2) Додати вміст акумулятора до молодшого байту доданку В.
- 3) Зберегти молодший байт суми доданків.
- 4) Завантажити старший байт доданку А в акумулятор.
- 5) Додати до вмісту акумулятора старший байт доданку В з врахуванням переносу із попередньої суми молодших байтів.
- 6) Зберегти старший байт суми.
- 7) В разі потреби зберегти (n+1)-й розряд суми.

Для запису підпрограми необхідно вивчити призначення та порядок виконання команд ADD A, Rn; MOV A, Rn; ADDC A, Rn; JNC Out; RET, а також їх вплив на стан прапорів акумулятора.

*Приклад виконання завдання.* Визначаємо перелік необхідних ресурсів МК:

Вхід: R3, R2 - перший доданок А (в R3 - старший байт, в R2 - молодший байт), R5, R4 - другий доданок В (R5 - старший байт, R4 - молодший байт).

Вихід: R5, R4 - розміщення суми (R5 - старший байт, R4 - молодший байт). Регістр R6 - для можливого збереження n+1-го біту суми. Прапори: CY - ознака переповнення при додаванні беззнакових чисел; OV - ознака переповнення при додаванні чисел із знаком. Використані регістри: A. Вимагає вільних байт в стеку: 2 (для збереження адреси повернення з підпрограми).



Рис. 1. Блок-схема алгоритму add16 додавання двохбайтових чисел

Таблиця 2

Підпрограма add16 додавання двох 16-ти розрядних чисел		
Вхід: R3R2 – перший доданок, R5R4 – другий доданок; Вихід: R5R4 - сума; Прапори: CY - прапор переповнення при додаванні чисел ; OV – прапор переповнення при додаванні чисел зі знаком ; Використані :регістри: A. Вимагає вільних байт в стеку: 2		
Мітка	Команда	Коментарі
	MOV A, R2	;завантажити в акумулятор молодший байт першого доданку
	ADD A, R4	;додати до акумулятора молодший байт другого доданку
	MOV R4, A	;молодший байт суми помістити в регістр R4
	MOV A, R3	;завантажити в акумулятор старший байт першого доданку
	ADDC A, R5	;додати до акумулятора старший байт другого доданку
	MOV R5, A	;старший байт суми помістити в регістр R5
	JNC Out	;вийти з підпрограми, якщо прапор переносу не встановлено
	MOV R6, #01	; записати в R6 1, якщо розмірність результату більше 16
Out:	RET	; вихід на підпрограму, що викликала Add16

Блок-схема алгоритму можна зобразити так (рис. 1), де блоки 1, 2, 3 призначені для додавання молодших байтів операндів та збереження суми в регістрі

R4, блоки 4, 5, 6 виконують додавання старших байтів операндів та збереження їх суми в РЗП R5. Блоки 7 та 8 здійснюють збереження n+1-го біту результату в разі потреби.

Підпрограму add16 з врахуванням необхідних коментарів можна записати таким чином, як показано у табл. 2.

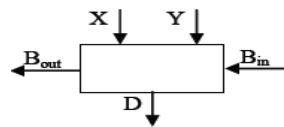


Рис. 2. Сигнали схеми віднімання

Таблиця істинності схеми двійкового віднімання

X	Y	$B_{in}$	$D = X - Y$	$B_{out}$
0	0	0	0	0
1	0	0	1	0
1	1	0	0	0
0	1	0	1	-1
0	0	-1	1	-1
1	0	-1	0	0
1	1	-1	1	-1
0	1	-1	0	-1

Повна однорозрядна схема віднімання для

двійкової системи числення (рис. 2.) функціонує у відповідності до наступної таблиці істинності, де для кожної комбінації вхідних сигналів X, Y та  $B_{in}$  вказані значення вихідних сигналів D та  $B_o^*$ . Щоб відрізнити позику від переносу перед одиницею проставлено знак (-).

Команда 8-ми бітового віднімання SUBB з врахуванням позики виконує віднімання від вмісту акумулятора X вміст вказаного операнда Y.

SUBB використовує для біту позики B нульовий розряд  $B_{in_0}$  поточне значення прапора переносу CY з регістра PSW мікроконтролера. При завершенні виконання інструкції SUBB в CY записується  $B_{out_7}$ , що є бітом позики з 7-го розряду. В свою чергу  $B_{out_7}$ , збережений в CY, може стати позикою з нульового розряду  $B_{in_0}$  при відніманні старших байтів чисел.

Таким чином, повторюючи SUBB, можна віднімати багатобайтові числа. Для цього треба виконати SUBB для кожного байта зменшуваного та від'ємника, не втративши при цьому значень CY між наступними командами SUBB. Перед відніманням молодших байтів операндів треба очистити біт CY командою CLR C, оскільки біт позики в нульовий розряд відсутній.

Для прикладу, алгоритм SUB16 віднімання двох цілих 16-розрядних чисел можна записати так:

- 1) Очистити прапор переносу CY.
- 2) Завантажити в акумулятор молодший байт зменшуваного.
- 3) Відняти від акумулятора молодший байт від'ємника.
- 4) Зберегти молодший байт різниці.
- 5) Завантажити старший байт зменшуваного в акумулятор.
- 6) Відняти від вмісту акумулятора старший байт від'ємника з врахуванням позики із молодшого байта.
- 7) Зберегти старший байт різниці.

Блок-схема алгоритму SUB16 може мати такий вигляд (рис. 3). В алгоритмі SUB16 блок 1 призначений для очищення прапора позики перед відніманням молодших байтів операндів. Блоки 2, 3, 4 виконують віднімання молодших байтів операндів та збереження різниці в регістрі R4. Блоки 5, 6, 7 виконують віднімання старших байтів операндів та збереження їх різниці в R5. Для реалізації наведеного на рис. 3 алгоритму можливо використати наступну підпрограму SUB 16 (табл. 3).



Рис. 3. Блок-схема алгоритму SUB16 віднімання двобайтових чисел

Підпрограма SUB16 віднімання двобайтових чисел

;Вхід: R3 R2 – від'ємник, R5 R4 – зменшуваче ;Вихід: R5 R4 – різниця. Прапори: CY – ознака позики(анти-переповнення) при відніманні беззнакових чисел ;OV – ознака переповнення при відніманні чисел із знаком ;Використані регістри: A; Вимагає вільних байт в стеку: 2		
Мітка	Команда	Коментарі
	CLR C	;Очистити прапор переносу для першого SUBB.
	MOV A, R4	;молодший байт зменшувачого записати в A
	SUBB A, R2	;Відняти молодший байт від'ємника.
	MOV R4, A	;Записати різницю молодших байтів на місце молодшого байта зменшувачого (R4).
	MOV A, R5	;Старший байт зменшувачого записати в A
	SUBB A, R3	;Відняти старший байт від'ємника і позику
	MOV R5, A	;Записати різницю старших байтів на місце старшого байта зменшувачого (R5).
	RET	;Повернення з підпрограми.

Операцію віднімання зручно використовувати для порівняння двох чисел. Якщо після віднімання беззнакових чисел утворюється позика, то зменшуваче менше від'ємника. Якщо позика не утворюється, то зменшуваче більше від'ємника. Рівність має місце, якщо не утворюється позика і всі байти різниці дорівнюють 0.

Для виконання роботи необхідно вивчити призначення та порядок виконання мікроконтролером команд CLR C, MOV A, Rn та SUBB A, R2

## 2. Програмування операцій додавання та віднімання операндів

*Варіант 1.* Додати два числа. Число  $A = 750_{10}$  знаходиться в R0, R1 число B – у R2, R3. Результат розташувати в резидентній пам'яті даних (РПД), починаючи з адреси 00h.

*Варіант 2.* Додати два числа. Число  $A = 273_{10}$  знаходиться в R2, R3 число B – у РПД у комірках пам'яті:  $31_{10}$ ,  $32_{10}$ . Результат розташувати в резидентній пам'яті даних (РПД), починаючи з адреси 02h.

*Варіант 3.* Додати два числа. Число  $A = 275_{10}$  знаходиться в РПД у комірках пам'яті з адресами  $33_{10}$ ,  $34_{10}$  число B – у РПД у комірках пам'яті з адресами  $35_{10}$ ,  $36_{10}$ . Результат розташувати в резидентній пам'яті даних (РПД), починаючи з адреси 04h.

*Варіант 4.* Додати два числа. Число  $A = 348_{10}$  знаходиться в РПД у комірках з адресами  $35_{10}$ ,  $36_{10}$  число B – у РПД у комірках пам'яті з адресами  $37_{10}$ ,  $38_{10}$ . Результат розташувати в резидентній пам'яті даних (РПД), починаючи з адреси 06h.

*Варіант 5.* Додати два числа. Число  $A = 128_{10}$  знаходиться в РПД у комірках пам'яті з адресами  $40_{10}$ ,  $41_{10}$ , число B – у резидентній пам'яті даних (РПД) у комірках пам'яті з адресами  $05_{10}$ ,  $06_{10}$  Результат розташувати в РПД, починаючи з адреси 07h.

*Варіант 6.* Відняти два числа. Число A знаходиться у R4, R5 число B – у R6, R7. Результат розташувати у РПД, починаючи з адреси 21h.



*Варіант 7.* Відняти два числа. Число  $A = 373_{10}$  знаходиться в R0, R1 число B – у РПД у комірках пам'яті з адресами  $42_{10}$ ,  $43_{10}$ . Результат розташувати в R4, R5.

*Варіант 8.* Відняти два числа. Число  $A = 375_{10}$  знаходиться в РПД у комірках пам'яті з адресами  $21h$ ,  $22h$ , число B – у РПД у комірках пам'яті з адресами  $40_{10}$ ,  $41_{10}$ . Результат розташувати в резидентної пам'яті даних (РПД) за адресою  $10h$ .

*Варіант 9.* Відняти два числа. Число  $A = 448_{10}$  знаходиться в РПД у комірках пам'яті з адресами  $42_{10}$ ,  $43_{10}$ , число B – у РПД у комірках пам'яті з адресами  $2Ch$ ,  $2Dh$ . Результат розташувати в резидентної пам'яті даних (РПД) за адресою  $0Ah$ .

*Варіант 10.* Відняти два числа. Число  $A = 148_{10}$  знаходиться в РПД у комірках пам'яті з адресами  $44_{10}$ ,  $45_{10}$ , число B – у РПД у комірках пам'яті з адресами  $40h$ ,  $41h$ . Результат розташувати в РПД в чарунках пам'яті з адресами  $42h$ ,  $43h$ .

*Варіант 11.* Додати два числа. Число A знаходиться в РПД у комірках пам'яті з адресами  $30h$ ,  $31h$ , число B – у зовнішньому ОЗП у комірках пам'яті з адресами  $00h$ ,  $01h$ . Результат розташувати в РПД у комірках, починаючи з адреси  $32h$ .

*Варіант 12.* Додати два числа. Число A знаходиться в РПД у комірках пам'яті з адресами  $30h$ ,  $31h$ , число B – у РПД у комірках пам'яті з адресами  $32h$ ,  $33h$ . Результат розташувати в РПД у комірках, починаючи з адреси  $34h$ .

*Варіант 13.* Відняти два числа. Число A знаходиться в РПД у комірках пам'яті з адресами  $40h$ ,  $41h$ , число B – у РПД у комірках пам'яті з адресами  $42h$ ,  $43h$ . Результат розташувати в резидентної пам'яті даних (РПД), починаючи з адреси  $00h$ .

*Варіант 14.* Відняти два числа. Число A знаходиться в резидентної пам'яті даних (РПД) у комірках пам'яті з адресами  $00h$ ,  $01h$ , число B – у РПД у комірках пам'яті з адресами  $50h$ ,  $51h$ . Результат розташувати в резидентної пам'яті даних (РПД), починаючи з адреси  $02h$ .

*Варіант 15.* Відняти два числа. Число A знаходиться в РПД у комірках пам'яті з адресами  $45h$ ,  $46h$ , число B – у R3, R4. Результат розташувати в резидентної пам'яті даних (РПД), починаючи з адреси  $10h$ .

*Приклад 1.* Додати два числа. Число  $A = 448_{10} = 1C0_{16}$  знаходиться в РПД у комірках пам'яті з адресами  $58_{10} = 3A_{16}$ ,  $59_{10} = 3B_{16}$  число  $B = 431_{10} = 1AF_{16}$  – у R3, R4. Результат розташувати в R6, R7.

0000 C3 CLR C;

0001 753AC0 MOV 3A, #C0;

0004 753D01 MOV 3B, #01;

0007 7DAF MOV R3, #AF;

0009 7C01 MOV R4, #01;

000B EB MOV A, R3;

000C 253A ADD A, 3A;

000E FD MOV R5, A;

000F EC MOV A, R4;

0010 353B ADDC A, 3B;

0012 FE MOV R6, A;

*Приклад 2.* Для МК51 складемо програму віднімання двох двобайтових чисел, якщо  $A > B$ :

ORG 00h ; початкова адреса пуску МК-51

AJMP ST ; короткий безумовний перехід на першу команду програми

ORG 30h ; адреса першої команди програми  
ST: CLR C; ; очищення біта переносу  
MOV A, R0 ; молодший байт першого числа в акумуляторі  
SUBB A, R2 ; віднімання молодшого байта другого числа  
MOV R0, A ; запам'ятовування молодшого байта різниці  
MOV A, R1 ; старший байт першого числа в акумуляторі  
SUBB A, R3 ; віднімання старшого байта другого числа і позики  
MOV R1, A ; запам'ятовування старшого байта різниці

### **3. Порядок роботи в інтегрованому середовищі EdSim51 Simulator**

Симулятор EdSim51DI TM Simulator працює під управлінням ОС Windows або Ubuntu в середовищі Java, яке можливо завантажити за посиланням <http://www.java.com> та попередньо встановити на свій пристрій.

Отримати на EdSim51DI TM Simulator можливо на сайті розробника за посиланням <https://www.edsim51.com/index.html> безкоштовно. Після того, як ви завантажили файл, розпакуйте його у своїй папці. Для того, щоб запустити в емулятор потрібно двічі клацнути на файлі з ім'ям edsim51di.jar.

Симулятор «edsim51» призначений для написання і налагодження програм для мікропроцесорів сімейства Intel mcs-51. Програмування процесора здійснюється за допомогою побудови програми на мові асемблер. Зовнішній вигляд симулятора наведено на рис. 4.

Як видно, робоче вікно симулятора складається з чотирьох частин:

1. Область пам'яті мікропроцесора, в якій відображаються значення регістрів загального призначення, регістрів спеціальних функцій. Показаний область пам'яті програм і пам'яті даних. В цій області відображається елементи управління при налагодження програми.

2. Вікно користувача. У цьому вікні відображається налагоджувальний код, який виконується в цій мікропроцесорному середовищі.

3. Вікно портів введення / виводу призначений для детального відображення стану значень портів P0-P3 в бінарному вигляді. Текстом підписані пристрої, які підключені до відповідних портів введення / виводу.

4. Панель зовнішніх пристроїв введення-виведення, підключених до мікропроцесора. Складається з наступних пристроїв (зліва - направо, зверху вниз):

- Аналого-цифровий перетворювач (АЦП), який перетворює аналогове напруга в цифровий вигляд (ADC);
- Дисплей, який складається з чотирьох семи сегментних індикаторів, підключених за схемою динамічної індикації (7 Segment LED Displays);
- Приймо-передавач послідовного порту (8-bit UART);
- Клавіатурний модуль (Keypad);
- Модуль світлодіодної індикації (LEDs);
- Імітатор обертання двигуна постійного струму (DC Motor);
- Модуль перемикачів (Switch bank);
- Цифро-аналоговий перетворювач (ЦАП), який формує аналогове напругу з цифрового коду. Вихід ЦАП підключений до імітатору осцилографа, який динамічно відображає форму і рівень сформованої напруги. (DAC output on scope).

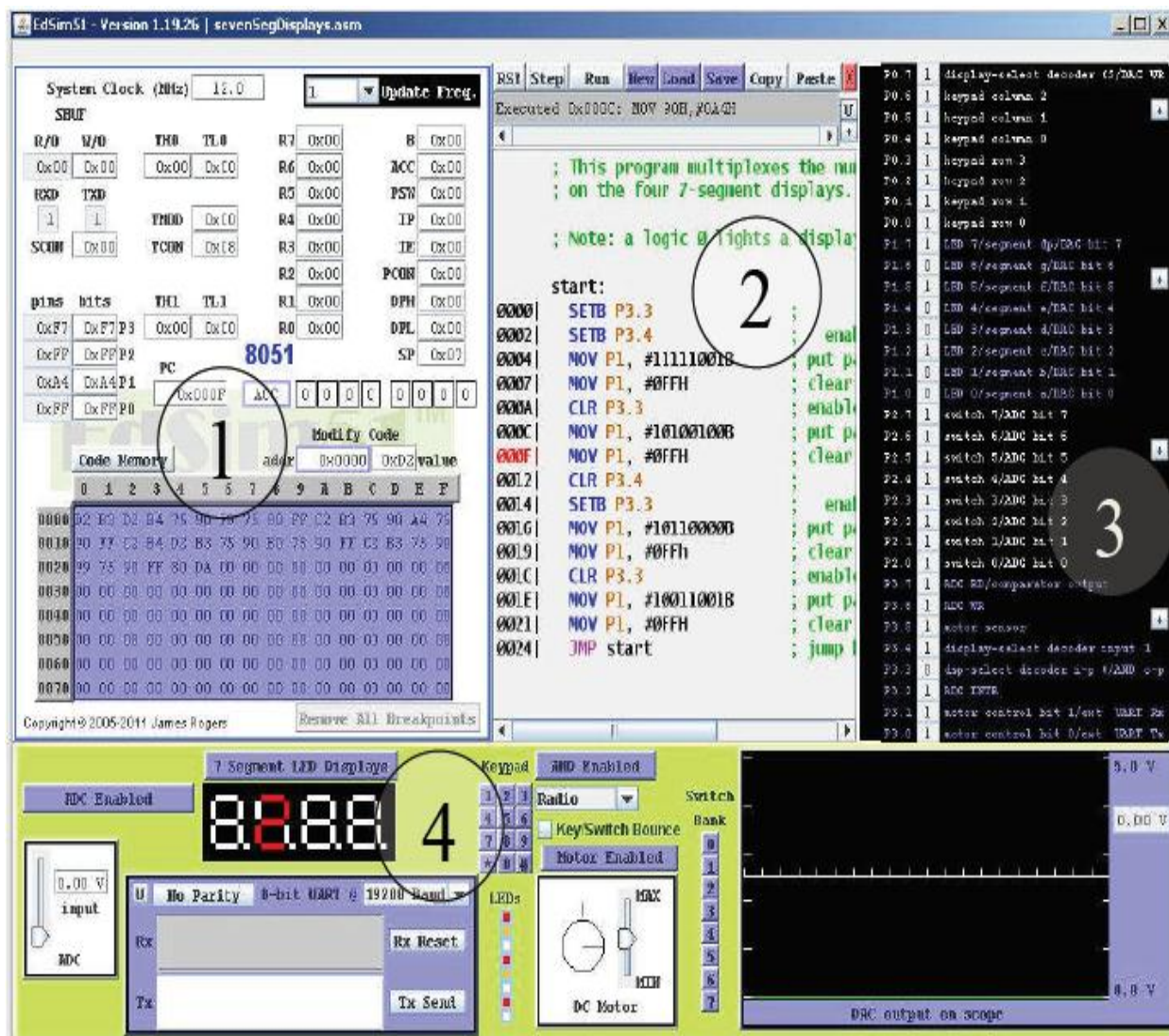


Рис. 4. Робоче вікно симулятора «edsim51»

Схема підключення цих пристроїв і модулів до мікропроцесора приведена на рис. 5. Завдяки цій схемі можна визначити як, до якого порту підключений виріб, також як організовано взаємодія з мікропроцесором.

Для того, щоб запустити середовище налагодження треба відкрити папку «edsim51» і в корені запустити подвійним натисканням файл «edsim51.jar», після цього можна побачити екранну форму, вид який представлений на рис.6.

Щоб відкрити робочий приклад програми треба скористатися кнопкою «Load», які розташовані у верхній частині вікна користувача. Після цього, вибрати приклад програми, виділивши одинарним натисканням миші і підтвердивши натисканням кнопки «Open», як показано на рис. 6.

За допомогою кнопки «RST», розташованої у верхній частині вікна користувача можна зробити початкову ініціалізацію імітованого процесора, при цьому обнуляться значення лічильника команд і регістрів. Значення покажчика стека при цьому, дорівнюватиме 0x07. Виконання програми можливо в автоматичному і покроковому режимі. Покроковий режим застосовується для



налагодження програми і активується натисканням кнопки «Assm», для продовження виконання програми по кроках необхідно натискати кнопку «Step».

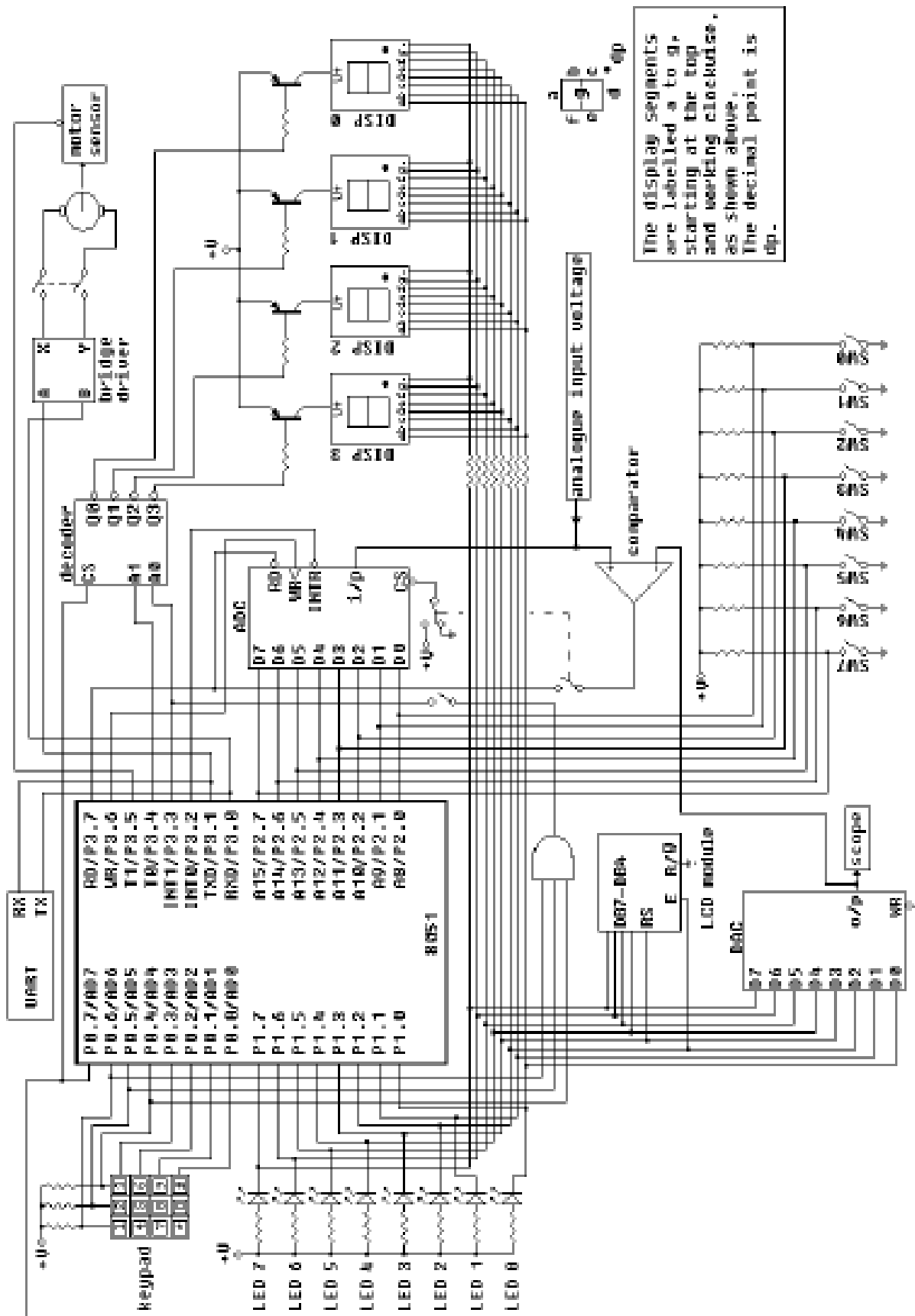


Рис. 5. Схема електрична принципова середовища навчання «edsim51»

Автоматичний режим дозволяє користувачеві запуснути програму на виконання, при цьому зупинка програми можлива в місцях, в яких встановлена точка зупинки програми (breakpoint). Точка, крапка зупинки програми може бути виставлена шляхом подвійного натискання лівої кнопки миші на адресу інструкції, як показано на рис. 7.

За допомогою кнопки «New» користувач може створити новий файл програми, а за допомогою кнопки «Save», зберегти файл програми на диск.

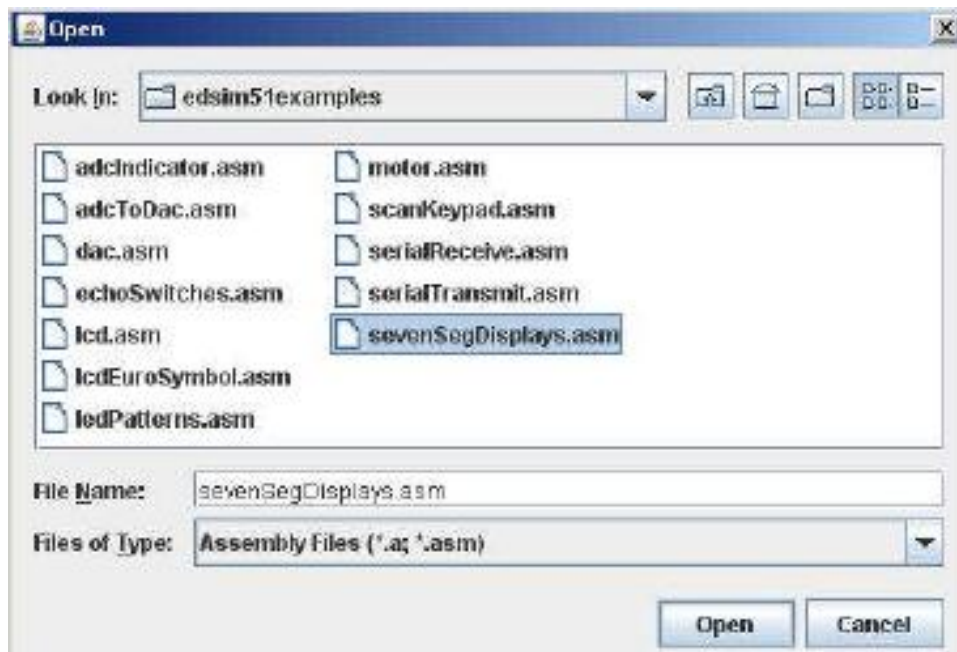


Рис 6. Зовнішній вигляд діалогового вікна відкриття коду програми

000C		MOV P1, #10100100B	; put pattern for 2 on display
000F		MOV P1, #0FFh	; clear the display
0012		CLR P3.4	;
0014		SETB P3.3	;   enable display 1
0016		MOV P1, #10110000B	; put pattern for 3 on display
0019*		MOV P1, #0FFh	; clear the display
001C		CLR P3.3	; enable display 0
001E		MOV P1, #10011001B	; put pattern for 4 on display
0021*		MOV P1, #0FFh	; clear display
0024		JMP start	; jump back to start

Рис. 7. Відображення точок зупинки програми (адреси 0019 і 0021)

#### 4. Контрольні питання

1. Вкажіть формат і порядок виконання команди ADD додавання двох чисел мікроконтролером МК-51.
2. Вкажіть формат і порядок виконання команди SUB для віднімання двох чисел мікроконтролером МК-51.
3. В яких випадках модифікується значення прапора CY PSW?
4. Чим відрізняється виконання команд ADD A, Rn та ADDC A, Rn?
5. Що треба забезпечити перед додаванням молодших байтів багатобайтових доданків?
6. В яких випадках модифікується значення прапора CY (C) PSW при виконанні операції віднімання?

7. Який стан регістра PSW треба забезпечити перед відніманням молодших байтів операндів?

## **Тема 9. Основи мікропроцесорної техніки**

### **Практичне заняття №6. Організація системи переривань та таймерів-лічильників у мікроконтролерах MCS-51**

#### **Навчальна мета заняття:**

1. Практично дослідити принципи організації переривань та можливості резидентних таймерів-лічильників у мікроконтролерах сімейства MCS-51.

2. Опанування методів використання таймерів MCS-51 для обробки часових параметрів сигналів.

3. Набуття навиків створення і моделювання функціонування мікроконтролерів.

**Кількість годин:** 4 год.

#### **Література:**

1. Круліковський Б.Б., Николайчук Я.М., Шатний С.В. Мікропроцесорні системи. Практикум. Навчальний посібник. Рівне : НУВГП, 2016. 191 с.

2. Кравець В.О., Сокол Є.І., Рисований О.М. Комп'ютерна схемотехніка. Підручник. Харків: НТУ "ХПІ", 2007. 480 с.

3. Методичні вказівки до лабораторних робіт з дисципліни «Мікропроцесорна техніка» Укл.: В.І. Рева. Запоріжжя: ЗНТУ, 2019. 114 с.

4. Плахтєєв А.П., Бабешко Є.В., Ткаченко В.А., Здоровець Ю.В. Архітектури та розроблення систем Інтернету / Вебу Речей на основі вбудованих платформ. Лабораторні роботи / За ред. В.С. Харченка. Міністерство освіти і науки України, Національний аерокосмічний університет, ХАІ, 2019. 147 с.

5. Михайлов С.Р. Основи мікропроцесорної техніки: Лабораторний практикум [Електронний ресурс] : навч. посіб. для студ. /; КПІ ім. Ігоря Сікорського. Київ : КПІ ім. Ігоря Сікорського, 2019. 59 с.

6. Мікроконтролери сімейства MCS-51 в задачах обробки інформації та керування. Методичні вказівки до виконання лабораторних робіт з дисципліни «Мікропроцесорні пристрої керування та обробки інформації» для студентів. Укл.: Войтенко В.П., Хоменко М.А. Чернігів: ЧНТУ, 2014. 71 с.

**Матеріально-технічне забезпечення:** комп'ютерна мережа з інтегрованим середовищем EdSim51 Simulator.

#### **Навчальні питання**

1. Вивчення алгоритмів додавання та віднімання багатобайтових чисел.....	43
2. Програмування операцій додавання та віднімання операндів.....	43
3. Дослідження функціонування модуля постійної пам'яті.....	47
3. КОНТРОЛЬНІ ПИТАННЯ .....	50

### **1. Організація системи переривань в MCS-51**

*Переривання* – це спосіб асинхронного обміну між ядром МП та периферійним пристроєм, призначення якого – скорочення простоювання (очікування, холостого ходу) МП. Якщо певне переривання дозволене (розблоковано), і сталася подія, що асоційована з цим перериванням, основна виконуєма програма тимчасово призупиняється, а поточна адреса лічильника команд зберігається у стеку. Далі

розпочинається підпрограма обробки переривання, яка обов'язково повинна завершуватися командою RETI, що забезпечує повернення адреси перерваної раніше основної програми з стеку. Таким чином, механізм переривань можна порівняти з асинхронним викликом підпрограми у момент, який визначає джерело запиту. Адреса першої команди обробки переривання (вектор переривання), визначається конкретним джерелом запиту (рис. 1).

Зовнішні переривання INT0 та INT1 в MCS-51 можуть бути викликані рівнем або переходом сигналу з 1 в 0 на входах МК в залежності від значень керуючих бітів IT0 та IT1 в регістрі спеціальних функцій TCON. Від зовнішніх переривань встановлюються прапори IE0 та IE1 в регістрі TCON, що ініціює виклик відповідної підпрограми обслуговування переривання. Скидання цих прапорів виконується апаратно тільки в тому випадку, коли переривання було викликане по переходу (зрізу) сигналу. Якщо ж переривання викликане рівнем входного сигналу, то скиданням прапору IE керує відповідна підпрограма обслуговування переривання шляхом впливу на джерело переривання з метою зняття їм запиту.

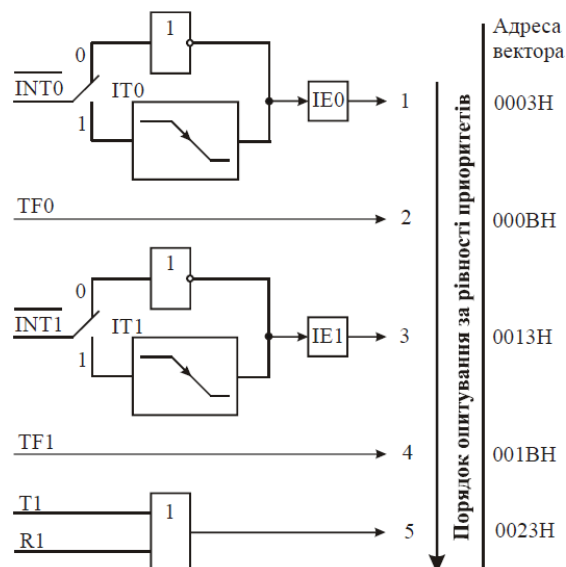


Рис. 1. Схема переривань в MCS-51

Прапори запитів переривань від таймерів TF0 і TF1 скидаються автоматично при передачі керування підпрограмі обслуговування. Прапори запитів переривань від послідовного порту RI та TI встановлюються UART апаратно, проте скидатися повинні програмою. Переривання можуть бути викликані або відмінені програмою, оскільки усі перераховані прапори програмно доступні.

У блоці регістрів спеціальних функцій є два регістри, призначені для керування режимом переривань і рівнями пріоритету. Формати цих регістрів, що мають символічні імена IE и IP, описано в таблиці 1 і 2.

Таблиця 1. Регістр масок переривань IE

Символ	Розряд	Ім'я та призначення
EA	IE.7	Зняття блокування переривань. Скидається програмно для заборони усіх переривань незалежно від стану IE4-IE0
-	IE.6, 5	Не використовується
ES	IE.4	Біт дозволу переривання від UART. Установка/скидання програмою для дозволу/заборони переривань від TI, RI
ET1	IE.3	Біт дозволу переривання від таймера 1. Установка/скидання програмою для дозволу/заборони переривань від таймера 1
EX1	IE.2	Дозвіл зовнішнього переривання 1. Установка/скидання програмою для дозволу/заборони переривання від INT1
ET0	IE.1	Дозвіл переривання від таймера 0. Працює аналогічно IE.3
EX0	IE.0	Дозвіл зовнішнього переривання 0. Працює аналогічно IE.2

Таблиця 2. Регістр пріоритетів переривань IP

Символ	Розряд	Ім'я та призначення
-	IP.7-5	Не використовується
PS	IP.4	Пріоритет UART. Установка/скидання програмою для призначення перериванню від UART вищого/нижчого пріоритету
PT1	IP.3	Пріоритет таймера 1. Установка/скидання програмою для призначення перериванню від таймера 1 вищого/нижчого пріоритету
PX1	IP.2	Пріоритет зовнішнього переривання 1. Установка/скидання програмою для призначення перериванню INT1 вищого/нижчого пріоритету
PT0	IP.1	Пріоритет таймера 0. Працює аналогічно IP.3
PX0	IP.0	Пріоритет зовнішнього переривання 0. Працює аналогічно IP.2

Можливість програмної установки/скидання будь-якого керуючого біту в цих двох регістрах робить систему переривань виключно гнучкою.

Прапори переривань опитуються в кожному машинному циклі. Ранжирування переривань за пріоритетами виконується протягом наступного машинного циклу. Система переривань формує апаратно виклик LCALL відповідної підпрограми обслуговування, якщо вона не заблокована однією з умов: в даний момент обслуговується запит переривання рівного або більш високого рівня пріоритету; поточний машинний цикл – не останній в циклі команди, що виконується; виконується команда RETI або будь-яка команда, пов'язана зі звертанням до регістрів IE або IP.

Якщо прапор переривання був встановлений, але за жодною з перерахованих умов не отримав обслуговування та до моменту закінчення блокування вже був скинутий, запит переривання губиться.

За апаратно сформованим кодом команди LCALL система переривань розміщує у стеку вміст програмного лічильника PC і завантажує в PC адресу вектора переривання відповідної підпрограми обслуговування. За цією адресою повинна бути розміщена команда безумовного переходу JMP до початкової адреси підпрограми обслуговування переривання. Ця підпрограма у разі необхідності повинна починатися командами запису в стек PUSH слова стану програми PSW, акумулятора A, розширювача акумулятора B, покажчика даних DPTR тощо та закінчуватися командами повернення з стеку POP. Підпрограми обслуговування переривання обов'язково завершуються командою RETI, за якою в програмний лічильник перезавантажується з стеку збережена адреса повернення в основну програму. Команда RET також повертає керування, але при цьому не знімає блокування переривання.

## 2. Організація таймерів/лічильників в MCS-51

В складі МК MCS-51 є регістрові пари з символічними іменами TH0, TL0 и TH1, TL1, на основі яких функціонують два незалежних програмно-керуємих 16-бітних таймера/лічильника подій (T/C0 и T/C1). Під час роботи в якості таймера вміст T/C інкрементується в кожному машинному циклі, тобто через кожні 12 періодів резонатора. При роботі в якості лічильника вміст T/C інкрементується під впливом переходу з 1 в 0 зовнішнього входного сигналу, який подається на відповідний (T0, T1) вхід МК. Опитування сигналів виконується в кожному машинному циклі. Оскільки на розпізнання переходу потрібно два машинних цикли,

максимальна частота підрахунку вхідних сигналів дорівнює  $1/24$  частоти резонатора. На тривалість періоду вхідних сигналів обмежень згори нема. Для гарантованого прочитання вхідного зчитуваного сигналу він повинен утримувати значення 1 як мінімум протягом одного машинного циклу.

Для керування режимами роботи та для організації взаємодії таймерів із системою переривань використовуються два регістри спеціальних функцій TMOD і TCON, опис яких наведений в таблицях 3, 4 та 5. Для обох T/C режими роботи 0, 1 і 2 однакові. Режими 3 для T/C0 і T/C1 відмінні.

*Режим 0.* Переведення будь-якого T/C в цей режим робить його 8-розрядним таймером, до входу якого підключений 5-бітний переддільник частоти на 32. В цьому режимі таймерний регістр має розрядність 13 біт. При переході із стану “всі одиниці” в стан “всі нулі” встановлюється прапор переривання від таймера TF1. Вхідний синхросигнал таймера 1 дозволений (поступає на вхід T/C), коли керуючий біт TR1 встановлений в 1 і або керуючий біт GATE (блокування) дорівнює 0, або на зовнішній вхід запиту переривання INT1 поступає рівень 1.

Таблиця 3. Регістр режиму роботи таймера/лічильника TMOD

Символ	Розряд	Ім'я та призначення
GATE	TMOD.7 для T/C1 TMOD.3 для T/C0	Керування блокуванням. Якщо біт встановлений, таймер/лічильник “х” дозволений доти, доки на вході “INT х” високий рівень і біт керування “TRх” встановлений. Якщо біт скинутий, T/C дозволений одразу після встановлення біту керування “TRх”
C/T	TMOD.6 для T/C1 TMOD.2 для T/C0	Біт вибору режиму таймера або лічильника подій. Якщо біт скинутий, працює таймер від внутрішнього джерела сигналів синхронізації. Якщо біт встановлений, працює лічильник зовнішніх сигналів на вході “Tx”
M1	TMOD.5 для T/C1 TMOD.1 для T/C0	Режим роботи (див. таблицю 4.4)
M0	TMOD.4 для T/C1 TMOD.0 для T/C0	

Таблиця 4. Режими роботи таймерів/лічильників

M1	M0	Режим роботи
0	0	“TLx” працює як 5-бітний переддільник
0	1	16-бітний таймер/лічильник. “THx” і “TLx” включені послідовно
1	0	8-бітний автоперезавантажуваний таймер/лічильник. “THx” зберігає значення, що має бути завантажено в “TLx” кожного разу за переповненням
1	1	Таймер/лічильник 1 зупиняється. Таймер/лічильник 0: TL0 працює як 8-бітний таймер/лічильник, а його режим визначають керуючі біти таймера 0. TH0 працює тільки як 8-бітний таймер, а його режим визначають керуючі біти таймера 1



Таблиця 5. Регістр керування/статусу таймера TCON

Символ	Розряд	Ім'я та призначення
TF1	TCON.7	Прапор переповнення таймера 1. Встановлюється апаратно при переповненні таймера/лічильника. Скидається при обслуговуванні переривання апаратно
TR1	TCON.6	Біт керування таймера 1. Встановлюється/скидається програмою для пуску/зупинки таймера/лічильника
TF0	TCON.5	Прапор переповнення таймера 0. Встановлюється апаратно. Скидається при обслуговуванні переривання
TR0	TCON.4	Біт керування таймера 0. Встановлюється/скидається програмою для пуску/зупинки таймера/лічильника
IE1	TCON.3	Прапор фронту переривання 1. Встановлюється апаратно, коли детектується зріз зовнішнього сигналу INT1. Скидається при обслуговуванні переривання
IT1	TCON.2	Біт керування типом переривання 1. Встановлюється/скидається програмно для специфікації запиту INT1 (зріз/низький рівень)
IE0	TCON.1	Прапор фронту переривання 0. Встановлюється за зрізом сигналу INT0. Скидається при обслуговуванні переривання
IT0	TCON.0	Біт керування типом переривання 0. Встановлюється/скидається програмно для специфікації запиту INT0 (зріз/низький рівень)

Встановлення біта GATE в 1 дозволяє використовувати таймер для вимірювання тривалості імпульсного сигналу, що подається на вхід запиту переривання.

*Режим 1.* Робота будь-якого Т/С в цьому режимі така ж, як і в режимі 0, за виключенням того, що таймерний регістр має розрядність 16 біт.

*Режим 2.* В цьому режимі роботу організовано таким чином, що переповнення (перехід із стану “всі одиниці” в стан “всі нулі”) 8-бітного лічильника TL1 призводить не тільки до встановлення прапора TF1, але й автоматично перезавантажує в TL1 вміст старшого байту (TH1) таймерного регістра, який попередньо був заданий програмним шляхом. Перезавантаження залишає вміст TH1 незмінним. В режимі 2 Т/С0 і Т/С1 працюють абсолютно однаково.

*Режим 3.* В цьому режимі Т/С0 і Т/С1 працюють по-різному. Т/С1 зберігає незмінним свій поточний вміст. Іншими словами, ефект такий же, як і при скиданні керуючого біта TR1 в нуль. В цьому режимі TL0 і TH0 функціонують як два незалежних 8-бітних лічильника. Роботу TL0 визначають керуючі біти Т/С0 (С/Т, GATE, TR0), вхідний сигнал INT0 та прапор переповнення TF0. Роботу TH0, який може виконувати тільки функції таймера (підрахунок машинних циклів МК), визначають біт керування TR1. При цьому TH0 використовує прапор переповнення TF1.

### 3. Формування часової затримки таймером

Недолік програмного способу реалізації часової затримки – це нераціональне використання ресурсів МК: під час формування затримки МП-ядро практично простоює та не може вирішувати ніяких задач формування закону керування об'єктом. В той же час резидентні апаратні засоби дозволяють реалізувати часові затримки на фоні основної програми роботи.

На вхід таймера/лічильника (Т/С) можуть поступати сигнали синхронізації з частотою  $f_Q/12$  (Т/С в режимі таймера) або сигнали від зовнішнього джерела (Т/С в



режимі лічильника). Обидва ці режими можна використати для формування затримок. Якщо використати Т/С в режимі таймера повного формату (16 біт), можна отримати затримки в діапазоні  $(1 \dots 65536) T_T$ .

Будь яка програма для МК керує не тільки програмними об'єктами (даними), а й апаратними ресурсами МК (порти введення-виведення, переривання, таймери-лічильники, послідовний інтерфейс тощо). Перш, ніж використовувати будь-які апаратні ресурси, їх необхідно налаштувати на відповідний режим роботи. Первинна процедура налаштування апаратних засобів називається ініціалізацією. Ініціалізація виконується шляхом запису відповідних кодів у регістри спеціальних функцій МК.

Переважає більшість мікроконтролерів працює з машинними циклами, що складаються з 12 циклів тактової частоти (рис. 2). У цьому випадку таймер буде інкрементуватись через кожні 12 циклів тактової частоти. Наприклад, якщо тактова частота, з якою працює мікроконтролер, дорівнює 11,059 МГц (стандартна частота для більшості мікроконтролерів), то таймер буде працювати на частоті, рівній  $11,059 \text{ МГц} / 12 = 921\,583 \text{ Гц}$ .

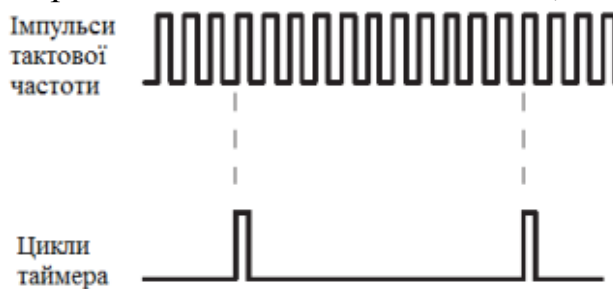


Рис. 2. Режим синхронізації

Оскільки таймер рахує з інкрементом, то настане момент, коли обидва 8-розрядних регістра міститимуть шістнадцяткові значення FF або 65535. У наступному машинному циклі значення регістрів ТНх і ТЛх будуть обнулені, після чого рахунок почнеться з 0. У цьому випадку говорять про переповнення таймера. Ситуація переповнення дуже часто використовується у високошвидкісних ресурсообмежених програмах, тим більше, що момент переповнення фіксується шляхом установки певних бітів (вони мають позначення ТFх,  $x = 0, 1$ ), названих бітами переповнення, в регістрі управління і контролю таймерів ТСОН.

У системах автоматичного керування часто приходиться вимірювати такі величини як: частота  $f$  період  $T$ , тривалість  $\tau$ , зсув фаз  $\varphi$ . Для цього застосовується перетворення частота-код. В залежності від того, який саме параметр потрібно виміряти застосовують різні підходи.

При вимірюванні частоти  $f_x$  виконується підрахунок кількості імпульсів вхідного сигналу протягом фіксованого відрізка часу  $T_0$  (рис. 3). Тоді частоту знаходять як

$$f_x = N_x / T_0.$$

Такий метод прийнятний для вимірювання частот більше 100 Гц. Верхня межа обмежується швидкодією елементів схеми і розрядністю лічильників використаного мікроконтролера.

Вимірювання частот нижче 100 Гц заміняється вимірюванням періоду  $T_x$ . При цьому виконується підрахунок

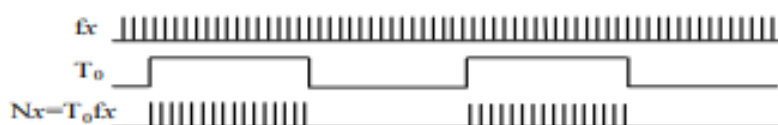


Рис. 3. Часова діаграма вимірювання частоти

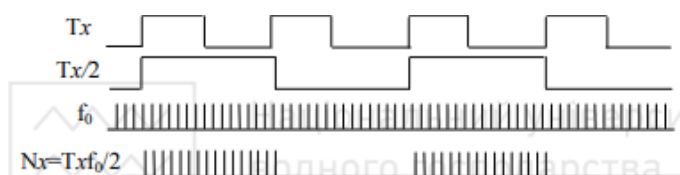


Рис. 4. Вимірювання періоду слідування методом підрахунку кількості імпульсів відомої частоти

кількості імпульсів фіксованої частоти  $f_0$  за інтервал часу, що рівний або кратний вимірюваному періоду (рис. 4).

У такий же спосіб вимірюється тривалість  $t$  імпульсів. Спосіб полягає в тому, що підраховується кількість  $N$  імпульсів відомої частоти  $f_0$  за час від початку (фронту) імпульсу до його завершення (спаду). Тоді тривалість вимірюваного імпульсу  $\tau = N/f_0$ .

Для вимірювання зсуву фаз між двома імпульсними сигналами  $f_1$  та  $f_2$  (часові діаграми 1 та 2) другий сигнал інвертується як показано на рис.5. Підрахунок кількості  $N$  імпульсів заданої частоти  $f_0$  (часова діаграма 5) запускається фронтом першого імпульса  $f_1$  і закінчується спадом проінвертованого другого імпульсу  $f_2$  з (часова діаграма 4). Тоді вимірюваний зсув фаз  $\varphi$  визначається з відомої формули  $\varphi = N/f_0$  (часова діаграма 6).



Рис. 5. Часова діаграма вимірювання зсуву фаз

Таблиця 6

Підпрограма GEN IMP - 2 генератора імпульсів скважністю 2		
Вхід:		
Вихід: світлова сигналізація на лінійці світлодіодів.		
Прапори: не використовуються		
Використані регістри: IE, IP, TCON, TMOD, A, DPTR		
Вимагає вільних байт в стеку: 2		
Мітка	Команда	Коментарі
		декларативний блок
	IE data 0A8h	регістр дозволу переривань
продовження табл. 2.26		
	IP data 0B8h	регістр пріоритетів переривань
	TCON data 088h	регістр контролю переривань
	TMOD data 089h	регістр налаштування таймерів
	TR1 bit TCON.6	власне ім'я біта TR1 регістра TCON
	TF1 bit TCON.7	власне ім'я біта TF1 регістра TCON
	TL1 data 8Bh	адреса молодшого байта T/C1
	TH1 data 8Dh	адреса старшого байта T/C1
	ORG 0	стартова адреса підпрограми
	Jmp start	
start:	mov IE, #00000000b	заборона усіх переривань
	mov IP, #000001000b	найвищий пріоритет для переривань від TC1
	mov TMOD, #000010000b	налаштування T/C1 в режим 16-бітного таймера
	mov R1, #0	
	mov R0, #0	R0 лічильник переповнень
loop:	setb TR1	запуск таймера
loop1:	inc A	
	mov DPTR, #0A001h	
	movx @DPTR, A	
	lb TF1, mit1	перехід на mit1 при переповненні T/C1
	jmp loop1	
mit1:	clr TF1	очистка біта переповнення

	clr TR1	зупинка роботи T/C1
	inc R0	
	cjne R0,#010, loop	перехід на loop якщо к-сть переповнень 10
	jmp startloop	
startloop:	mov R0,#00b	очищення регістра R0
	cjne R1,#01, mit2	
	mov A, #0255D	підпрограма формування на ЦАП
	mov DPTR, #0F000h	напруги високого рівня і засвічення
	movx @DPTR, A	діодів
	mov DPTR, #0A006h	
	movx @DPTR, A	
	mov R1, #0b	запис у R1 нуля
	jmp loop	
mit2:	mov A, #0	
	mov DPTR, #0F000h	
	movx @DPTR, Ah	
	mov DPTR, #0A006h	
	movx @DPTR, A	
	mov R1, #01	
	jmp loop	
	END	

#### Варіанти індивідуальних завдань

№	Зміст завдання
1	Здійснити широтно-імпульсну модуляцію за допомогою T/C1 та T/C0 з можливістю змінювати скважність імпульсів за допомогою SW15 (збільшення) та SW16 (зменшення).
2	Здійснити динамічне підсвічення статичного індикатора з наступними вимогами: тривалість підсвічення триває на протязі 2 секунд, а у погашеному стані індикатор знаходиться 1 секунду.
3	Реалізувати біжучий вогник на світлодіодах із затримкою 1,8 сек.
4	Здійснити перевірку лабораторного генератора імпульсів.
5	Здійснити поступове засвічення динамічного індикатора у наступній послідовності: спочатку відбувається поступове засвічення одного розряду з нагонуванням в один сегмент, коли у розряді підсвічені усі вісім сегментів, то відбувається засвічування наступного розряду і т.д.
6	Реалізувати біжучий вогник на матриці 5×7 із можливістю зміни часової затримки між ітераціями. Використати T/C1.

Для демонстрації механізму використання таймерів MCS-51 розглянемо підпрограму встановлення режимів роботи таймерів і керування. Ядро мікроконтролера містить два регістри - TMOD і TCON. В окремих бітах регістра TMOD задаються режими роботи таймерів, а старші 4 біти регістра TCON керують роботою таймерів 0 і 1. формування прямокутних імпульсів (табл. 6), тобто імпульсного генератора з періодом  $T=2$  с та скважністю  $S=2$ .

#### 4. Контрольні питання

1. Основні технічні характеристики МК i8051.
2. Призначення регістрів A та B.
3. Призначення регістрів PC та DPTR.
4. Особливості будови та використання портів P0, P1, P2 та P3.
5. Банки регістрів загального призначення: будова та застосування.
6. Пояснити роботу команд MOV, MOVC та MOVX.
7. Назвати та пояснити роботу арифметичних команд.

8. Назвати та пояснити роботу логічних команд.
9. Назвати та пояснити роботу команд безумовних та умовних переходів.

## **Тема 10. Цифро-аналогові та аналого-цифрові перетворювачі**

### **Практичне заняття №7. Дослідження цифро-аналогових та аналого-цифрових перетворювачів**

#### **Навчальна мета заняття:**

1. Вивчення принципу дії цифро-аналогового перетворювача з ваговими двійково-зваженими опорам.
2. Практичне дослідження цифро-аналогового перетворення з використанням резистивної матриці.
3. Практичне дослідження принципу дії цифро-аналогових перетворювачів, побудованих на основі матриці з двома номіналами опорів R-2R.
4. Практичне дослідження роботи цифро-аналогового, аналого-цифрового перетворювачів, що представлені в бібліотеці програми Electronics Workbench.

**Кількість годин:** 4 год.

#### **Література:**

1. Онанченко Є.Л., Бражник І.Є. Методичні вказівки до виконання лабораторних робіт «Дослідження роботи ЦАП і АЦП» з дисципліни "Електронні системи". Суми : Сумський державний університет, 2012. 30 с.
2. Кравець В.О., Сокол Є.І., Рисований О.М. Комп'ютерна схемотехніка. Підручник. Харків: НТУ "ХПІ", 2007. 480 с.
3. Рева В.І. Методичні вказівки до лабораторних робіт з дисципліни «Мікропроцесорна техніка». Запоріжжя: ЗНТУ, 2019. 114 с.

**Матеріально-технічне забезпечення:** комп'ютерна мережа із системою модулювання електронних схем Electronics Workbench.

#### **Навчальні питання**

1. Основні характеристики ЦАП і АЦП ..... 51
2. Дослідження цифро-аналогового перетворювача із двійково-зваженими опорами ..... 53
3. Дослідження ЦАП на основі матриці R-2R..... 54
4. Дослідження роботи мікросхем АЦП..... 55
5. Контрольні питання ..... 56

### **1. Основні характеристики ЦАП і АЦП**

Характеристики ЦАП і АЦП визначаються видом характеристики перетворення, що встановлює відповідність між значеннями аналогової величини і цифрового коду, і відповідають за точність перетворення.

До *статичних параметрів* відносять:

- *розрядність  $n$*  – число розрядів цифрового коду, який формується на виході АЦП або подається на вхід ЦАП;
- *максимальна кількість кодових комбінацій* (рівнів квантування) на виході АЦП або вході ЦАП, що визначається числом розрядів цифрового коду і дорівнює:  $2^n$  – для двійкових ЦАП (АЦП);  $3^n$  – для трійкових ЦАП (АЦП);
- *максимальна напруга  $U_{max}$*  (діапазон зміни вихідної напруги) – це максимальна вхідна напруга для АЦП і вихідна для ЦАП;

– *роздільна здатність* (абсолютна роздільна здатність) – найменше змінне значення вхідної величини, що розрізняється пристроєм і фіксується на виході; роздільна здатність для ЦАП – це мінімальне значення зміни вихідного сигналу, обумовлене зміною вхідного коду на одиницю у молодшому розряді; для АЦП – це мінімальна зміна величини аналогового сигналу, що викликає збільшення або зменшення вихідного коду на одиницю у молодшому розряді;

– *нелінійність*  $\delta_L$  – це максимальне відхилення точки реальної характеристики перетворення від ідеальної;

– *диференціальна нелінійність* – це відхилення дійсного кроку квантування від його середнього значення;

– *абсолютна похибка* перетворення в кінцевій точці шкали – відхилення реальних максимальних значень вхідного для АЦП і вихідного для ЦАП аналогових сигналів від значень, що відповідають кінцевій точці ідеальної характеристики перетворення;

– *напруга зсуву нуля*  $U_0$ . Для АЦП це напруга, яку необхідно прикласти до його входу для отримання нульового вихідного коду. Для ЦАП – це напруга, наявна на його виході при подачі на вхід нульового коду.

*Динамічні властивості* ЦАП і АЦП характеризуються такими параметрами:

– *максимальна частота перетворення* – найбільша частота дискретизації, при якій задані параметри відповідають встановленим нормам;

– *час перетворення* – це інтервал часу від подачі цифрового коду на вхід ЦАП до появи вихідної напруги або інтервал часу від моменту зміни аналогового сигналу на виході АЦП до появи на його виході відповідного стійкого коду.

*Процес аналого-цифрового перетворення* показаний на рисунку 1 і полягає у послідовному виконанні таких дій:

– *процедура вибірки* – вибірка значень вхідної аналогової величини у деякий заданий момент часу, тобто відбувається дискретизація сигналу у часі (рис. 1а). Один із показників якості трансформації безперервного аналогового сигналу в цифровий сигнал – це *частота дискретизації*;

– *процес квантування* – округлення до деяких відомих величин (рівнів квантування) отриманих у дискретні моменти часу значень аналогової величини. У даному випадку на якість аналого-цифрового перетворення впливає кількість рівнів квантування, використовуваних для заміни безперервного аналогового сигналу на цифровий сигнал (рис. 1б);

– *кодування* – заміна знайдених окремих у часі значень вхідного сигналу на числові коди (рис. 1г).

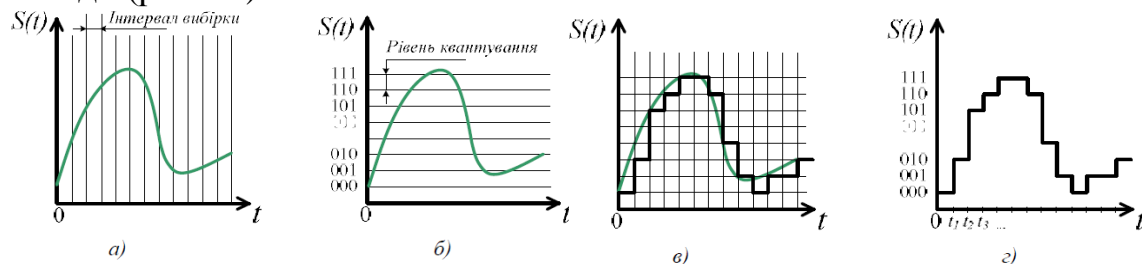


Рис.1. Перетворення аналогового сигналу у цифровий: а) процедура вибірки; б) процес квантування; в) вибірки і квантування; г) цифровий сигнал



Одна з основних проблем, з якою стикаються при перетворенні безперервного сигналу в дискретний, це проблема вибірки інтервалу дискретизації (частоти дискретизації). Відповідно до теорема Котельникова (теорема відліків, теорема Найквіста - Шенона), якщо аналоговий сигнал має обмежений по ширині спектр, то він може бути відновлений однозначно і без втрат за своїми дискретними відліками, узятими із частотою строго більше подвоєної верхньої (максимальної) частоти спектра.

## 2. Дослідження цифро-аналогового перетворювача із двійково-зваженими опорами

Таблиця 1

№	Опорна напруга $U_{оп}, В$	Вхідний код $a_3a_2a_1a_0$
1	7	
2	12	
3	13	
4	14	
5	16	0000,
6	8	0001,
7	9	0010,
8	10	0011,
9	11	0100,
10	12	0101,
11	14	0110,
12	9	0111,
13	18	1000,
14	16	1001,
15	8	1010,
16	11	1011,
17	10	1100,
18	12	1101,
19	13	1110,
20	14	1111
21	15	
22	16	
23	17	
24	12	
25	14	

1. Зібрати схему ЦАП із двійково-зваженими опорами (рис. 2). Схема складається з двох блоків: резистивної матриці ( $R_1-R_4$ ) і підсумувального підсилювача. Останній реалізований на операційному підсилювачі (ОП) з

Таблиця 2

Вхідний код $a_3a_2a_1a_0(R_1-R_4)$	Вихідна напруга	
	$U_{вих1}$	$U_{вих2}$
0000		
0001		
...		
1111		

резистором  $R_{33}$  у зворотному зв'язку.  $U_{оп}$  – опорна напруга. Значення кожного розряду (вага) задається опором, зворотним зв'язку  $R_{33}$  і відповідними резисторами

$R_1-R_4$ .

Якщо всі перемикачі замкнуті на «землю», то напруга на виході ОП буде дорівнювати 0. При замиканні, наприклад, ключа  $K_4$  в положення, що відповідає логічній одиниці, на вхід ОП через резистор  $R_4$  подається напруга  $U_{оп}$ . Тоді коефіцієнт посилення вхідної напруги дорівнює  $k=R_{33}/R_4$ , а на виході маємо напругу  $(U_{оп} R_{33})/R_4$ .

Отже, при подачі на вхід ЦАП цифрового коду  $a_3a_2a_1a_0$  вихідну напругу можна розрахувати за формулою:

$$U_{вих} = a_3 \cdot \underbrace{\frac{U_{оп} \cdot R_{33}}{R_1}}_{\text{вага старшого розряду}} + a_2 \cdot \frac{U_{оп} \cdot R_{33}}{R_2} + a_1 \cdot \frac{U_{оп} \cdot R_{33}}{R_3} + a_0 \cdot \underbrace{\frac{U_{оп} \cdot R_{33}}{R_4}}_{\text{вага молодшого розряду}},$$

2. Встановити напругу на джерелі  $U_{оп}$  згідно зі своїм варіантом відповідно з даними табл. 1.

3. Задати відповідні номінали опорів  $R_1-R_4$ :  $R_1=R$ ,  $R_2=2R$ ,  $R_3=4R$ ,  $R_4=8R$ . Опір зворотного зв'язку дорівнює  $R_{33}=R$  або  $R_{33}=R/2$ . Значення  $R$  може вибиратися будь-яке у розумних межах.

4. Комутовуючи ключі  $K_1$ ,  $K_2$ ,  $K_3$  і  $K_4$ , тим самим набираючи послідовно вхідний код із табл. 1, виконати цифро-аналогове перетворення у покроковому режимі. Для

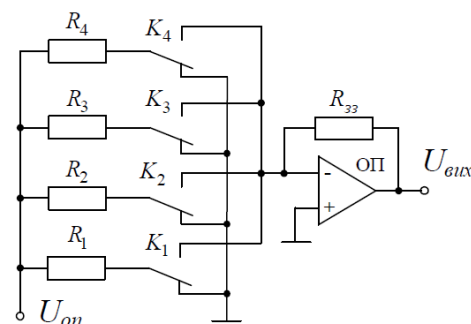


Рис. 2. ЦАП із двійково-зваженими опорами

кожного випадку зняти двома показання мультиметра(вольтметра) і заповнити колонки  $U_{вих1}$  і  $U_{вих2}$  у табл. 2.

На рис. 3 показаний момент подачі на ЦАП цифрового сигналу 1011 ( $K_1$  - ключ старшого розряду). У даному випадку розрахункова вихідна напруга дорівнює

$$U_{вих} = 1 \cdot \frac{U_{on} \cdot R_{33}}{R_1} + 0 \cdot \frac{U_{on} \cdot R_{33}}{R_2} + 1 \cdot \frac{U_{on} \cdot R_{33}}{R_3} + 1 \cdot \frac{U_{on} \cdot R_{33}}{R_4}, \quad U_{вих} = 1 \cdot 12 + 0 \cdot 6 + 1 \cdot 3 + 1 \cdot 1,5 = 12 + 3 + 1,5 = 16,5 \text{ В}$$

і, як бачимо, збігається з даними, які відображаються на мультиметрі (рис. 3).

5. Побудувати графіки залежності вихідної напруги від вхідного коду (характеристику перетворення) для обох випадків.

6. Визначити основні параметри ЦАП. Зробити висновки.

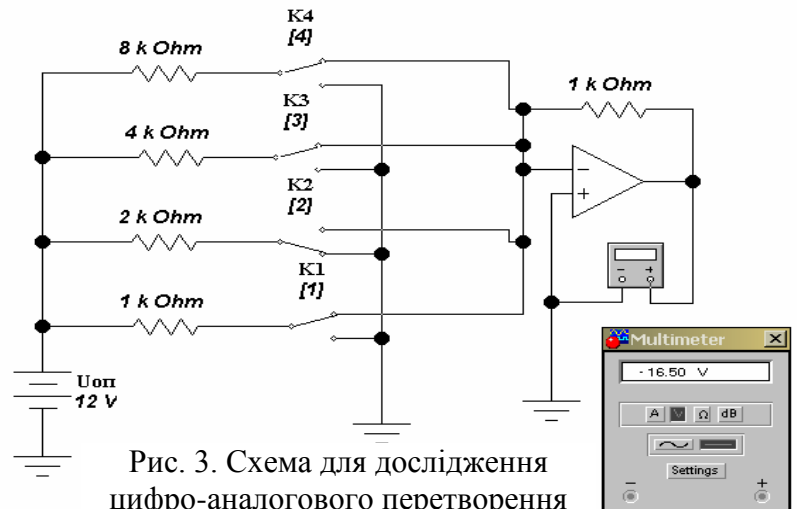


Рис. 3. Схема для дослідження цифро-аналогового перетворення

### 3. Дослідження ЦАП на основі матриці R-2R

1. Зібрати схему ЦАП на основі матриці R-2R з підсумовуванням напруги або струму з завдання викладача (рис. 4 або рис. 5). Перевага даних перетворювачів полягає в тому, що резистивна матриця містить тільки два номінали опорів на відміну від ЦАП із двійково-зваженими опорами, що має широкий діапазон номіналів.

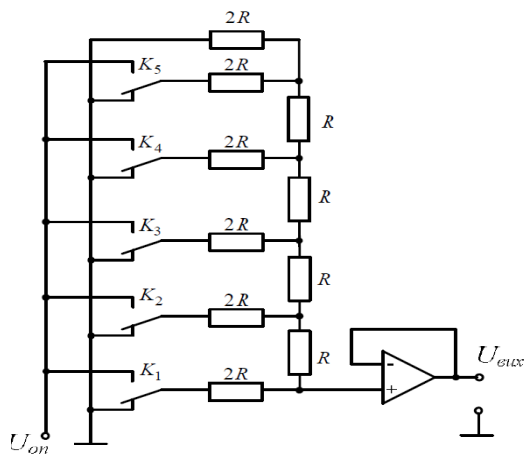


Рис. 4. ЦАП із матрицею R-2R з підсумовуванням напруги

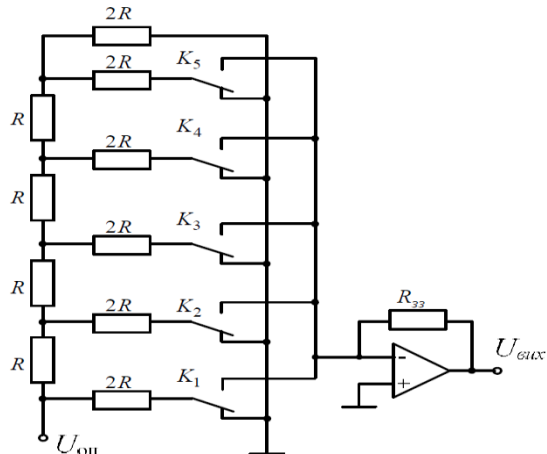


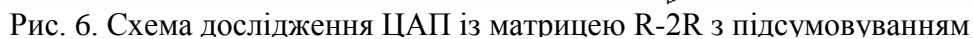
Рис. 5. ЦАП із матрицею R-2R із підсумовуванням струмів

2. Установити напругу джерела  $U_{оп}$  згідно зі своїм варіантом (табл. 2, тут тільки вхідної код 5-ті розрядний). Задати відповідні номінали опорів. Значення опору R та  $R_{33}$  вибрати самостійно.

3. Комутуючи ключі  $K_1-K_5$ , послідовно подавати на вхід схеми цифровий код з урахуванням того, що ключ  $K_1$  відповідає старшому розряду, а ключ  $K_5$  - молодшому розряду цифрового коду. Кожного разу у колонку  $U_{вих1}$  і  $U_{вих2}$



5. Зняти осцилограми. Для цього подати на вхід схеми цифровий код, який формується за допомогою генератора слів (Word Generator), а на виході операційного підсилювача підключити осцилограф (рис. 6).



1. Зібрати схему, наведену на рис. 7. Задати довільну частоту на генераторі (наприклад, 1 кГц), прямокутну форму коливань і амплітуду у межах 3 – 5 В. Установіть опорну напругу, що дорівнює номеру варіанта.



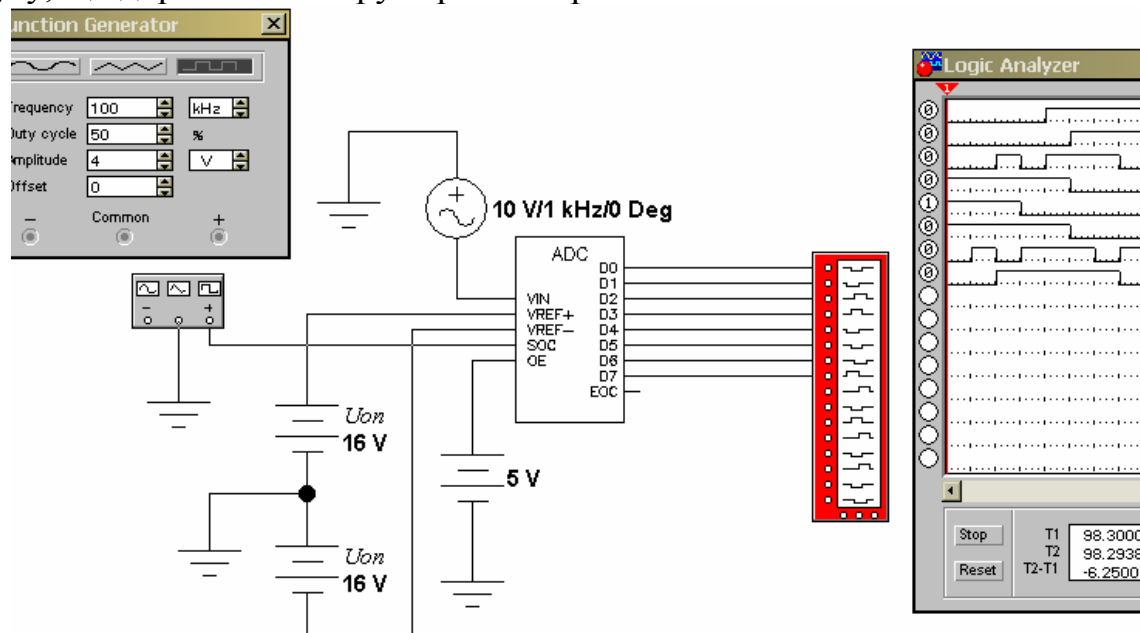
Призначення виводів мікросхеми АЦП: VIN - підключення джерела перетворюваного сигналу; VREF+, VREF- - підключення джерел опорної напруги; SOC - вхід синхронізації; OE - дозвіл на видачу вихідної двійкової комбінації на виходи D0...D7; EOC - сигнал готовності

[illegible]

даних (наприклад, при видачі даних на комп'ютер).

2. Змінюючи вхідну напругу на вході VIN, заповнити табл. 3. У останню колонку D таблиці занести розрахункове значення вихідного коду в десятковій системі числення.

3. Для схеми АЦП на рис. 8 побудувати часові діаграми. Установити опорну напругу, що дорівнює номеру варіанта. Зробити висновки.



## 5. Контрольні питання

1. Що розуміють під системою дискретного часу?
2. Яка головна вимога оптимальності методу дискретизації?
3. У чому полягає суть квантування?
4. У чому полягає сутність кодування інформації?
5. У чому сутність теореми Котельникова?
6. Назвіть та наведіть схеми найбільш поширених типів АЦП.
7. Назвіть та наведіть схеми найбільш поширених типів ЦАП.

## 3. Рекомендована література (основна, допоміжна), інформаційні ресурси в Інтернеті

### Основна

1. Болюх В. Ф., Данько В. Г. Основи електроніки і мікропроцесорної техніки: Навч. посібник. Харків: НТУ «ХП», 2011. 257 с.
2. Зиков І.С., Межерицький С.Г., Подорожняк Л.О., Хавіна І.П. Програмування мікропроцесорів у захищеному режимі: навч.-метод. Посібник. Харків : ТОВ «ДІСА ПЛЮС», 2018. 264 с.
3. Квітка С.О., Яковлев В.Ф., Нікітіна О.В. Електроніка та мікросхемотехніка: Навчальний посібник / За ред. проф. В.Ф. Яковлева. Київ: Аграрна освіта, 2010. 329 с.
4. Кравець В.О., Сокол Є.І., Рисований О.М. Комп'ютерна схемотехніка. Підручник. Харків: НТУ «ХП», 2007. 480 с.
5. Крилик Л.В., Селецька О.О. Матеріали електронної техніки: навчальний посібник. Вінниця: ВНТУ, 2017. 120 с.

6. Круліковський Б.Б., Николайчук Я.М., Шатний С.В. Мікропроцесорні системи. Практикум. Навчальний посібник. Рівне : НУВГП, 2016. 191 с.

7. Методи перетворення сигналів. Навчально-методичний посібник до самостійної роботи і контролю знань студентів / Укл.: Ю.В. Головка. Запоріжжя: ЗДІА, 2011. 61 с.

8. Методичні вказівки до виконання лабораторних робіт «Дослідження роботи ЦАП і АЦП» з дисципліни "Електронні системи" / укладачі: Є. Л. Онанченко, І. Є. Бражник. Суми : Сумський державний університет, 2012. 30 с.

9. Методичні вказівки до лабораторних робіт з дисципліни «Мікропроцесорна техніка» Укл.: В.І. Рева. Запоріжжя: ЗНТУ, 2019. 114 с.

10. Методичні вказівки до лабораторних робіт з курсу «Моделювання на ЕОМ» для студентів / Уклад. Харченко О. Л., Балев В. М. Харків : НТУ «ХПІ», 2010. 68 с.

11. Чешко І.В. Вступ до спеціальності «Електроніка» : навчальний посібник. Суми : Сумський державний університет, 2017. 148 с.

#### **Додаткова**

1. Електроніка і мікропроцесорна техніка / Сенько В.І., Лисенко В.П., Юрченко О.М., Лукін В.Є., Руденський А.А. Київ: «Агроосвіта», 2015. 676 с.

2. Кирик В.В. Мікропроцесорна техніка: Навчальний посібник. Київ: ІВЦ «Видавництво «Політехніка», 2014. 183 с.

3. Колонтаєвський Ю.П., Сосков А.Г. Електроніка і мікросхемотехніка: Підручник. 2-е вид. / За ред. А.Г. Соскова. Київ: Каравела, 2009. 416 с.

4. Матвійків М.Д., Вус Б.С., Матвійків О.М. Елементи та компоненти електронних пристроїв: підруч. для студентів ВНЗ, які навчаються за напрямом «Радіоелектрон. апарати». Львів: Вид-во Львів. політехніки, 2015. 496 с.

5. Мілих В.І., Шавьолкін О.О. Електротехніка, електроніка та мікропроцесорна техніка: Підручник. Київ: Каравела, 2008. 688 с.

6. Плахтєєв А.П., Бабешко Є.В., Ткаченко В.А., Здоровець Ю.В. Архітектури та розроблення систем Інтернету / Вебу Речей на основі вбудованих платформ. Лабораторні роботи / За ред. В.С. Харченка. Міністерство освіти і науки України, Національний аерокосмічний університет ХАІ, 2019. 147 с.

7. Теорія цифрових автоматів та формальних мов. Вступний курс : навч. посібник / Гавриленко С. Ю., Клименко А. М., Любченко Н.Ю. та ін. Харків : НТУ "ХПІ", 2011. 176 с.

8. Хіхловська І.В., Антонов О.С. Обчислювальна техніка та мікропроцесори. Підручник. Одеса: 2011. 440 с.

9. Цирульник С.М., Лисенко Г.Л. Проектування мікропроцесорних систем: навчальний посібник. Вінниця : ВНТУ, 2012. 201 с.

#### **Інформаційні ресурси**

1. KTechLab . URL: <https://sourceforge.net/projects/ktechlab/>

2. Курс електроніки. Лекції. URL: <http://vozom.ho.ua/index.html>

3. Михайлов С.Р. Основи мікропроцесорної техніки: Лабораторний практикум [Електронний ресурс] : навч. посіб. Київ : КПІ ім. Ігоря Сікорського, 2019. 59 с.

4. Основи електроніки та мікропроцесорної техніки. URL:

<http://repository.kpi.kharkov.ua/handle/KhPI-Press/18457>

5. Основи електроніки та схемотехніки. URL:  
<http://moodle.ipo.kpi.ua/moodle/mod/resource/view.php?id=12914>

6. Терещенко Т.О., Хоменко О.В. Мікропроцесорна техніка [Електронний ресурс]: конспект лекцій. Київ: КПІ ім. Ігоря Сікорського, 2017. 165 с.