

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Запорізький національний технічний університет

МЕТОДИЧНІ ВКАЗІВКИ

до виконання лабораторних робіт з дисципліни «Вбудовані
комп'ютерні системи» у середовищі MultiSim, LabView
для студентів усіх форм навчання спеціальності
8.091503 «Спеціалізовані комп'ютерні системи»
усіх форм навчання (частина I)

Методичні вказівки до виконання лабораторних робіт з дисципліни «Вбудовані комп'ютерні системи» у середовищі MultiSim, LabView для студентів усіх форм навчання спеціальності 8.091503 «Спеціалізовані комп'ютерні системи» усіх форм навчання / Укладач: Проскурін М.П. - Запоріжжя: ЗНТУ, 2019- 64 с.

Укладач: М.П. Проскурін, к.т.н., доцент

Рецензент: І.Я. Зеленьова, к.т.н., доцент

Відповідальний за випуск: М.П. Проскурін, к.т.н., доцент.

Висловлюємо окрему щиру подяку керівництву фірми National Instruments (NI), представництву NI в Україні (м. Київ) та особисто його чільнику Гладкову М.В. за сприяння в отриманні учбових версій ліцензійних програм MultiSim v.12, LabView v.12, AWR для ознайомлення студентів із сучасними САПР, навчання та проведення лабораторних, дослідницьких робіт по навчальних дисциплінах на II і V курсах в ПРЕ ЗНТУ у період 01.2017-2018 рр..

Затверджено
на засіданні кафедри
«Комп'ютерні системи і мережі»
Протокол № 10
від « 22 » квітня 2019р.

Рекомендовано до видання
НМК факультету
Протокол № 9
від « 26 » квітня 2019р.

ЗМІСТ

ВСТУП.....	5
1. Метод структурного проектування МК пристроїв	7
1.1 Мови програмування мікроконтролерів	7
1.2. Структурне програмування МК	8
2. Лабораторна робота №1. САПР Multisim и LabVIEW та мета їх сумісного використання	12
2.1 Що таке САПР Multisim и LabVIEW	12
2.1.1 Інтерфейс користувача САПР MS	18
2.1.2 Створення проекту і програмного файлу	32
2.1.3 Виконання завдання по створенню електронної схеми	35
2.1.4 Особливості інструментарію для створення спільних проектів електронних схем в MS і LV	37
2.1.5 Хід виконання ЛР і завдання за варіантами 1-12	44
2.1.6 Загальні зауваження до порядку проведення і змісту звіту	45
2.1.7 Зміст звіту.....	46
2.1.8 Контрольні питання	46
3. Лабораторна робота №2. Вивчення Принципів и особливо роботи цифроаналогових перетворювачів	46
3.1 Загальні відомості про цифроаналогові перетворювачі	47
3.1.1 Особливості роботи паралельних ЦАП з підсумовуванням струмів	47
3.1.2 Основні технічні характеристики	49
3.1.3 Хід виконання лабораторної роботи	51
3.1.4 Завдання для лабораторної роботи	52
3.1.5 Зміст звіту.....	53
3.1.6 Питання для самоконтролю	53
4. Лабораторна робота №3. Вивчення принципів і особливостей роботи аналого-цифрових перетворювачів.....	54
4.1 Основні відомості про аналого-цифрові перетворювачі	54
4.1.1 Принципи перетворення безперервних сигналів у цифрові коди	54
4.1.2 Основні параметри АЦП.....	59
4.1.3 Хід виконання лабораторної роботи	60
4.1.4 Завдання для лабораторної роботи	63
4.1.5 Зміст звіту.....	63
4.1.6 Питання для самоконтролю	63
ЛІТЕРАТУРА	64

Перелік використовуваних скорочень

АЦП - аналого-цифровий перетворювач
БД - база даних
ВКС - вбудовані комп'ютерні системи
ВП - віртуальні пристрої
ЗМР - значення молодшого розряду
КМОН - комплементарна логіка на транзисторах метал-оксид-напівпровідник
ЛР - лабораторная работа
ЛКМ - ліва кнопка миші
МК - мікроконтролер
МП - мікропроцесор
ОЗП - оперативний запам'ятовуючий пристрій
ПЗ - програмне забезпечення
ПЗП - постійний запам'ятовуючий пристрій
ПК - персональний комп'ютер
ПКМ - права кнопка миші
ППП - пакет прикладних програм
ТЗ - технічне завдання
САПР - система автоматичного проектування
ЦАП - цифро-аналоговий перетворювач
ЦА - цифровий автомат
ЕОЗ - електронно-обчислювальні засоби
NI - National Instruments
EWB - Electronics WorkBench – «робочий стіл» електронщика
LV - LabView
MS - MultiSim
SFR - (registr of special functions) - реєстр спеціальних функцій

ВСТУП

Навчальна дисципліна (НД) «**Вбудовані комп'ютерні системи**» (ВКС) є однією з завершальних в системі **знань і вмінь**, що до створення, налагодження і використання програмованих цифрових автоматів (ЦА) у вигляді перетворюючих, керуючих та операційних пристроїв на основі мікроконтролерів (МК), мікропроцесорних пристроїв і систем (МПС), для вирішення завдань в галузях науки, техніки, зв'язку, інформаційних, комп'ютерних технологій, виробництва, ін..

Курс лабораторних робіт (ЛР) по ВКС має за ціль сформувати навички фахівця - інженера системотехніки (магістра) за спеціальністю «Спеціалізовані комп'ютерні системи» (напрямок - «Комп'ютерні науки і технології») при вирішенні ним конкретних завдань на основі відомих підходів. Вивчення дисципліни ВКС надає студентам можливість **отримати**: теоретичну підготовку для складання уявної і реальної моделі простих цифрових **керуючих і операційних** пристроїв на основі цифрових інтегральних схем (ІС), МПС і МК, **ознайомитися** з їх структурними і електричними схемами, різновидами діаграм процесів обміну сигналами, інтерфейсів; **розглянути** різновиди операційних систем (ОС) для МК і їх основними характеристиками; **навчитися розробляти і аналізувати** алгоритми отримання, кодування (декодування), зберігання і переробки потоків аналогових і/або дискретних даних, що несуть інформацію про хід процесів в галузях науки, техніки і технологій (параметри: об'єктів системи, мережі, цифрових пристроїв, електродвигунів, турбін, робота, зв'язку, обробки зображень, автоматизованих комплексів, ін.).

Метою запропонованого курсу ЛР з дисципліни ВКС є **практичне** вивчення методів, підходів для розвитку і формування у студентів навичок ознайомлення із технічними вимогами, **створення та налагодження** цифрових схем для побудови типових цифрових пристроїв на базі відомих і надійних типів МП, МК згідно з отриманим варіантним завданням, **перевірка і закріплення** отриманих теоретичних знань і практичних навичок. Основою курсу ЛР є **розробка і моделювання** типових пристроїв перетворення, схем, функціональних блоків ВКС з використанням системи автоматизованого проектування (САПР) у вигляді пакетів прикладних програм (ППП) MultiSim12(13), LabView12(13) фірми National Instruments (NI) (ліцензійна учбова версія). САПР MultiSim (MS) стала подальшим розвитком відомого ППП Electronics Workbench (EWB, демоверсія), який студенти використовували при виконанні ЛР і курсових робіт з дисципліни «Комп'ютерна логіка» на молодших курсах. В свою чергу ППП LabView (LV) є інструментом для відображення і уявлення існуючих схем е ППП MS або створених систем вимірювання, контролю, а також систем автоматизації управління на основі технології віртуальних приладів (ВП). Ефективність

використання ППП LabView полягає в тому, що можливо як **розробляти логіко-математичну і візуально-інформаційну модель об'єкта**, так і постачати цю модель експериментальними/тестовими даними за допомогою апаратних засобів-пристроїв введення/виведення (ПВВ), пов'язаних з реальним (або віртуальним) об'єктом-наприклад, зі схемою з САПР MS.

В результаті проведення і захисту ЛР **студенти повинні знати**: вигляд і форму технічного завдання (ТЗ) по створенню конкретного пристрою, блоку, схеми; основні засоби отримання, зберігання, перетворення і обробки аналогових, цифрових інформаційних потоків; згідно вимог ТЗ проводити вибір необхідного МП, МК та його периферійних пристроїв ПВВ (давачів, цифро-аналогових і аналого-цифрових перетворювачів, генераторів кодів, команд, адресів/даних з блоків регістрів і пам'яті, ін.); **вміти розробляти** схеми, алгоритми і налагоджувати програми для них; використовувати для цього моделі МП, МК, ІС ППП типу EWB, MS, LV, Proteus, ін..

Виконання студентами ЛР з курсу ВКС дозволяє їм ознайомитись з основними етапами формування технічного завдання (ТЗ) на розробку цифрового керуючого і/або операційного пристрою, скласти його блок-схему і модель функціонування, провести макетування і верифікацію функціонування в ППП MS, а також відобразити процес представлення її у інтуїтивно зрозумілому вигляді в ППП LV і використання ВП (з можливістю подальшого використання продукту від **NI** для проектування/виготовлення плати **Utilboard 2012**). Цикл ЛР призначений для вивчення принципів побудови і функціонування елементів ВКС, побудованих на базі МК сімейства MCS 51/52 з використанням ППП MS. МК сімейства MCS 51/52 обрані за основу для моделювання з кількох причин:

- наявність моделі МК в бібліотеці моделей (БМ) демоверсій MS, Proteus, ін.;
- кількість поколінь МК, що створені на основі архітектури MCS 51/52, ін.;
- спрощені моделі перетворення даних на прикладі використання відомих МК для більшого розуміння їх суті студентами, які мають невеликий обсяг навиків по розробці і схемотехніці в ППП MS і LV;
- освоєння типових прийомів, підходів в проектуванні ВКС на основі простих видів МК з прикладами ПЗ (Асемблер, С, ін.) і типових схемних рішень, ін.;
- відсутність необхідності вибору і завантаження ОС, її тестування в МК (з можливістю використання ін. МК з ОС для керування, наприклад, кількох операційних пристроїв на основі безопераційних MCS 51/52).

Основний блок ЛР побудовано із джерела [1], які доповнено можливостями використання додаткового ППП LV разом з ППП MS, причому курс ЛР включає варіанти проектування таких схем: аналогові, аналогово-цифрові, цифроаналогові, цифрові, арифметичний пристрій (+, x, - /) модель «конвеєр».

Опис особливостей МК сімейства MCS 51 надано в Додатку А.1 (див.ч.2) або наведено у відповідній літературі, мережі Internet [2-4].

1. Метод структурного проектування МК пристроїв

Основна ідея методу полягає в тому, що при **використанні регламентованої послідовності дій по проектуванню апаратних засобів і виборі певної структури програми** вдається розробити мікроконтролерну систему, що відповідає поставленим технічним вимогам.

Слід зазначити, що метод структурного проектування мікроконтролерних пристроїв хоча і не гарантує обов'язково успішне завершення проекту, але значно збільшує ймовірність його виконання. Рекомендується наступні етапи структурного проектування [4], при цьому в проектах різної складності деякі із зазначених етапів можуть бути об'єднані, наприклад, 3 і 4;

1) детальний аналіз вимог, що пред'являються до проектуваного пристрою, на основі яких формуються технічні характеристики розроблюваного устрою;

2) декомпозиція проектуваної системи на кілька ієрархічно взаємопов'язаних функціональних підсистем (модулів) з визначенням взаємозв'язків між ними. Для кожної підсистеми визначається виконувана функція, входи і виходи, без конкретизації внутрішньої структури (модель «чорного ящика»);

3) розробка структурної схеми апаратури і алгоритму її функціонування;

4) визначення способу реалізації кожного функціонального модуля (вибір апаратних і програмних рішень);

5) комплексне налагодження проектуваного пристрою.

У циклі лабораторних робіт (ЛР), що виконується студентами, тема кожної пов'язана з проектуванням типового функціонального модуля на МК. Спільна реалізація певної сукупності функціональних модулів, розроблених у ході виконання ЛР, забезпечить успішне проектування пристрою на основі МК для ВКС, що відповідає варіанту технічного завдання (ТЗ) магістерської дипломної роботи (ДР).

1.1 Мови програмування мікроконтролерів

Розробники систем на МК пишуть свої програми або на асемблері (машинно-орієнтований мова), або на мовах високого рівня (МВР). В даний час в організаціях розробників ВКС, як правило, застосовується мова програмування С (його модифікації С+, С++), що забезпечує хороший доступ до апаратних ресурсів МК або асемблер.

Як будь-яка інша мова високого рівня, **мова С** (її версії) дозволяє розробити такий вихідний текст програми, який володіє властивістю **переносимості** коду. Це означає, що програма, написана на С для одного

МК, потім може бути скомпільована іншим компілятором для МК з відмінним процесорним ядром. Для того, щоб програма володіла властивістю переносимості, синтаксис мови високого рівня для різних компіляторів повинен бути однаковий, такий, наприклад, як мова С стандарту ANSI. Основна мета стандартизації полягає в тому, щоб забезпечити розробнику можливість написання типових функцій управління (функціональних модулів) один раз з подальшим їх багаторазовим використанням у проєктах з іншими МК.

Мова С також забезпечує хорошу **читаність** коду. Ця властивість означає, що програміст, який не є розробником програми, може з початкового тексту програми зрозуміти, який алгоритм реалізований і як програма працює. Ще однією перевагою С (як мови високого рівня) є наявність **бібліотек математичних функцій** над числами, представленими в різних форматах, в тому числі і в форматі з плаваючою точкою.

Асемблер, в порівнянні з мовою С, має важливу перевагу, а саме: **добре написана на асемблері програма виконується за менший час і займає в пам'яті менший обсяг, ніж програма на С.**

Саме ці характеристики: час виконання і розмір програмного коду є критичними для додатків, що використовуються в проєктуванні ВКС, де елементна база має відносно невисоку швидкодію, пам'ять програм обмежена в обсязі, а кошторис ВКС не має бути затратним.

Практика спільного застосування мов С і асемблера в одному проєкті показує, що такий підхід дозволяє отримати оптимальний результат як з точки зору економії часу на розробку, так і часу виконання програми [4]. Основна частина прикладної програми, в якій виробляється перетворення даних, пишеться на С, в той час, як критичні за часом реалізації фрагменти алгоритму слід оформити на асемблері. У деяких ситуаціях асемблер також використовують для програмної підтримки зовнішніх периферійних ІС.

У зв'язку з вищесказаним в методичних вказівках до ЛР наводяться приклади реалізації функціональних модулів як з використанням мови С, так і асемблера МК. Студенти можуть вибрати мову програмування самостійно, проте слід зазначити, що **метод структурного проєктування МК**, який дозволяє збільшити швидкість написання програм і полегшити їх налагодження за рахунок скорочення кількості доступних програмних конструкцій, найкращим чином відображається на програмі, написану на С.

1.2. Структурне програмування МК

Структурне програмування є складовою частиною структурного проєктування мікроконтролерного пристрою. При розробці програм для МК і їх налагодженні, як правило, використовуються два способи програмування: від низу до верху і зверху донизу. При написанні програми **від низу до верху**

приступити до її налагодження не можна, не завершивши повністю її програмування. До того ж, кожна з невиявлених програмних помилок може призвести до неприцездатності розроблювального пристрою, може проявитися і протилежна ситуація: внаслідок допущеної апаратної помилки не функціонує програмне забезпечення.

При використанні технології програмування **зверху вниз** на будь-якому етапі вона може бути відтрансльована і виконана, при цьому відстежується виконання всіх фрагментів алгоритму, написаних до цього часу. Мова С найкращим чином відповідає такій технології програмування. На початковому етапі розробки програма складатиметься з основної функції (main.c) - програми управління, в якій всі викликані функції симулюються порожніми програмними модулями (заглушками). У міру просування зверху вниз все більше число функцій наповнюється змістом, і для них записується вихідний текст. Для кращої читаності програмного коду бажано, щоб назва підпрограми/функції відображала алгоритмічну дію. Приклад такого підходу [5] показаний в представлений нижче програмі.

```
// Підпрограма читання порту
void Prochitat_Port (void)
{
// Поки це тільки підпрограма заглушка!
}
// Підпрограма включення індикатора
void Vkluchit_Indicanor (void)
{
// Поки це тільки підпрограма заглушка!
}
// Основна програма
void main (void)
{
Prochitat_Port (); // Прочитати порт
Vkluchit_Indicanor (); // Включити індикатор }

```

Застосування підпрограм є не тільки засобом структурування програми, але і істотно скорочує розмір машинного коду, якщо підпрограма багаторазово використовується, хоча при цьому зменшується швидкодія розроблювального пристрою.

Основна ідея структурного програмування полягає в тому, що використовуються тільки чотири структурних оператора (прямокутник - операційний, ромб - умовний, їх комбінації), які дозволяють отримати програму будь-якої складності.

1. Лінійний ланцюжок операторів. Ця конструкція часто використовується, оскільки підзадачі можуть бути розбиті на більш прості послідовно виконувані дії (див., наприклад, структуру головної програми).

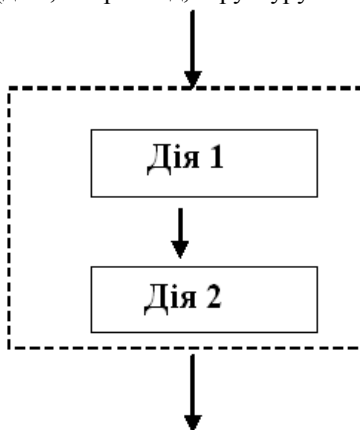


Рисунок 1.1 - Блок-схема конструкції управління «лінійний ланцюжок операторів»

2. Умове виконання операторів

Ця конструкція використовується, коли поточна дія залежить від результату виконання попередньої частини програми. Далі можливий рух або за єдиною гілкою (з кількох) програми (рис.1.2) при виконанні/невиконанні умовного оператора.

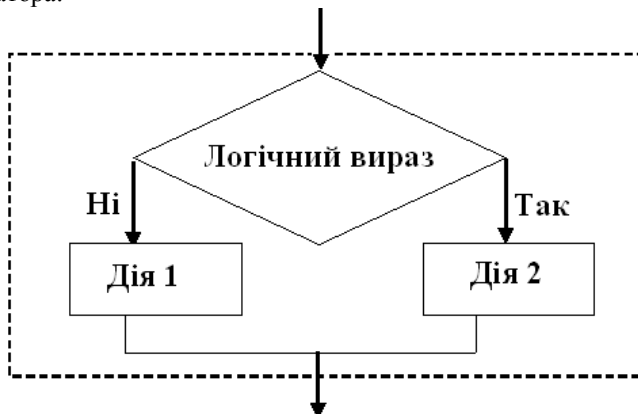


Рисунок 1.2 - Блок-схема конструкції управління «Умове виконання операторів»

3. Циклічне виконання оператора з перевіркою умови після тіла циклу

Всі оператори, які повинні повторюватися в процесі виконання цієї конструкції, називаються тілом циклу. При виконанні цих операторів зазвичай модифікується деяка змінна (параметр циклу), яка впливає на завершення циклу. Такий оператор реалізує конструкція мови С "do ... while". Блок-схема конструкції представлена на рис. 1.3.

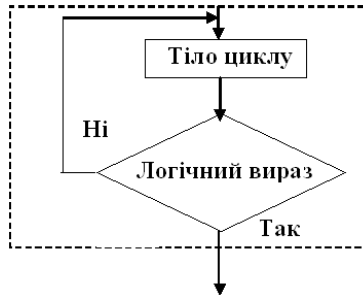


Рисунок 1.3 - Блок-схема конструкції управління «циклічне виконання оператора з перевіркою умови після тіла циклу»

4. Циклічне виконання оператора з перевіркою умови до тіла циклу

На відміну від попередньої конструкції, тіло циклу може жодного разу не виконуватися (рис. 1.4).

В процесі написання програм зазвичай накопичуються підпрограми і фрагменти коду, які можна використовувати в інших програмах. Можна копіювати такі фрагменти з програми в програму, проте це призводить до незручностей: програмний код збільшується, виникають складності при написанні і редагуванні програми.

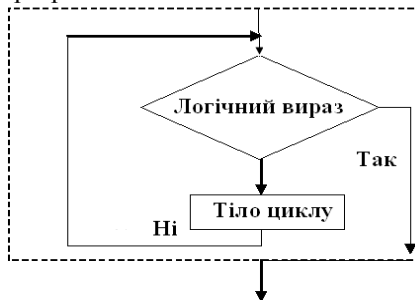


Рисунок 1.4 - Блок-схема конструкції управління «циклічне виконання оператора з перевіркою умови до тіла циклу»

Крім того, при зміні типів мікросхем периферійних пристроїв, наприклад, зміні АЦП або індикатора, доводиться шукати раніше скопійовані ділянки коду і замінювати їх новими. Це може викликати нові помилки і ускладнює написання та налагодження програми. Тому, простіше зберігати вихідний текст програми в декількох файлах, а поєднання цих файлів в єдину програму буде виконувати транслятор. Як правило, в один програмний файл поміщається ділянка коду, що працює з якимось конкретним пристроєм або відповідає за певну підзадачу.

У мові C і асемблері для цих цілей використовується ключове слово (Директива) `include`. Наприклад, в мові C-51* директива включення програмного файлу опису регістрів спеціальних функцій МК з зазначенням їх фізичних адрес запишеться так: `#include <reg51.h>`. Такий файл називається заголовним, так як зазвичай поміщений на початку програми. У програмі може бути кілька заголовків файлів.

2. Лабораторна робота №1. САПР Multisim и LabVIEW та мета їх сумісного використання

Мета ЛР: ознайомити студентів з продуктами фірми NI САПР MultiSim и LabView [6-8], показати можливість їх спільного використання при проектуванні елементів ВКС на прикладі дослідження пасивної радіоелектронної схеми - ємнісного входного контуру Wi-Fi зв'язку.

2.1 Що таке САПР Multisim и LabVIEW

САПР Multisim (MS) - це унікальна можливість розробки схеми і її тестування та емуляції з одного середовища розробки. У такого підходу є безліч переваг. Новачкам в ППП MS не потрібно турбуватися про складний синтаксис SPICE (Simulation Program with Integrated Circuit Emphasis - програма емуляції з вбудованим оброблювачем схем) і його команд, а у просунутих користувачів є можливість налаштування всіх параметрів SPICE. Завдяки Multisim опис схеми став як ніколи простим і інтуїтивно зрозумілим. Подання у вигляді електронної таблиці дозволяє одночасно змінювати характеристики будь-якої кількості елементів: від схеми друкованої плати до моделі SPICE.

Безрежимне редагування - це найбільш ефективний спосіб розміщення і з'єднання компонентів. Працювати з аналоговими і цифровими складовими елементами інтуїтивно просто і зрозуміло.

Крім традиційного аналізу SPICE, MS дозволяє користувачам підключати до схеми віртуальні прилади (ВІП).

* В мову програмування C-51 (різновид C для МК сімейства MCS-51) введений ряд змін, що відображають особливості побудови їх пам'яті, портів, регістрів, ін..

Концепція віртуальних інструментів - це простий і швидкий спосіб побачити результат за допомогою імітації реальних подій. Також в MS є спеціальні компоненти під назвою «інтерактивні елементи» (Interactive parts), які можна змінювати під час їх емуляції. До інтерактивних елементів відносяться перемикачі, потенціометри, найменші зміни елемента відразу відображаються в імітації.

При необхідності більш складного аналізу MS пропонує більше 15 різних функцій аналізу. Деякі приклади включають використання змінного струму, Монте-Карло, аналіз найбільш несприятливих умов і Фур'є. У MS входить Grapher – потужна засіб перегляду та аналізу даних емуляції. Функції опису і тестування схеми, представлені в MS допоможуть будь-якому розробнику схем, заощадають його час і врятовують від помилок на всьому шляху розробки схеми.

САПР LabView (LV) - це оригінальний ППП для створення систем виміру, а також систем автоматизації управління на основі технології віртуальних приборів. До недавніх пір фахівці при вирішенні задач в власній предметній області були змушені вдаватися до допомоги професійних програмістів які, як правило, не є носіями знань в цій області. Таке посередництво найчастіше збільшувало матеріальні і тимчасові витрати, а найголовніше, знижувало якість досліджень і розробок. Навіть використання спеціалізованих програмних засобів лише частково знімало цю проблему. Поява програмних продуктів останнього покоління з досить дружніми інтерфейсами, адаптованими до менталітету і професійних навичок фахівців, зробило можливим їх використання фахівцями безпосередньо, не вдаючись до допомоги посередників. До таких нових програмних продуктів належить LW, що має вельми зручний призначений для користувача інтерфейс і потужні засоби графічного програмування. З кожною подальшою версією LV зростає рівень інтелектуалізації інтерфейсу користувача і зручність його використання.

LV- ППП в комплексі з такими апаратними засобами, як **вбудовувані в комп'ютер багатоканальні вимірювальні** аналого-цифрові (АЦП) і цифро-аналогові (ЦАП) перетворювачі, плати захоплення і синхронізації відеозображення для систем машинного зору, **плати управління** рухом і виконавчі механізми, а також **вимірювальні прилади**, що підключаються до комп'ютера через стандартні інтерфейси RS-232, RS-485, USB, GPIB (КОП), PXI, VXI, що дозволяє розробляти ВКС вимірювання, контролю, діагностики та управління практично будь-якої складності. LV має власну потужну математичну підтримку. Крім того, L може інтегрувати в себе програми, написані в середовищі MatLab. Велика кількість вбудованих алгоритмів цифрової обробки одновимірних і двовимірних сигналів дозволяє здійснювати досить складну обробку сигналу, зображення і

експериментальних даних в тимчасовій, просторовій і спектральній областях. ППП LV постійно розширюється новими засобами обробки сигналів на основі вейвлет-аналізу, алгоритмів нечіткої логіки, мережевих технологій і т.д.

Роль програмних середовищ, подібних LV, в наукових дослідженнях і технічних експериментах досить велика. В даний час в науці спостерігається свого роду «ренесанс» експерименту. Викликано це розвитком і вдосконаленням вимірювальної техніки і появою нового покоління високоточних і високочутливих вимірювальних приладів і автоматизованих вимірювальних систем, за допомогою яких реєструються тонкі ефекти 3..4-го порядку малості. Незважаючи на свою малість, ці ефекти часто грають ключову роль у формуванні основного явища, впливу на якість кінцевої продукції і т.д. Аналіз подібних ефектів, як правило, виявляється поза досяжністю теоретичних моделей. Тому, в більшості випадків експеримент є єдиним джерелом якісно нової і надійної інформації. При цьому результат досягається набагато швидше, ніж методами «Чистої» теорії. Найчастіше це вигідно і економічно, так як рядовий експериментатор, озброєний сучасним інструментом подібним LV, може скоро отримати потрібну інформацію.

Традиційно для дослідника **функції моделювання** і експериментування розділені. Моделювання здійснюється в середовищі математичних програмних пакетів, а експерименти підтримуються іншими програмними засобами, що аж ніяк не підвищує ефективність досліджень. **Ефективність використання середовища LV** в наукових дослідженнях полягає в тому, що, залишаючись в її рамках, можна розробляти як математичну модель об'єкта, так і постачати цю модель експериментальними/тестовими даними за допомогою апаратних засобів введення-виведення, пов'язаних з віртуальним/реальним об'єктом.

Зв'язок САПР MS і LabVIEW може здійснюватися **ряд способів (залежать від № версії у міру розвитку самих САПР):**

1. Контрольно-вимірювальні **віртуальні прилади (ВП)** написані для LV і працюють в MS, починаючи з версії 10.1.

В посиланнях <http://zone.ni.com/devzone/cda/tut/p/id/5635>
<http://zone.ni.com/devzone/cda/tut/p/id/13715> наведені готові ВП, які можна завантажити і додати в MS (і скорегувати, якщо потрібно).

Моделювання ведеться в MS, а на LV можна виконати генератори, вимірювальні прилади, змінні регулятори (резистори, індуктивності, конденсатори, ін.). Особливості ВП в тому, що вони генерують сигнали (**масив даних Waveform в термінах LV**) до запуску моделювання, а після запуску моделювання модель працює з цим попередньо створеним масивом даних Waveform. Тобто така модель **не працює з даними в реальному часі**, а тільки з **попередньо генерованими даними**. Вимірювальні ВП викликаються в кожному циклі моделювання і отримують по 1 точці

приєднання в схемі кожен виклик (з кожного входу - тільки один відлік). Циклічність виклику вимірювального приладу **визначається настройками циклу моделювання**, що накладає обмеження на деякі можливості цих LV-приладів.

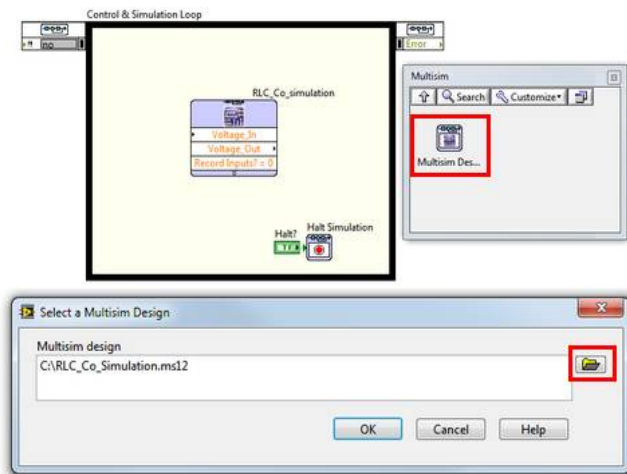
Особливістю є те, що такі ВП, які генерують і вимірюють, **складно налагоджувати окремо від MS**, тому що їх внутрішня логіка сильно зав'язана з MS, і повноцінне налагодження гальмується: доводиться при кожній зміні налаштувань копіювати прилад в MS і перезапустити його (без MS прилади не можна запускати тільки засобами LV). По цій темі можливо переглянути / використовувати дані спільноти розробників приладів MS (моделі готових ВП): <https://decibel.ni.com/content/groups/MS-custom-simulation-analyses-and-instruments-powered-by-labview?view=overview>

2. **Процес Ко-емуляції:** Digital and Analog Co-simulation Between NI LV and NI MS (<http://www.ni.com/white-paper/13663/en/>), йде в комплекті з MS, починаючи з 12-ї версії <http://www.ni.com/product-documentation/5590/en/>. Для роботи необхідний LV Control Design and Simulation Module (<http://www.ni.com/download/labview-control-design-and-simulation-module-2015/5449/en>).

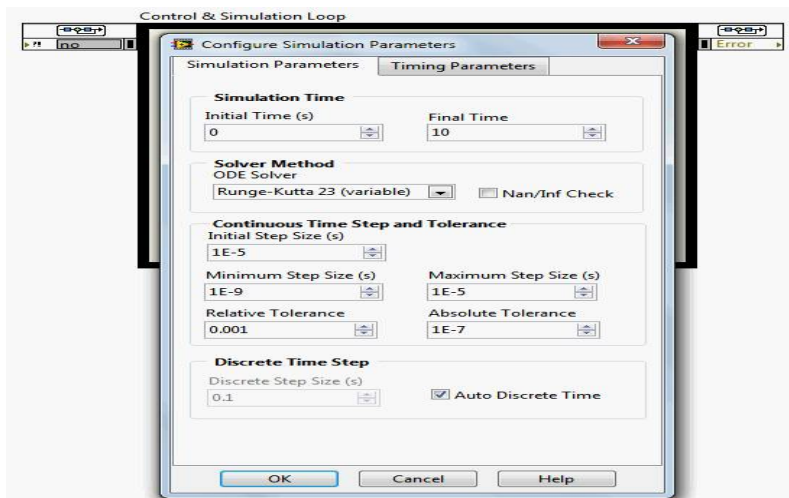
У LV для моделювання додається спеціальна конструкція – цикл моделювання. У неї додається блок (в який завантажується схема MS - файл .ms12) і в даному випадку **настройки процесу визначаються в LV**, а не в MS - рис.2.1. Для цього способу вибирають / існують настройки (періоди обчислень, тобто в такому режимі йде покрокове моделювання, кожен цикл на виході моделі формує одне значення на кожен вихід (**output1,2..k**) в режимі точка за точкою (point by point), т.д. Можливо переглянути додатково описи, документи, ін.: <http://zone.ni.com/devzone/cda/tut/p/id/13663>, <http://zone.ni.com/devzone/cda/tut/p/id/13721>.

3. Для зв'язку **MS (версія 12)** з LV може застосовуватися «Движок» (<http://www.ni.com/tutorial/5933/en/>): An Introduction to MS Automation with the **LV MS Connectivity Toolkit (beta)**, який можна завантажити: <http://forums.ni.com/t5/NI-Labs-Toolkits/LabVIEW-MS-Connectivity-Toolkit-Beta-version-0-2-Now-with/ta-p/3526121>

Розробнику досить зібрати схему в MS, повторити її (згідно певними правилами, див. нижче), провести моделювання в LV і отримати результат в залежності від введених налаштувань на ВП схеми: <http://zone.ni.com/devzone/cda/tut/p/id/7785>. Цей «движок» взаємодіє з MS своєрідно (мається деяка схожість зі способом 2): в ньому існує набір блоків, які дозволяють викликати і працювати з MS, як з ActiveX сервером (фреймворк для визначення ППП, придатних до використання з програмами, написаними на різних мовах програмування).



а



б

Рисунок 2.1- Вибір підменю MS - а, настройки цикла моделирования - б

Як це працює: відкривається з'єднання з ActiveX сервером MS (у якого немає графічної оболонки), але є **методи і властивості** до яких можна програмно звертатися. Далі проводиться завантаження електронної схеми моделі (з файлу), настройка параметрів і запуск моделювання. Тобто

моделювання відбувається в MS, а дані потім через **виклик методів і властивостей сервера ActiveX** повертаються в LV. Крім того є можливість змінювати параметри елементів схеми (значення опорів, ємностей, тощо) і отримувати зображення схеми (у вигляді статичної картинки png, тощо).

Результати моделювання виходять у вигляді масиву значень Waveform (Див. Сп.1) - на кожен виклик функцій опитування даних; також можливо вибирати різні варіанти аналізу.

У свою чергу для версії **MS 13** встановлюється **MS Automation API** для **LV** (<http://www.ni.com/white-paper/14836/en/>), який є аналогом розглянутої **LV MS Connectivity Toolkit (beta)**.

MS Automation API, який дозволяє виробляти і отримувати дані моделювання MS через COM-інтерфейс. Тобто API дозволяє програмно керувати моделюванням MS без необхідності перегляду процесу в ньому.

Таким чином клієнтські файли, описані на будь-якому COM-Aware мовою програмування (таких як LV) можуть отримувати доступ до MS через цей інтерфейс, а так само управляти і отримувати дані моделювання.

Через **LV MS Connectivity Toolkit (beta)** або **MS Automation API** можливо:

1. Відкривати і закривати існуючі схеми MS.
2. Змінити тон в місці встановленого джерела струму або напруги.
3. Керувати запуском, зупинкою, паузою і т.п. моделювання.
4. Отримувати результати моделювання від встановленого «пробника».
5. Отримати перерахування компонентів схеми.
6. Замінити компоненти компонентами з бази даних.
7. Отримати і встановити значення частин електронної схеми - резисторів, конденсаторів і котушок індуктивності, їх комбінацій.
8. Отримати варіанти схемних рішень і зображення схем, встановити активний варіант для моделювання.
9. Створити звіт за схемою, включаючи перелік елементів і список з'єднань.
10. Змінювати / замінювати компоненти: схеми на верхньому рівні проектування, підсхеми і ієрархічні блоки не можуть бути змінені.

4. Обмін через файли LVM (TDM) між окремою програмою, що виконується в середовищі LW і MS. Переглянути, як це реалізується можна в відео **lvm.wmv** і **MS.swf** в каталозі **MS**. Для завантаження lvm в MS і запис результатів моделювання існують вже готові ВП, написані в LV (спосіб1).

Важлива особливість: при експорті в TDM створюються сигнальні групи, що **об'єднують сигнал і тимчасові мітки**; а в LVM теж для кожного стовпчика сигналу є **відповідний стовпець тимчасових міток**. Крім того, в LVM є delta T - часовий крок між відліками, а ось в TDM MS (судячи з

усього) ΔT не записує, тому цей інтервал доведеться самостійно витягувати з шпальти часу сигнальної групи.

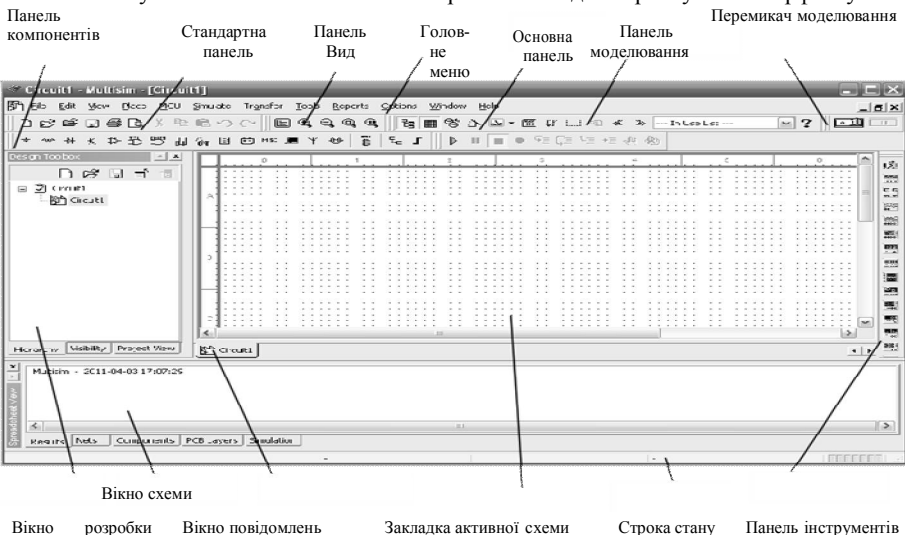
Всі перераховані способи реалізують різні парадигми. Так, способи 1, 2 орієнтовані на обробку даних, що надходять в реальному часі (точка за точкою) від моделі в MS. 3-й спосіб оперує блоками даних, що надходять попередньо Waveform (середнє між способами 1, 2 і 4). 4-й спосіб оперується тільки результатами моделювання.

2.1.1 Інтерфейс користувача САПР MS

2.1.1.1 Основні елементи меню користувача

Після установки програми в комп'ютері її запуск здійснюється наступним чином: Пуск-Всі програми-National Instruments-Circuit Design Suite 10.0-MS.

Рисунок 2.1 - Основні елементи призначення для користувача інтерфейсу



Для активізації вікна повідомлень необхідно натиснути на основній панелі значок або вибрати в пункті меню View-Spreadsheet View. Головне меню програми MS володіє великим набором інструментів для підготовки схеми і проведення аналізу.

Пункти головного меню File, Edit, View є звичайними меню з набором команд для роботи з файлами і проектами, друку (File), редагування і зміни властивостей креслення, орієнтування, видалення, виділення, переміщення елементів схем (Edit), настройки призначеного для користувача інтерфейсу з можливістю зміни набору інструментальних панелей, масштабування робочої області (View).

пункт меню Place:

Component... - призначений для вибору і розміщення компонентів схеми;

Wire - нанесення сполучних проводів; Bus - побудова інформаційних шин;

Junction - визначення електричних вузлів-з'єднувачів;

Comment - визначення коментаря в схемі;

Connectors - введення з'єднувачів;

Text - введення пояснювального тексту;

Graphics - вставка елементарних графічних елементів і графічних зображень із зовнішнього файлу;

Title Block - завдання параметрів штампа для схеми.

Пункт меню MCU дозволяє записати програмний код для роботи обраного мікропроцесора і налагодити програму.

Пункт меню Simulate задає типи проведеного аналізу, дозволяє вибрати вбудовані прилади, зберегти результати і провести їх обробку. У цьому меню можна змінити часовий крок моделювання за допомогою пункту Interactive Simulation Settings (рис. 2.2).

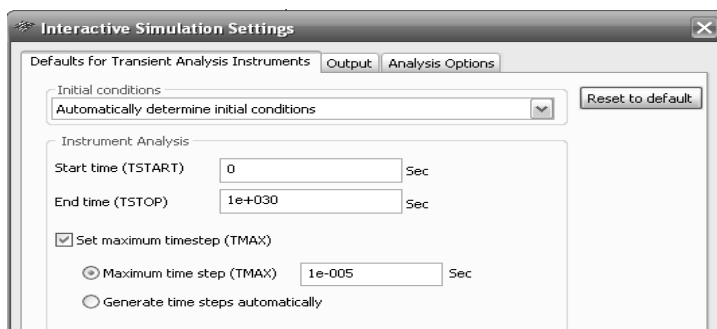


Рисунок 2.2 - Вікно настройки тимчасового кроку моделювання

У програмі за умовчанням вибрана **автоматична генерація кроку моделювання** (10^{-5} с). Для зміни часу кроку моделювання активувати Maximum time step (TMAX), вказати необхідний крок.

Пункт меню Transfer - тут можна передати вихідні дані для трасування друкованих плат в програму Ultiboard.

Меню Tools дозволяє працювати з базою даних компонентів програми, використовувати можливості автоматизованого проектування, майстри проектування наявних типових пристроїв на основі таймера 555 серії, смугових фільтрів, операційних підсилювачів і каскадів посилення на біполярних транзисторах, для яких можна вибрати необхідні вихідні

параметри. У цьому меню також можна підібрати вихідні дані для багатоваріантного аналізу досліджуваної схеми, перевірити схему на помилки, редагувати імена електронних компонентів і навіть реалізовано подальше збереження отриманої картинки у вигляді графічного файлу.

Пункт меню Reports пропонує детальний звіт про схему: числі і типі компонентів, їх параметрах, відомості про вузли схеми та багато іншого. Крім цього отриману інформацію можна передати в офісні програми для подальшого використання. Пункт меню Option задає умови роботи з підготовки і з-зберігання схеми, визначення зовнішнього вигляду схеми і умов введення і розміщення елементів при малюванні. Він містить наступні підменю: Global Preferences..., Sheet Properties... и Customize User Interface...

Підменю Global Preferences визначає режими и умови роботи програми як у процесі вводу схеми, так і при збереженні введенної схеми у вигляді файла. Так, на закладці Paths (рис. 2.3) надається шлях до тек збереження файлів схем, файлів конфігурації і БД.

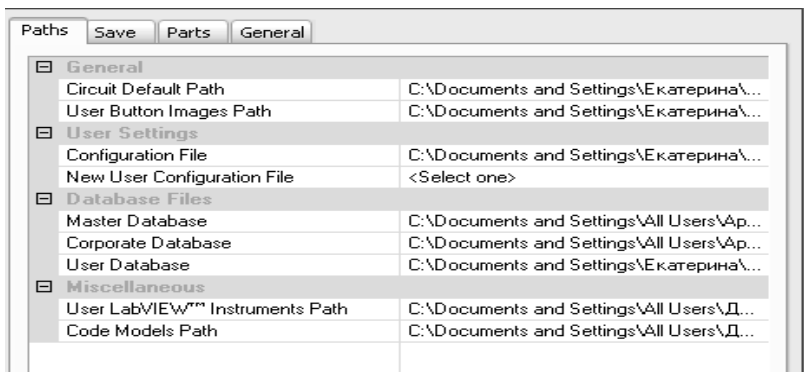


Рисунок 2.3 - Вкладка Paths підменю Global Preferences пункта меню Option

На закладці Save (рис. 2.4) визначаються режими збереження і розміру файлу даних.

На закладці Parts (рис. 2.5) визначаються дії програми при виборі і установці компонента в робочу область схеми, вибір стандарту виду компонента, параметри автоматизації вимірювання результатів аналізу, параметри моделювання цифрових пристроїв.

На закладці General (рис. 2.6) визначаються дії при русі коліщатка миші, задаються можливості автоматизації з'єднання провідників і можливості вибору виділяються рухом курсора областей.

Підменю Sheet Properties визначає зовнішній вигляд підготовленої, намальованої схеми. На її закладці Circuit (рис. 2.7) можна вказати, яка інформація буде відображатися на поле схеми біля введенного компонента.

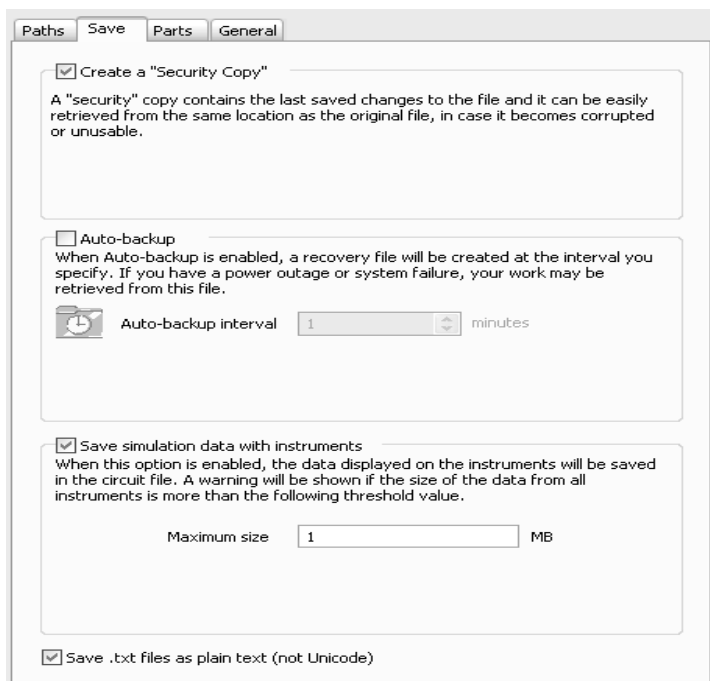


Рисунок 2.4 - Вкладка Save підменю Global Preferences пункта меню Option

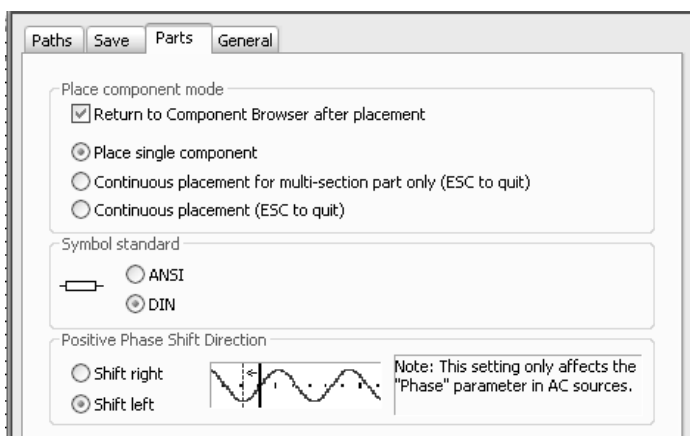


Рисунок 2.5- Закладка Parts підменю Global Preferences пункта меню Option

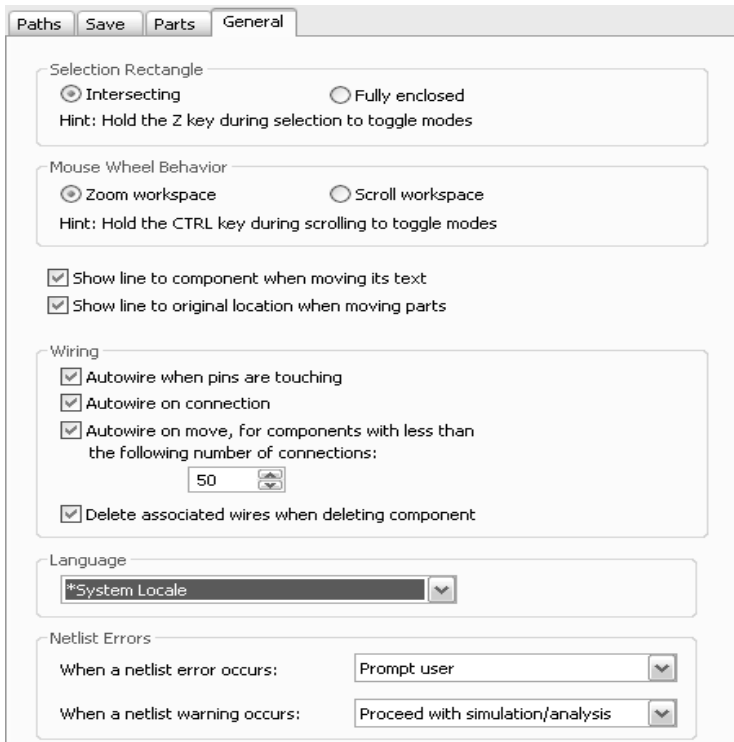


Рисунок 2.6 - Закладка General підменю Global Preferences пункта меню Option

Рекомендується залишити для відображення тільки найнеобхіднішу інформацію: текстові мітки (Labels), позиційне позначення (RefDes), його значення (Values). За допомогою цієї ж закладки можна задати також колірне рішення елементів, з'єднувальних проводів, фону схеми зі стандартних наборів або ж задати призначену для користувача колірну палітру (Color). Якщо потім натиснути екранну кнопку вікна ОК, то всі вибрані параметри зберуться тільки для поточної схеми і не будуть використовуватися в наступних схемах. Для збереження налаштувань для наступних схем необхідно відзначити прапорець Save as default і після цього натиснути ОК.

Закладка Workspace (рис. 2.8) дозволяє встановити видимість сітки, межі креслення, показ межі сторінки, визначити розмір сторінки для схеми зі стандартних наборів, її орієнтацію або ж визначити нестандартні розміри сторінки і метрику розмірів. Закладка Wiring (рис. 2.9) визначає товщину ліній з'єднувальних проводів і товщину зображення шин.

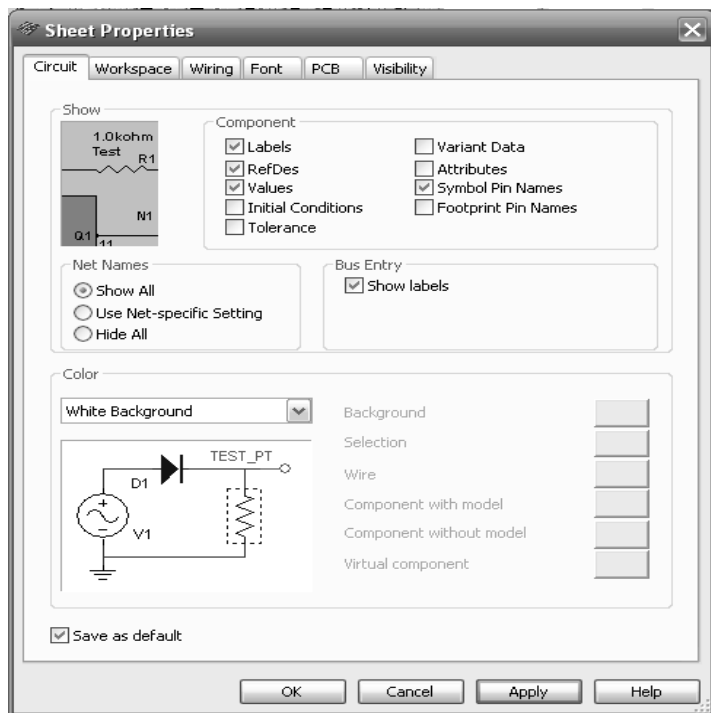


Рисунок 2.7- Вкладка Circuit підменю Sheet Properties пункту меню Option

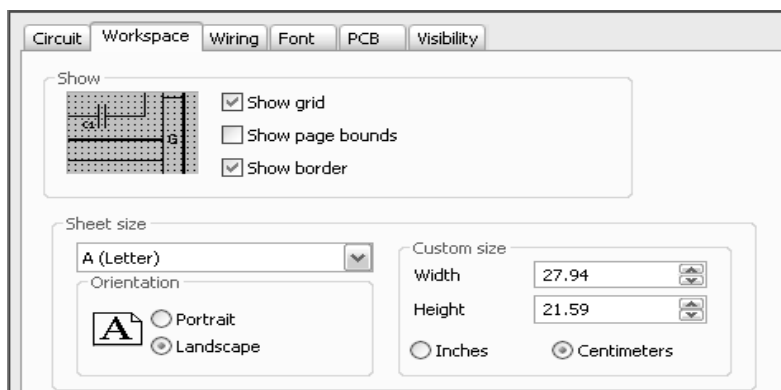


Рисунок 2.8 - Вкладка Workspace підменю Sheet Properties пункту меню Option

Пункт меню Window - стандартний, здійснює навігацію між внутрішніми вікнами програми і регулює розташування цих вікон. Аналогічне можна сказати і про пункт меню Help.

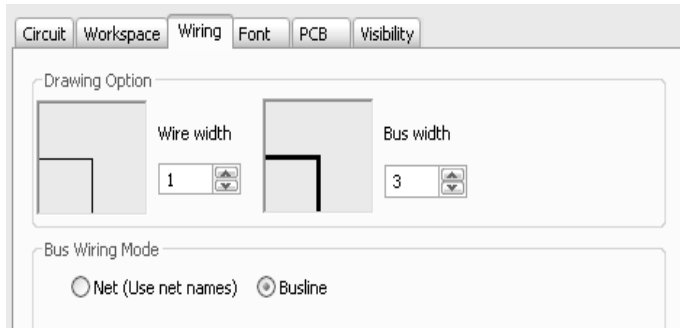


Рисунок 2.9 - Вкладка Wiring підменю Sheet Properties пункту меню Option

2.1.1.1 Віртуальні інструменти

Програма MS містить велику кількість віртуальних вимірювальних приладів (інструментів), які можна використовувати з метою проведення вимірювання або ж дослідження схемотехнічних рішень. Віртуальні прилади (ВП) за своєю дією відповідають реальним вимірювальним приладам. З їх допомогою можна не тільки візуалізувати інформацію, а й зберегти її у вигляді файлу даних, який в подальшому можна буде використовувати для обміну з іншими програмами, наприклад з LV.

Панель інструментів на екрані може бути розташована вільно, але, як правило, вона закріплюється біля кордонів вікна (зверху або зправа). Вид панелі представлено на рис. 2.10



Рисунок 2.10 - Вид панелі інструментів

Вимірювальні прилади можуть мати різний зовнішній вигляд, в залежності від того, яке завдання ставить перед собою користувач і де розташований сам прилад (на панелі інструментів або на полі схеми), приклад показаний в табл. 2.1.

Таблиця 2.1- Подання віртуальних приладів в MS

Форма представлення	Опис	Зовнішній вигляд
Іконка	Представляє іструмент в панелі інструментів в Multisim від NI	
Символ	Представляє інструмент в ланцюгу схеми. Для під'єднання до схеми необхідно використовувати зовнішні висновки інструменту. Для відкриття приладової панелі необхідно двічі клацнути ЛКМ на символі інструменту в ланцюгу схеми.	
Інструментальна панель (панель прилада)	Відкривається подвійним клацанням ЛКМ на символі інструменту. Дозволяє користувачеві взаємодіяти з інструментом - встановити параметри вимірювання. Показує результати вимірювання	

Прилади MS дозволяють користувачеві вимірювати параметри модельованої схеми, навіть якщо він не знайомий з основами мови моделювання SPICE. Якщо користувач змінює настройки приладу, тут же автоматично змінюються і параметри моделювання.

При проведенні моделювання показання приладів постійно змінюються. В одній і тій же схемі може бути кілька екземплярів приладу. Атрибути настройки приладу і відповідні цим налаштуванням параметри моделювання можуть бути збережені в файлі конфігурації. Отримані при моделюванні дані при використанні вбудованих приладів можуть бути оброблені постпроцесором і показані у вікні Grapher View. Зовнішній вигляд (розміри) інструментальної панелі приладу можуть бути змінені відповідно до необхідного дозволом екрану і способом відображення даних. Дані, отримані в результаті аналізу, можуть бути збережені у форматі файлів TXT, LVM, і TDM.NI Electronics Workbench Group має партнерські зв'язки з представниками провідних фірм в області вимірювальної техніки, таких Agilent® і Tektronix®, тому прилади, розміщені на панелі інструментів MS, виглядають і працюють так само, як і реальні фізичне прилади цих виробників. Вбудовані в ППП MS прилади можуть бути сгруповані за шістьма категоріями (табл. 2.2-2.7).

Таблиця 2.2 Інструменти для аналізу напруги і струмів

Тип прилада	Функціональні можливості	Іконка
1	2	3
Функціональні генератори (Function generator)	Генерування синусоїдальних, трапецеїдальних і імпульсних сигналів. Установка частоти, шпаруватості, амплітуди сигналу	
Мультиметр (Multimeter)	Вимірювання постійного і змінного струму, напруги і втрат.	
2-канальний осцилограф (Oscilloscope)	Вимірювання сигналу в двох каналах. Масштабування Y і X осей. Зсув по Y осі. Синхронізація	
4-канальний осцилограф (4 channel scilloscope)	Вимірювання сигналу в чотирьох каналах. Масштабування Y і X осей. Зсув по Y осі. Синхронізація	
Ваттметр (Wattmeter)	Вимірювання потужності сигналу	
Вимірювач BAX (IV-analysis)	Досліджуються діоди, біполярні PNP і NPN-транзистори (BJT). Канальні транзистори (PMOS), (NMOS) і польові КМОП структури (CMOS)	
Лічильники (Frequency counter)	Вимірюються частота, період, фронти імпульсів, АЧХ, фазові зрушення. Підтримується частота вимірювань понад 10 ГГц, синхронізація, розв'язка по постійному струму	
Будівник графіка Боде (Bode plotter).	Досліджуються частотна характеристика, фазові зрушення. Підтримується частота вимірювань понад 10 ГГц	
Вимірювач частотних спотворень (Distortion analyzer)	Вимірюються інтермодуляційні спотворення, сумарний коефіцієнт гармонійних спотворень (коефіцієнт гармонік)	

Таблиця 2.3- Логічні інструменти

Тип прилада	Функціональні можливості	Іконка
Логічний аналізатор (Logic analyzer)	Вимірюються 16 каналів, історія вимірювань. Підтримується синхронізація. Зовнішня / внутрішня опорна частота	
Логічний конвертер (Logic converter)	Цифрові схеми, побудовані за таблицями истинності і логічним виразами. Таблиці істинності для цифрових схем. Логічні вирази для цифрових схем. Реалізуються цикли, оновлення кроку, скидання. HEX, DEC, Boolean, ASCII-коди	
Генератор слів (Word generator)	Реалізуються HEX, DEC, Boolean, ASCII уявлення даних, синхронізація, тимчасова селекція. Режими: цикли, оновлення кроку, скидання	

Таблиця 2.4 - Прилади радіочастотного діапазону

Тип прилада	Функціональні можливості	Іконка
Аналізатор спектра (Spectrum analyzer)	Вимірюються спектр, компоненти спектра(Потужність, частота),	
Прилад для аналізу електричних ланцюгів в узагальненому вигляді (Network analyzer)	Побудова по цифровій схемі таблиці істинності або логічного виразу. Зворотне перетворення таблиці істинності або логічного вираження в цифрову схему	

Таблиця 2.5 Інструменти, що моделюють вимірювальні прилади фірм-виробників вимірювальних пристроїв

Тип прилада	Функціональні можливості	Іконка
Генератор Agilent (Agilent function generator)	Тип генератора 33120A. Моделювання реального приладу	
Мультиметр DMM Agilent (Agilent multimeter)	Тип генератора 34401A. Моделювання реального приладу	
Осцилограф Agilent (Agilent oscilloscope)	Тип осцилографа 54622D. Моделювання реального приладу	
Осцилограф Tektronix (Tektronix oscilloscope)	Тип осцилографа TDS 2024. Моделювання реального приладу	

Таблиця - 2.6 Вимірювальні пробники

Тип прилада	Функціональні можливості	Іконка
Пробник	Вимірювання струму, напруги і частоти відносно землі	
Пробник	Вимірювання струму, напруги і частоти щодо іншого пробника	
Пробник	Імітація поведінки струмових вимірників (струмових кліщів)	

Таблиця 2.7 - Інструменти, що базуються на ВП LabVIEW

Тип прилада	Функціональні можливості	Іконка
Мікрофон	Підключення до звукової плати комп'ютера. Запис звуку	
Динамік	Підключення до звукової плати комп'ютера	
Аналізатор сигналу	Аналіз сигналу в тимчасовій області. Спектр потужності	
Генератор сигналу	Гармонійний, імпульсний, пилоподібний, трикутний сигнали	

2.1.1.2 Організація бази даних MS

Елементи схеми вибираються з бази даних і розміщуються на робочому полі трьома способами:

- 1) Через Головне меню (Place - Component ...);
- 2) Через контекстне меню робочої області (Place Component ...);
- 3) Через панель компонентів (рис. 2.11).



Рис. 2.11 Панель компонентів

Є такі групи елементів:



Група Sources - джерела енергії і сигналів

Power sources - джерела живлення (AC - джерело живлення змінного струму, DC - джерело живлення постійного струму, DGND - цифрова земля, GROUND - аналогова земля, Vcc, Vdd - позитивне напруження живлення, Vee, Vss - негативна напруга, земля).

Аналогове заземлення використовується у всіх процесах моделювання за винятком моделювання цифрових пристроїв в реальному часі (в цьому режимі, крім затримки в логічному елементі, моделюється час фронту і час спаду, вихідна напруга має згладжені фронти, а сам процес моделювання потребує більшого часу).

Signal voltage sources - сигнальні джерела напруги. Signal current sources - сигнальні джерела струму.

Controlled voltage sources - регульовані джерела напруги. Controlled current sources - регульовані джерела струму.

Control function blocks - функціональні блоки управління.



Група Basic - група з базовими елементами

Rpack - резистивна збірка.

Switch - перемикачі (ключі, push button - кнопка, блоки з 2-10 перемикачами

Transformer - трансформатор.

Non linear transformer - нелінійний трансформатор.

Relay - реле.

Connectors - з'єднувачі, роз'єми. Sockets - сокети.

Resistor - резистори.

Capacitor - конденсатори.

Inductor - котушки індуктивності.

Cap electrolit - електролітичні конденсатори. Variable capacitor - змінні конденсатори.

Variable inductor - змінні котушки індуктивності. Potentiometer - потенціометри.



Група Diodes - діоди

Diode - діоди.

Zener - стабілітрони (діоди Зенера). Led - світлодіоди.

FWB - діодні мости.

Schootky diode - діоди Шотткі.

Scr - тиристори тріодних, замикаються в зворотному напрямку з керуванням по катоду.

Diac - діоди двонаправлені.

Triac - тиристори і сіммістори (одно- і двонаправлені).

Varactor - варикапи (ємнісні діоди).

Pin diod - pin діоди (містять область власної провідності між сильнолегованих областями).



Група Transistors - транзистори BJT NPN - біполярні транзистори типу NPN.

BJT PNP - біполярні транзистори типу PNP.

BJT ARRAY - мікроскладення транзисторів.

DARLINGTON NPN - транзистори Дарлінгтона NPN (складові транзистори).

DARLINGTON PNP - транзистори Дарлінгтона PNP.

DARLINGTON ARRAY - масив транзисторів Дарлінгтона.

MOS 3TDN - транзистор каналний з вбудованим каналом N-типу.

MOS 3TEN - транзистор каналний з ізольованим затвором про- чення типу з N-каналом, з внутрішнім з'єднанням витoku і підкладки (індукований канал).

MOS 3TEP - транзистор каналний з індукованим каналом P-типу.

JFET N - транзистор польовий з провідністю типу N.

JFET P - транзистор польовий з провідністю типу P.

POWER MOS N - потужний каналний транзистор з каналом N-типу.

POWER MOS P - потужний каналний транзистор з каналом P-типу.

POWER MOS COM - потужний каналний транзистор (комплементарна технологія).

UJT - тиристор тріодний, замикається в зворотному напрямку з керуванням по аноду.

THERMAL MODELS - температурні моделі.



Група Analog - аналогові компоненти

Opamp - операційні підсилювачі. Comparator - компаратори.

Wideband amps - широкосмугові підсилювачі.

Special function - компоненти, реалізовані спеціальні функції. Група TTL - елементи транзисторних-транзисторної логіки 74-серії



Група CMOS - комплементарна МОП-структура

(Компліментарні транзистори)



Група MCU

Мікроконтролери - МК (805х, PIC, ін.

Мікросхеми пам'яті - RAM, ROM (HM-65642-883 (8k x 8), HM61116A120 (2k x 8)), ПЗУ (27C128-12L (16k x 8), 27C256-15L (32K x 8)), ППЗУ (27C64Q350-883)).



Група Advanced peripherals

Вдосконалені периферійні пристрої, такі як віртуальна цифрова клавіатура (4x4, 4x5), LCD-дисплей, світлофор.



Група Misc digital - різні цифрові мікросхеми

DSP - пристрої DSP (цифрові сигнальні процесори).

FPGA - пристрої FPGA (програмована користувачем вентиляційна матриця).

PLD - програмовані логічні пристрої.

CPLD - комплементарні програмовані логічні схеми. Microcontrollers /

Microprocessors - мікроконтролер / мікропроцесор.

Memory - мікросхеми пам'яті.

Line driver - лінійний формувач.

Line transceiver / receiver - лінійний приймач / приймач.



Група Mixed - пристрої змішаного сигналу

Analog switch - аналогові перемикачі.

Analog switch IC - IC аналогового перемикача.

Timer - таймер.

Multivibrators - мультивібратори.

ADC, DAC - АЦП (ADS8364Y, AD16), ЦАП (DAC7643_FP32).



Група Indicators - індикатори

Voltmeter - вольтметри.

Ammeter - амперметри.

Probe - пробники.

Buzzer - автоматичні переривники.

Lamp - лампи.

Hex display - дисплеї (світлоіндикатор: 15-сегментні, 7-сегментні, з загальними катодом (ОК) або анодом (ОА), з десятковою крапкою, без точки, дисплеї з «+» або «-», 7-сегментні дисплеї з двома цифрами (з десятковою крапкою, з ОА або з ОК).

Bargraph - столбцовая діаграма.



Група Power - компоненти, що відносяться до джерел живлення і пов'язані з ними:

Fuse - плавкі запобіжники.

Voltage reference - джерела опорного напруги.

Voltage regulator - потенціометри.

Voltage suppressor - обмежувальні діоди.

Power supply controller - контролери джерел живлення

Misc power - інші джерела живлення.

PWM controller - широтно-імпульсний модулятор.

MISC

Група Misc - інше

Optocoupler - оптрони.

Crystal - кварцові резонатори.

Vacuum tube - електронні лампи.

Boost converter - підсилювачі-перетворювачі.

Lossy transmission line - лінія передачі з втратами.

Lossless line type1 - лінія без втрат, тип 1.

Lossless line type2 - лінія без втрат, тип 2.

Filters - фільтри.

Mosfet driver - драйвер польового транзистора.

Net-мережа зв'язку.



Група RF - радіочастотні пристрої

RF capacitor - радіочастотні конденсатори.

RF inductor - радіочастотні котушки індуктивності.

RF BJT NPN - радіочастотні біполярні транзистори типу NPN.

RF BJT PNP - радіочастотні біполярні транзистори типу PNP.

RF MOS 3TDN- радіочастотні польові транзистори: вбудований канал N-типу

Tunnel diode - радіочастотні тунельні діоди.

Strip line - полоскові лінії.



Група Electro Mechanical - електромеханічні пристрої

Sensing switches - сенсорні перемикачі. Momentary switches - миттєві перемикачі. Supplementary contacts - додаткові контакти. Timed contacts - синхронізовані контакти.

Coils relays - реле.

Protection devices - елементи захисту (запобіжники). Output devices - пристрої виведення.

Всі розглянуті вище компоненти є реальними (промисловими) і мають певні, незмінні значення параметрів. У схемі за замовчуванням вони позначаються синім кольором. У базі даних є також і віртуальні компоненти, які мають у своїй назві приставку VIRTUAL (неявно відображаються на схемі чорним кольором). Віртуальні компоненти необхідні для дослідження, так як користувач може призначити їм довільні значення параметрів.

2.1.2 Створення проекту і програмного файлу

При відкритті програми MS автоматично створюється проект схеми під назвою Circuit1. Для зміни імені схеми необхідно зберегти його через пункт меню File-Save As, бажано використовувати в директорії англійські букви. Для створення проекту схеми (якщо необхідно - програмного файлу) слід вибрати її елементи (резистори, конденсатори, індуктивності, МК, ІС, ін.) Для її реалізації (при необхідності-програмування МК). У MS, наприклад, програмуються МК, розташовані в БД в групі MCU, при цьому МК вибираються двома способами:

- 1) через пункт меню Place - Component - MCU - 805x - 8051;
- 2) через панель компонентів: - 805x - 8051.

Встановлюємо МК в робочій області, з'являється спливаюче вікно «Майстер зі створення програмного файлу», яке пропонує виконати три кроки для створення проекту і програмного файлу.

Крок 1. Визначення робочого простору (рис. 2.12)

У першому рядку спливаючого вікна вказується шлях робочого простору для обраного МК. Використовуючи кнопку «Browse», можна змінити шлях робочого простору, запропонований програмою. У наступному рядку вікна пропонується ввести ім'я робочого простору.

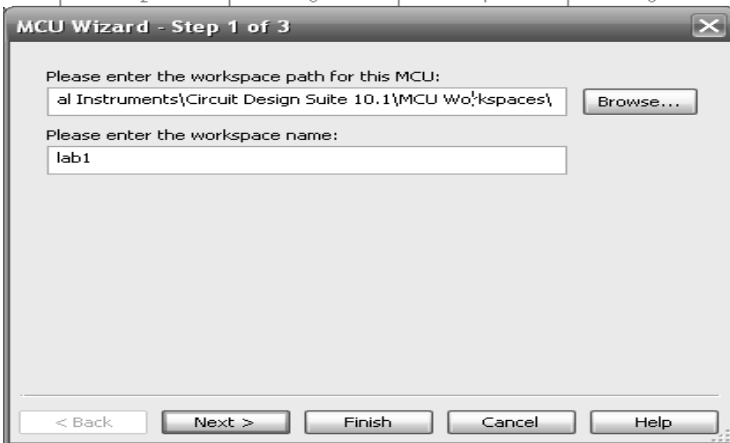


Рисунок 2.12 - Вікно завдання робочого простору

Крок 2. Створення проекту для мікроконтролера (рис. 2.13)

У цьому вікні пропонується встановити наступні настройки для майбутнього проекту:

- 1) тип проекту: Standard або Use External Hex File;
- 2) мова програмування: C, Assembly;
- 3) компілятор;
- 4) ім'я проекту, наприклад Circuit1.

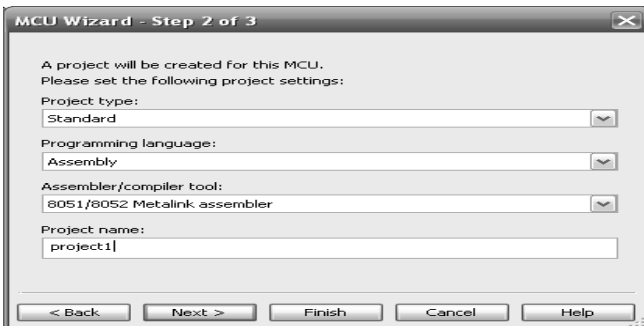


Рисунок 2.13 - Створення проекту для мікроконтролера

Крок 3. Створення програмного файлу (рис. 2.14)

У вікні пропонується створити або порожній проект (без програмного файлу), або додати вихідний програмний файл, вказавши його ім'я. Робота з Майстром закінчується натисканням кнопки Finish.



Рисунок 2.14 - Створення програмного файлу

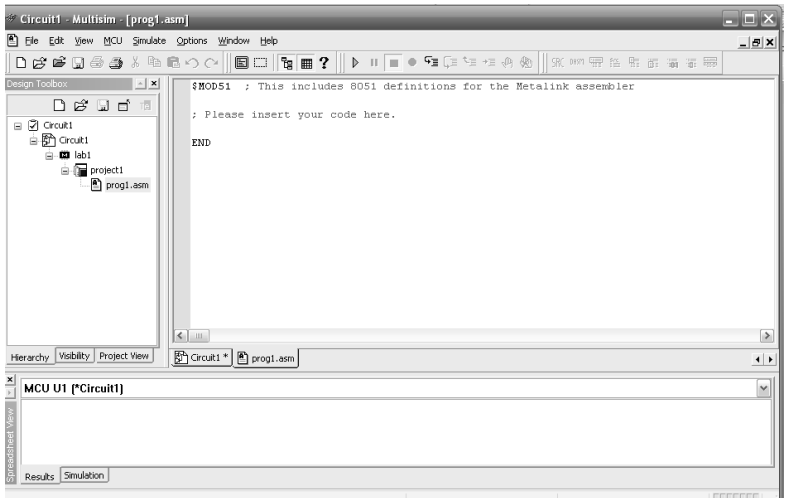


Рисунок 2.15 - Вікно програмного файлу

У вікні Design Toolbox (рис. 2.15) на закладці Hierarchy можливо переглянути структуру створеного проекту. Відкриття програмного файлу здійснюється подвійним клацанням ЛКМ по його назві у вікні Design Toolbox.

2.1.3 Виконання завдання по створенню електронної схеми

Згідно варіанту завдання створити схему з використанням її складових і зазначених елементів. Виконати з'єднання елементів (довільно), ввести позиційні позначення і нумерацію ланцюгів. При виконанні завдання використовувати такі нормативні документи: ДСТУ (ГОСТ) 2.702-75 «Правила виконання електричних схем»; ДСТУ (ГОСТ) 2.710-81 «Правила виконання схем». З'єднання елементів (наприклад, з резистором, конденсатором, індуктивністю, МК, ІС, ін.) Позначити зеленим (або на вибір) кольором, порівняти реальний і віртуальний компоненти, зазначені в завданні (пояснити, в чому їх відмінність). Елементи схеми вибрати самостійно з бази даних (БД) MS.

Розміщення компонентів здійснюється через пункт меню Place або гарячу клавішу Ctrl-W, які викликають звернення до провідника компонентів (рис. 2.16).

У провіднику компонентів відображається поточна база даних зі схемними елементами. У MS вони організовані в групи (groups) і сімейства (families). Також в провіднику компонентів відображається опис компонента (поле Function), модель і виробник.

Для пошуку потрібного елемента схеми необхідно набрати назву компонента в поле Component, і провідник автоматично підбере відповідні елементи. Також необхідний елемент схеми можна знайти у відповідній йому групі (Group). За допомогою підменю

«Пошук» (Search) відкривається розширений пошук елементів.

Символ зірочки («*») в назві компонента замінює будь-який набір символів. Наприклад, серед результатів запиту на елемент "74LS * N" будуть інтегральні схеми (IC) «74LS01N» і «74LS183N».

При роботі з компонентами слід мати на увазі, що будь-якого компонента відповідає певна модель в БД, що враховує різні фізичні характеристики компонента. Наприклад, операційний підсилювач LM358M має 5 зовнішніх контактів, але в цій моделі БД з них використовується тільки 3, контакти живлення не задіяні (неявно задані). Інформація про особливості використовуваної моделі елемента знаходиться в полі провідника «Виробник / ідентифікатор» (Model Manuf. \ ID), для цього необхідно виділити ЛКМ «Модель» (Model).

Подвійне клацання ЛКМ по компоненту або натискання кнопки ОК у вікні провідника компонентів прикріпить його до курсора. Після цього компонент поміщається на схему в бажаному місці робочого простору за допомогою ЛКМ. До установки або після установки елемента в схему його можна повернути за / проти годинникової стрілки за допомогою гарячої клавіші Ctrl-R / Ctrl-R-Shift або вибрати в контекстному меню пункт «90 Clockwise» або «90 Counter CW» відповідно.

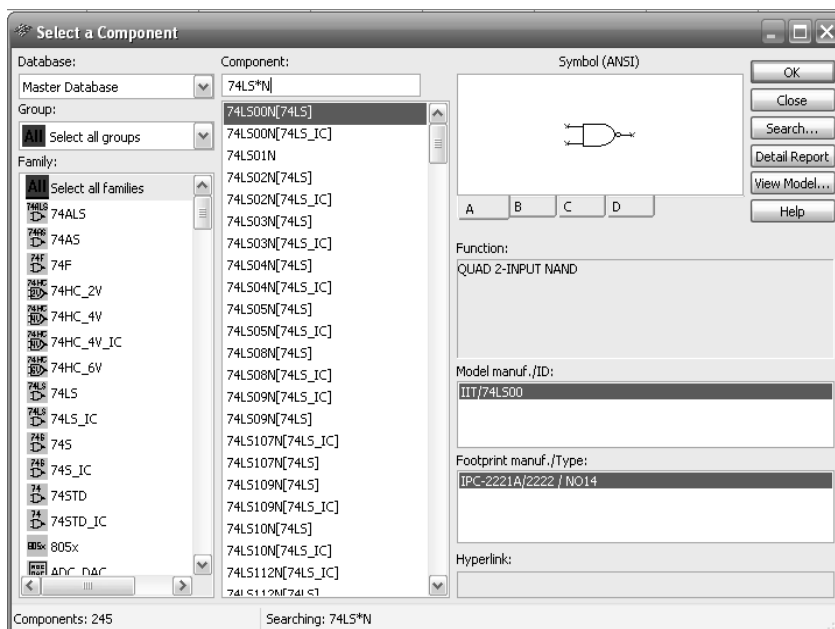


Рисунок 2.16 - Провідник компонентів

Щоб вибрати компонент на схемі, необхідно клацнути по ньому ЛК миші. Для одночасного вибору декількох компонентів потрібно притиснути ЛК миші і переміщати її, маюючи прямокутник навколо потрібних компонентів. Обрані на схемі компоненти виділяються пунктирною лінією. Виділення окремих атрибутів компонента, наприклад значення або мітки, здійснюється одинарним клацанням по відповідному атрибуту. Клавіша Shift дозволяє додавати або знімати виділення з декількох компонентів. Обрані компоненти, МК, ІС з БД можна замінити на інші, подібні компоненти за допомогою їх контекстного меню, пункту Replace Components, при цьому відкривається вікно провідника компонентів. Після заміни елемента MS відновить з'єднання з іншими елементами схеми (необхідна повторна перевірка правильності приєднання).

2.1.3.1 Порядок виконання ЛР, створення схемного проекту

Відкриваємо і зберігаємо новий схемний проект, наприклад Circuit2. Розміщуємо на робочому полі потрібні елементи, перемикачі, ІС землю і живлення (Place - Component - Sources - POWER_SOURCES), ін..

Після вибору компонентів з БД і розміщення їх на схемі, необхідно з'єднати компоненти між собою. У програмі Multisim дія миші на схемі залежить від положення курсора. Зовнішній вид курсора змінюється в

залежності від того, на який об'єкт він наведений (рис.2.17) Для того щоб провести з'єднувальний провід між елементами, необхідно клікнути ЛКМ по виводу одного елемента, потім, не відпускаючи кнопку, провести з'єднання до виводу іншого елемента. Після появи з'єднувального провідника між елементами MS автоматично присвоїть йому номер. Нумерація ліній збільшується послідовно, починаючи з 1.

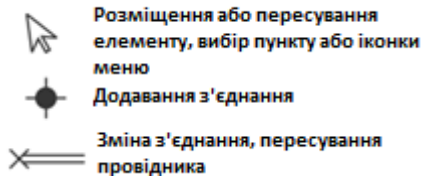


Рисунок 2.17 - Види курсора

Заземляючі дроти завжди мають номер 0, це вимога пов'язана з роботою прихованого емулятора SPICE. Щоб змінити нумерацію з'єднання або привласнити йому логічне ім'я, необхідно двічі клікнути по сполучному провіднику

2.1.4 Особливості інструментарію для створення спільних проектів електронних схем в MS і LV

2.1.4.1 Підготовка схеми в MS для її зв'язку з LV

Найбільш простими способами здійснити зв'язок MS12 з LV (отримавши модель ВКС з візуальними органами ПБВ, управління, і контролю) є способи 1 та 2 (см.п.2.1). Нижче наведено короткий опис для реалізації зв'язку MS12 (13) і LV за способом 3.

Налаштування входів. Можливо міняти значення джерела сигналу (живлення). Для цього необхідно в схемі встановити це джерело. Залежно від потреби, джерел може бути кілька, і потрібно встановити їх там, де знадобиться їх контролювання.

Установка джерела:

1. Виберіть Place → Component
2. Database встановіть Master Database
3. Group встановіть Sources
4. В Family виберіть POWER_SOURCES
5. Тут вибираємо AC_POWER або DC_POWER

Налаштування виходів. Для того щоб «движок» зрозумів з якої гілки схеми потрібно брати сигнал необхідно встановити «Пробник» (Probe). Кожен «Пробник» в схемі повинен мати своє індивідуальне ім'я.

Установка «Пробника»:

1. Виберіть Simulate → Instruments → Measurement Probe
2. Підключіть «Пробник» до цікавить гілки
3. Двічі клацніть по «пробники»
4. Виберіть вкладку «Monitor»
5. У розділі RedDes введіть ім'я «Пробника». Для гілки, яке є виходом, використовують ім'я «output».

2.1.4.2 Інструментарій MS Automation with the LV MS Connectivity Toolkit (beta)

MS Automation with the LV MS Connectivity Toolkit (beta) дозволяє **пов'язувати схему в MS12 (13) з LV**, наприклад, через MS Automation API. Його інструменти, такі як **відкриття, закриття, перегляд схеми, запуск, пауза і т.п.**, представлені в LV як віртуальні пристрої (ВП). Дане доповнення можна знайти за адресою ni.com/labs.

Після установки MS Automation with the LV MS Connectivity Toolkit (beta) нову палітру ВП можна знайти в **Меню**; Functions → Connectivity → MS і в Functions → Addons → MS (рис.2.17).

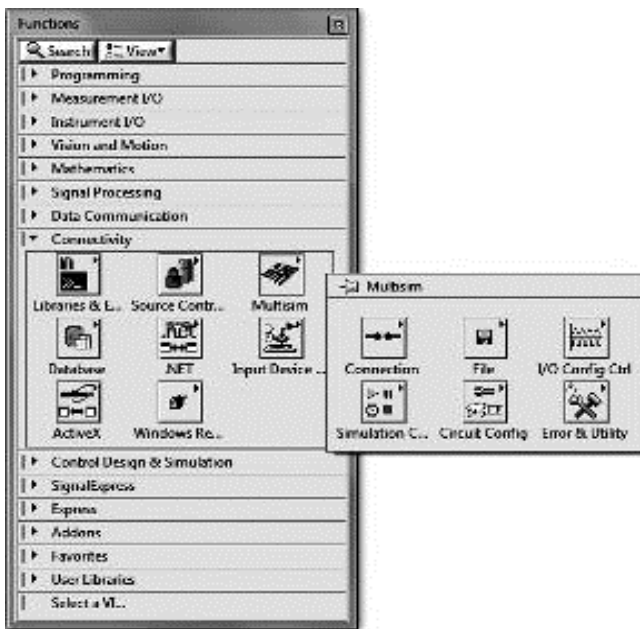
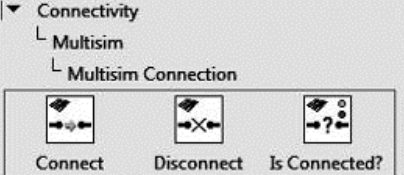
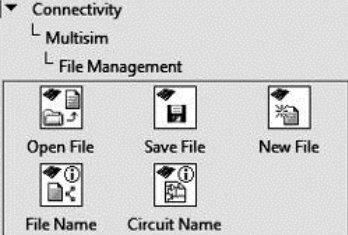
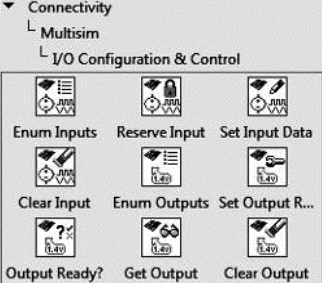
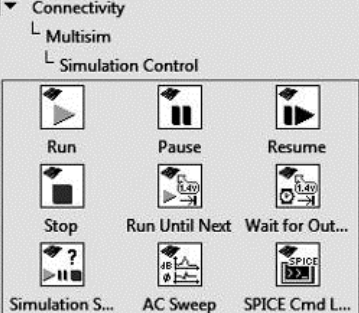
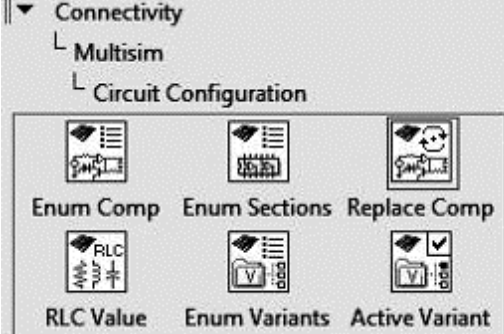
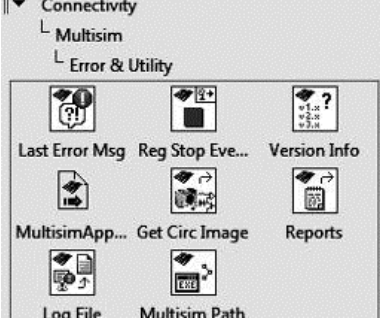


Рисунок 2.17 - Палітра Automation API для MS12 (13)

Нижче наведена характеристика функцій, що підключаються

Підключені функції	Опис
	Ці ВП дозволяють підключати і відключати двигок MS.
	Ці ВП дозволяють відкрити, зберегти, створити новий порожній файл MS, побачити ім'я файлу і схеми..
	Ці ВП введення / виводу дозволяють отримати, встановити і видалити елементи введення / виведення при моделюванні.
	Ці ВП дозволяють контролювати процес моделювання: запуск, пауза, зупинка, відновлення і т.п. Також виробляти частотні характеристики і контролювати процес за допомогою команд командного рядка (SPICE).

	Ці ВП дозволяють змінювати значення компонентів схеми.
	Тут частина ВП дозволяють обробляти помилки і частина ВП - утиліти, які дозволяють LV узгоджувати з automation API.

Поліморфні функції. Для полегшення багато з функцій LWe поліморфними, тобто один блок має кілька функцій. Наприклад, якщо розглянемо функцію Log File (Файл Журналу), то можемо побачити що вона може мати одну з трьох підфункцій:

- Get the Log File Path (Отримати шлях до Файлу Журналу);
- Set the Log File Path (Встановити шлях до Файлу Журналу);
- Disable the Log File Path (Відключити шлях до Файлу Журналу).

Наявні поліморфні функції і пов'язані з ними підфункції:

Функція	Підфункція
Enum Inputs	• 1D String Array • Variant
Enum Outputs	• 1D String Array • Variant
Set Input Data	• Raw • Sampled
Clear Input	• Input

	• All Inputs
Clear Output	• Output • All Outputs
AC Sweep	• String • 1D String Array • Variant
Save	• Save • Save As...
Enum Components	• 1D String Array • Variant
Active Variant	• Get • Set
RLC Value	• Get • Set
Last Error Message	• From Application • From Circuit
Report	• BOM (Real, Virtual, Txt, Csv) • Netlist (Probes, No Probes, Txt, Csv)
Log File	• Get • Set • Disable
MS Path	• Get • Set

2.1.4.3 Модель макета RLC схеми «вхідні ланцюг пасивного радіоприймаючого контуру» - «схема з внутріємкостними зв'язками: антенний вхід і вихід з навантаженням»

Використовуючи вищевказані інструкції зберемо «схему з внутріємкостними зв'язками: антенний вхід і вихід з навантаженням» (табл.2.1, далі за варіантами завдань 1-12 ЛР) - ланцюг пасивного радіоприймаючого контуру (рис.2.18), наприклад, схеми виділення сигналу Wi-Fi (частота 1,2 ... м) бездротової мережі типу LAN для ВКС типу Smart House (реалізація блоку входу в модулі пристроїв прийому, ін.). Основою схеми рис.2.18 є коливальний контур, утворений C1 і L1; також на «проходження» вхідний несучої частоти FH від входу (input) до виходу (output) впливають і інші параметри схеми (R1, R2, R3, C2, C3, C4): див. варіанти 1-12 завдання дослідження (п .2.2).

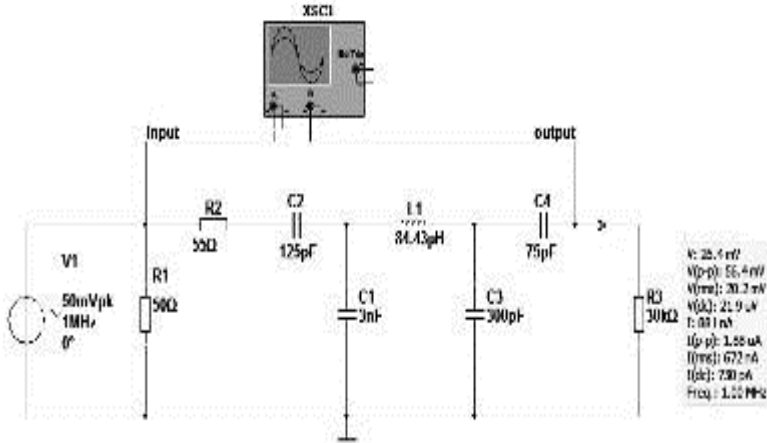


Рисунок 2.18 - Вигляд зібраної схеми в MS

На вхід (**гілка «input»**) реального макета подається синусоїдальний сигнал несучої частоти F з ВЧ генератора амплітудою 50 мВ і частотою, змінною в межах 600 ... 1400 кГц (20% від значення F). У схемі моделі, зібраної в MS, встановимо джерело змінної напруги - DC_POWER. Далі, в ППП LV, значення джерела змінюється по частоті F в заданому діапазоні, але для правильної синхронізації програмних засобів, нам потрібно встановити будь-яке значення. Тому вибрано середнє значення частоти 1 МГц. Резистор R3 вихідний ланцюга є навантаженням схеми, тому її визначаємо як **гілка «output»**, де встановлюємо «пробник». Змінимо ім'я пробника з «Probe1» на «output1». На цьому підготовка в середовищі MS закінчена. Переходимо до складання моделі дослідження в LV. Спрощено зв'язок між MS і LV здійснюється як показано на рис.2.19 (для схеми рис.2.18).



Рисунок 2.19 - Обмін даними між MS і LV моделлю

3. Визначаємо вхідний джерело вхідного сигналу;
4. Включаємо блок, за допомогою якого можна подавати сигнал з LV в досліджувану схему в MS, через джерело V1;
5. Генератор, який подає сигнал на блок синхронізації з MS;
6. Встановлюємо значення ферроваріометра L1;
7. Визначаємо місце розташування «Пробника»;
8. Визначаємо які дані потрібно отримати з «Пробника»;
9. Запускаємо моделювання в MS (до речі, в цей момент починає працювати MS в фоновому режимі);
10. Отримуємо дані після моделювання;
11. Дивимось, чи змінилося значення ферроваріометра L1;
12. Виводимо необхідні дані для контролю сигналу (на осцилограф, мультиметр і \ або вольтметр);
13. Зупиняємо моделювання;
14. Відключаємо MS і LV MS Connectivity Toolkit (beta);
15. Відключаємо осцилограф.

Таблиця 2.8 - Дані параметрів вихідної схеми «вхідні ланцюг пасивного радіоприймального контуру» по рис.2.18.

Пара-метр	Амплі-туда	Резис-тор	Резис-тор	Резис-тор	Емність	Емність	Емність	Емність	Індукти-вність
F _н , МГц	U, mV	R1, Ω	R2, Ω	R3, кΩ	C1*, nF	C2, pF	C3 pF	C4, pF	L1*, pH
1,0 +/- 20%	50	50	5,5	30	3	125	300	75	84,43

* змінювані параметри коливального контуру LC

Далі переходимо у вікно Front Panel в LV, встановимо і розмістимо необхідні інструменти керування і контролю (рис.2.21). Отримана віртуальна модель схеми відповідає досліджуваній схемі рис.2.18 і можливого реального / віртуального лабораторного її макету.

2.1.5 Хід виконання ЛР і завдання за варіантами 1-12

Для виконання ЛР№1 необхідно виконати наступне:

- завантажити ППП MS і LV, реалізувати схему рис.2.18 і її зв'язки між елементами в MS (за параметрами табл.2.18), підключити вхідні і вихідні прилади (генератор, осцилограф, заземлення, ін.) та отримати стійкий вихідний сигнал на екрані віртуального осцилографа;
- виконати пп.2.1.4.1;
- проаналізувати вихідну схему **коливального контуру** рис.2.18 (MS) і створити її варіант / аналог у вигляді схеми типу рис.2.21 (LV), для чого використовувати один із способів 1...4 (встановити ВП і перевірити їх роботу);

- отримати діаграми моделювання (варіанти 1...12, крок - 6...8 точок **зміни параметрів** від зазначених табл.2.8 за варіантами). Для цього використовувати **незмінні** (для U, R1, R2, R3, C2, C3, C4) і **змінні** параметри C1 (підгрупа 1) і L1 (підгрупа 2) в зазначених межах. Порівняти значення вихідного коливального контуру (табл.2.8) з параметрами, заданими варіантом (змінною значень параметрів C1, L1):
- при якому амплітуда колювання LC- контура **мінімальна, максимальна**;
- вказати напрямок зміни параметру (C1 і / або L1) для зменшення, збільшення вихідної напруги Uвих. контура;
- порівняти значення Uвх., Uвих., визначити % загасання (початкові умови).

№ варіантів і зміна параметрів схеми по табл.2.8

1. + 25% группа1 і -25% група 2; частота F 1,0МГц;
2. -25% группа1 і + 50% група 2; частота F 1,05МГц;
3. + 50% группа1 і -25% група 2; частота F 1,1МГц;
4. -50% группа1 і + 50% група 2; частота F 1,15МГц;
5. + 75% группа1 і + 25% група 2; частота F 1,2МГц;
6. -75% группа1 і -25% група 2; частота F 1,25МГц;
7. + 75% группа1 і -50% група 2; частота F 0,95МГц;
8. -10,0% группа1 і + 50% група 2; частота F 0,9МГц;
9. + 10,0% группа1 і -50% група 2; частота F 0,85МГц;
10. -12,5% группа1 і + 50% група 2; частота F 0,8МГц;
11. + 15,0% группа1 і -75% група 2; частота F 0,75МГц;
12. -15,0% группа1 і + 75% група 2; частота F 0,70МГц.

2.1.6 Загальні зауваження до порядку проведення і змісту звітів

На занятті студент повинен отримати допуск до виконання ЛР, вивчити тему ЛР, при необхідності налагодити схему і / або програму на ПК, прокоментувати викладачеві роботу її частин (в т.ч. покроково - в звіті) та обґрунтувати правильність функціонування схеми відповідно до варіанта завдання. Після демонстрації функціонування схеми в ППП MS (з елементами LV) ЛР вважається виконаною, викладач контролює її електронний вигляд і підписує чернетку. До наступного заняття студент може оформити звіт про виконану ЛР (на основі чернетки) і захистити його. Позитивна оцінка і підпис звіту означає, що ЛР здана. Рекомендується звіти з ЛР зберігати до кінця навчання по ВКС. На екзамені з ВКС студентам дозволено користуватися своїми звітами при відповідях на екзаменаційні питання білетів і при вирішенні задач / завдань.

Звіти по курсу ЛР1-6 (ч.1, ч.2) повинні мати такі розділи: Теоретична частина. Варіант завдання. Схеми підключення елементів схеми і / або МК до входів і / або виходам, які реєструють їх стану: індикатори, осцилографи, аналізатори, задатчики кодів, генератор, ін., в т.ч. прибори LV. Перевірена

електронна схема і / або програмний код з коментарями до кожної операції. Теоретичний вибір способу (1...4) переходу / перетворення схеми з MS в LV, отримання результатів на моделі (за варіантами), їх аналіз. Діаграми режимів, їх параметри і значення. Результати розрахунків. Висновки по ЛР.

2.1.7 Зміст звіту

1. Найменування і мета роботи, теоретична частина;
2. Опис схеми, призначення елементів, їх функція, осцилографа / аналізатора (використаного в експерименті) з його короткими характеристиками.
3. Копія схемного файлу в MS, LV під час моделювання із зазначенням позиційних позначень елементів, їх типу, номіналу, копія програмного файлу (асемблер і / або на C+) з обов'язковим коментарем.
4. Отримані результати і висновки по ЛР.

2.1.8 Контрольні питання

1. Навіщо використовують коливальний контур, його склад, параметри?
2. Чим відрізняються промислові компоненти, ІС від віртуальних?
3. Як можна змінювати параметри контура без зміни його частоти?
4. Які ВП використовуються для аналізу схем по струму, напрузі, їх комбінації? Яким чином редагують, перевіряють ланцюга схем в MS, LV?
5. У чому відмінність представлених оболонок САПР MS, LV від інших?
6. Вкажіть основні способи «зв'язування» MS з LV?
7. Які програмовані пристрої представлені в MS і LV, які мови програмування для них використовуються?
8. Які типи ІС (МК, ALU, DC, MX, ін.) є в БМ демоверсії MS?
9. Які типи віртуальних приладів (ВП) використано в MS і LV?
10. Яка особливість застосування ВП MS, LV, їх основна відмінність?

3. Лабораторна робота №2. Вивчення Принципів и особливо роботи цифроаналогових перетворювачів

Мета роботи: Ознайомлення з особливостями роботи інтегрального цифроаналогового перетворювача (ЦАП)

3.1 Загальні відомості про цифроаналогові перетворювачі

Цифроаналоговим перетворювачем (ЦАП) називають пристрій, що здійснює автоматичне перетворення входних значень, представлених числовими кодами, в еквівалентні їм значення якої-небудь фізичної величини (напруги, струму, випромінювання, ін.).

Основи класифікації ЦАП представлені на рис. 3.1.

Детальна інформація про ЦАП викладена, наприклад, в [10]. Раз розподіл на послідовні і паралельні пристрої розглядається з точки зору паралельного або послідовного під час перетворення сигналів.

3.1.1 Особливості роботи паралельних ЦАП з підсумовуванням струмів

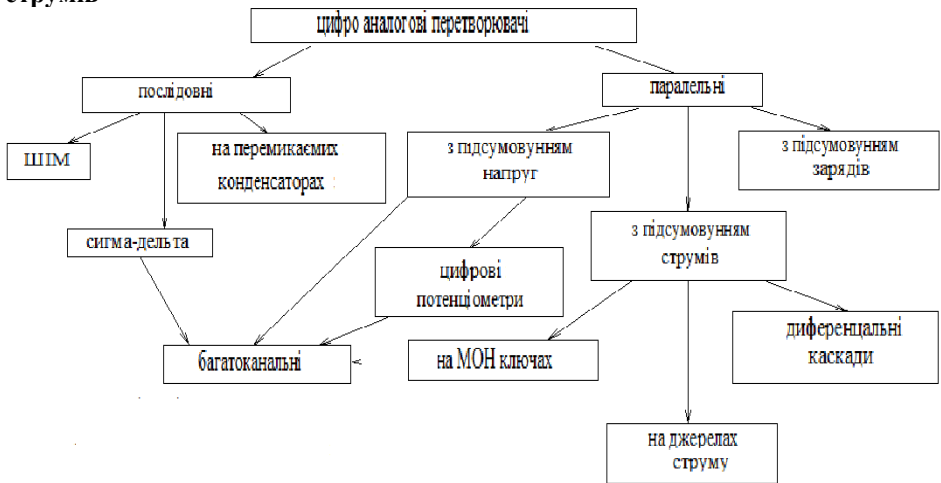


Рисунок 3.1 - Загальна класифікація ЦАП

Існує два широко поширених паралельних способи цифроаналогового перетворення за допомогою підсумовування струмів, показаних на рис. 3.1: з двома номіналами опорів $R-2R$ і з ваговими двійковій-зваженими опорам.

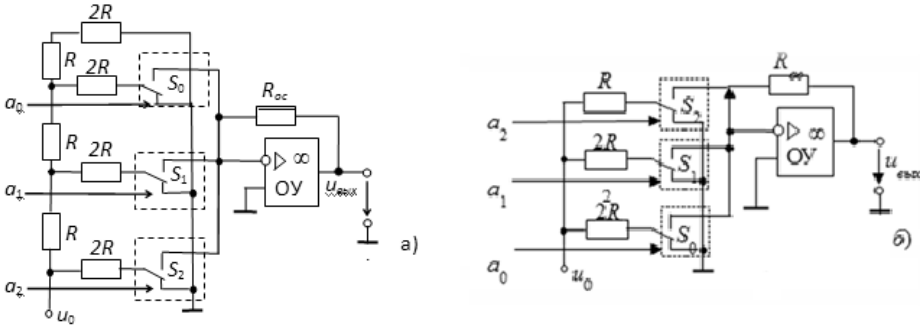


Рисунок 3.2 - Паралельні 3-х розрядні ЦАП з підсумовуванням струмів на резистивних матрицях: з двома номіналами опорів R - $2R$ - а; з ваговими двійково-зваженими опорами - б.

ЦАП з ваговими двійкової-зваженими опорами (рис. 2.34, б) складається з n перемикачів S_i (по одному на кожен розряд), керованих двійковим кодом A_i ; з матриці двійково зважених резисторів з опором $2^{(n-1-i)}R$; джерела опорної напруги U_0 і вихідного операційного підсилювача ОП (ОУ), з допомогою якого підсумовуються струми, що протікають через резистори, для отримання аналогової вихідної напруги $U_{вих..}$.

Кожен i -ий розряд управляє перемикачем S_i , який підключається до джерела опорної напруги u_0 , коли a_i дорівнює 1, або до загальної шини, коли a_i дорівнює 0. Опору резисторів $2^{(n-1-i)}R$ (n - розрядність ЦАП), з'єднаних з ключами, такі, що забезпечують пропорційність в них струму бінарній вазі відповідного розряду вхідного коду. Отже, струм на вході ОП і вихідна напруга ЦАП визначаються як сума:

$$U_{вих..} = \left(\frac{a_{n-1}u_0}{R} + \frac{a_{n-2}u_0}{2R} \cdots + \frac{a_1u_0}{2^{n-1}R} + \frac{a_0u_0}{2^nR} \right) R_{oc} = \sum_{i=0}^{n-1} \frac{a_{n-1-i} u_0 R_{oc}}{2^i R_i} ;$$

Напруга на виході ЦАП пропорційна «вазі» присутнього на входах коду, а максимальне значення має місце, коли всі розряди приймуть значення 1, тобто

$$U_{max} = \left| \frac{(2^n - 1)}{R} R_{oc} \right| u_0$$

і воно завжди менше опорної напруги на крок квантування $U_{oRoz} / (2^n R)$.

Номінали опорів резисторів в молодшому і старшому розрядах відрізняються в $2^n - 1$ раз і повинні бути витримані з високою точністю. Наприклад, для 12-розрядної ЦАП використання в старшому розряді резистора з опором 10 кОм потребує включення в молодший розряд

перетворювача резистора з опором близько 20 Мом (неможливо реалізувати). Широкий набір номіналів резисторів і вимоги їх високої точності, особливо при значному числі розрядів п вхідного коду, створюють труднощі при реалізації ЦАП за допомогою інтегральної технології.

У схемі ЦАП (рис.3.2 а) з **матрицею R-2R** використовують резистори з двома номіналами опорів, причому резистори з опором R включені в кожен розряд, проте в цій схемі збільшуються значення паразитних ємностей.

Принцип роботи представленої схеми заснований на властивості резистивного ділителя R-2R зберігати постійний опір навантаження для джерела опорної напруги при замиканні ключів. Внаслідок цього на виході резистора R, починаючи зі старшого $n-1$ розряду, опорна напруга u_0 послідовно ділиться навпіл, як і струм, що входить в кожен вузол. При цьому напруга на виході Увих. перетворювача з матрицею R-2R дорівнює:

$$U_{\text{вих.}} = -u_0 \frac{R_{oc} (a \cdot 2^{-1} + a \cdot 2^{-2} + \dots + a \cdot 2^{-(n-1)} + a \cdot 2^{-n})}{R_{(n-1; n-2; \dots 0)}} = \sum_{i=0}^{n-1} \frac{a_i R_{oc}}{R_{(n-1; n-2; \dots 0)}} \quad (3.1.1)$$

3.1.2 Основні технічні характеристики

Основними параметрами ЦАП є число розрядів коду, абсолютна роздільна здатність, точність ЦАП, максимальна частота перетворення.

Число розрядів коду n , зазвичай становить величину 8, ..., 12.

Абсолютна роздільна здатність - середнє значення мінімальної зміни сигналу на виході ЦАП, обумовлене збільшенням або зменшенням його коду на одиницю.

Теоретично ЦАП, що перетворює n -розрядні двійкові коди, повинен забезпечити 2^n різних значень вихідного сигналу з роздільною здатністю 1 / ($2^n - 1$). При числі розрядів $n = 8$, кількість незалежних квантів (ступенів) вихідної напруги ЦАП становить $2^8 - 1 = 255$, при $n = 12$, кількість незалежних ступенів дорівнює $2^{12} - 1 = 4095$ і т. д.

Абсолютне значення мінімального кванта напруги визначається як гранично прийнятним числом $2^n - 1$, так і максимальною вихідною напругою ЦАП, також званим напругою шкали або опорною напругою U_0 . Значення абсолютної роздільної здатності ЦАП, часто позначається як значення молодшого розряду (ЗМР). При $n = 8$ і опорній напрузі $U_0 = 5$ В, ЗМР = $U_0 / (2^8 - 1) = 5/255 = 0,0196$ В = 19,6 мВ.

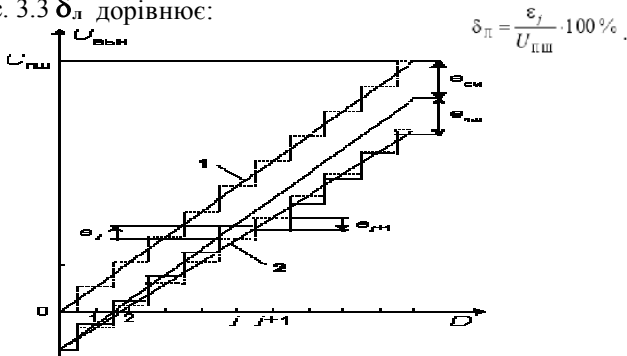
Відмінність реального значення роздільної здатності від теоретично обумовлено похибками і шумами вхідних в ЦАП вузлів.

Точність ЦАП визначається значенням абсолютної погрішності Δa і нелінійністю перетворювача $\delta_{\text{л}}$. Абсолютна похибка Δa характеризується відхиленням максимального значення вихідної напруги U_{max} від

розрахункового, відповідаючого кінцевій точці характеристики ідеального перетворювача, і вимірюється зазвичай в одиницях ЗМР (рис. 2.49, лінія 1).

Нелінійність перетворювача $\delta_{\text{л}}$ характеризує відхилення дійсної характеристики 2 від ідеальної 1 (рис. 2.49), проведеної через центри сходинок або через нуль і точку максимального значення вихідного сигналу.

Нелінійність зазвичай визначається у відносних одиницях, але в довідкових даних наводиться також і в ЗМР. Для характеристики, наведеної на рис. 3.3 $\delta_{\text{л}}$ дорівнює:



1 - ідеальна характеристика; 2 - реальна. $U_{\text{пш}}$ - похибка повної шкали

Рисунок 3.3 - Характеристики перетворення ЦАП

Диференціальна нелінійність $\delta_{\text{дл}}$ - максимальне зміння (з урахуванням знака) відхилення реальної характеристики перетворення $U_{\text{вих}}$ (D) від оптимальної при переході від одного значення вхідного коду до іншого суміжного значенню. Визначається у відносних одиницях або в ЗМР. Для характеристики, приведеної на рис. 3.3

$$\delta_{\text{дл}} = \frac{\epsilon_j + \epsilon_{j+1}}{U_{\text{пш}}} \cdot 100\%$$

3 динамічних параметрів найбільш важливим параметром є **максимальна частота перетворення f_{max}** (десятки і сотні кілогерц) - найбільше їх значення, при якій параметри ЦАП відповідають заданим.

Перетворення сигналу в ЦАП часто супроводжується специфічними перехідними імпульсами в вихідному сигналі. Вони виникають через різницю часу відкриття і закривання аналогових перемикачів в ЦАП. Особливо сильний вплив перехідних процесів проявляється, наприклад, коли вхідний код 01...111 змінюється кодом 10...000, а перемикач старшого розряду ЦАП відкривається пізніше, ніж закриваються перемикачі молодших розрядів. Слід зазначити, що сучасні ІС ЦАП відрізняються особливостями підключення (інтерфейсом) до МК: паралельним (як у розглянутій нижче моделі), так і послідовними стандартними інтерфейсами типу I2C або SPI [10,11].

3.1.3 Хід виконання лабораторної роботи

Завдання 1. Зібрати схему (рис. 3.4) для випробування інтегрального ЦАП [11]. У схемі використаний бібліотечний віртуальний 8-розрядний цифроаналоговий перетворювач *A1* (з групи елементів *Mixed*), на входи якого можуть подаватися сформовані за допомогою перемикачів *J1*, ..., *J8* (група *Basic*), відповідні завданням двійкові коди (табл. 3.1). Потрібно виміряти за допомогою вольтметра *V1* (група *Indicator*) або осцилографа *XSC1* вихідні напруги ЦАП, що відповідають вхідним кодами завдання (табл. 3.1). Потім отримані результати заносяться в звіт по ЛР.

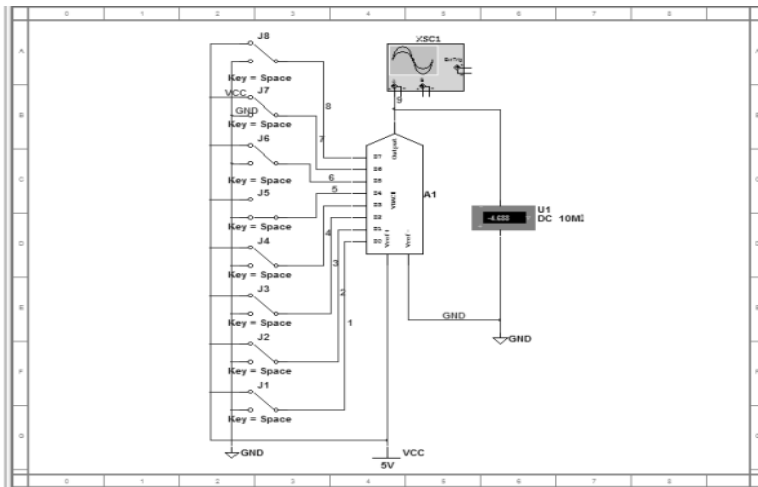


Рисунок 3.4 - Схема спрощеного перетворення цифрового сигналу в аналоговий

Завдання 2. Отримати необхідні вхідні коди ЦАП за допомогою МК-51. Для цього створюємо схемний проект *Circuit6* (рис.3.5), підключаємо ЦАП до виходів порту P2 МК (або будь-якого іншого порту). Отримані значення напруги перетворення ЦАП порівнюємо з результатами завдання 1. Програма спільної роботи МК і ЦАП на C++ показана нижче:

```
#include<8051.h> void main()
{ int i = 0;
  int j = 0;
  unsigned char var[3]; // масив вхідних значень (відповідно до
  варіанту);
```

```
  P2=0x00; // настройка порту P2 на вихід;
```

```
  for(i=0;i<10;i++) // потім виводимо код в циклі в порт P2;
```

```
  {
```

```
    P2 = var [ i]; // перетворення змінної в аналогове значення;
```

```

for(j=0;j<100;j++)
// тимчасова затримка для стійкого вимірювання результату;
continue;
}
while(1);
}

```

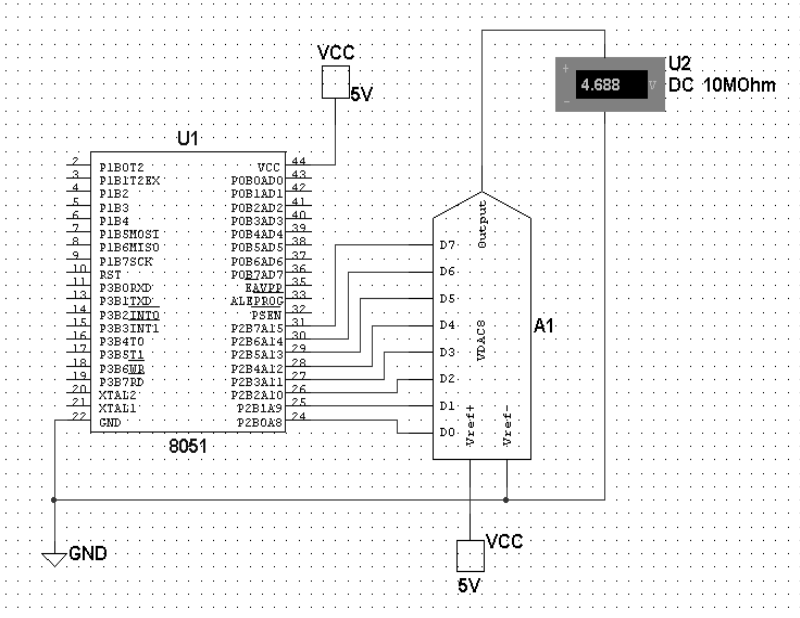


Рисунок 3.5 - Схема взаємодії МК-51 з обраним віртуальним ЦАП

3.1.4 Завдання для лабораторної роботи

Необхідні значення вхідних 8-бітних кодів ЦАП, що відповідають варіанту завдання (табл. 3.1), отримати за допомогою перемикачів або з виходів ВП «генератор кодів» (рис. 3.4, завдання 1) і при підключенні ЦАП до одного з портів МК (рис. 3.5, завдання 3). Використовуючи вихідні схеми рис.3.4, 3.5, створити її варіант / аналог в середовищі MS, LV (ВП типу осцилограф /аналізатор рис.2.10, 2.11, 2.18), перевірити їх функціонування в MS (теоретично: обрати один із способів 1...4 перетворення схеми з MS в LV, див.2.1) і пристрій LV.

Таблиця 3.1- Варіанти завдань

Параметри	Варіанти									
	1	2	3	4	5	6	7	8	9	10
Коди	8	19	13	28	39	47	1	30	55	59
	36	101	99	125	88	165	100	145	155	144
	107	205	243	222	167	250	200	244	198	217
Порт підключення	P1	P2	P3	P1	P2	P3	P1	P2	P3	P1

3.1.5 Зміст звіту

1. Найменування і мета роботи.
2. Копія схемного файлу в MS і LV, діаграми вхідних і вихідних сигналів.
3. Копія програмного файлу (на асемблері або на C+) з подробицями і коментарями по кожній операції і/або модулю.
4. Отримані результати і висновки по роботі.

3.1.6 Питання для самоконтролю

1. Наведіть приклади реалізації послідовних ЦАП, їх основні параметри.
2. Назвіть відмінні риси ЦАП з двійковій зваженими опорами від ЦАП на основі опорів R-2R.
3. Вкажіть переваги і недоліки ЦАП з послідовним інтерфейсом.
4. Назвіть переваги і недоліки ЦАП з паралельним інтерфейсом.
5. Якими параметрами визначається точність перетворення ЦАП?
6. Які схемні особливості використання порту P0 для підключення ЦАП?
7. Як визначається абсолютна роздільна здатність ЦАП при відомому опорному напрузі і розрядності?
8. Чому виникають перехідні процеси на виході ЦАП?

4. Лабораторна робота №3. Вивчення принципів і особливостей роботи аналого-цифрових перетворювачів

Мета роботи: вивчення основ роботи аналоговоцифрових перетворювачів (АЦП) і дослідження особливостей їх функціонування на прикладі віртуального 8-розрядного АЦП, представленого в БМ MS.

4.1 Основні відомості про аналого-цифрові перетворювачі

4.1.1 Принципи перетворення безперервних сигналів у цифрові коди

Аналого-цифровий перетворювач (Analog-to-digital converter, ADC) - це пристрій, який приймає вхідні аналогові сигнали і генерує відповідні їм цифрові сигнали, при- придатні для обробки мікропроцесорами та іншими цифровими пристроями [10].

Як правило, спочатку різні по фізичній природі величини перетворюють в функціонально пов'язані з ними електричні величини, а потім (за допомогою перетворювачів напруга-код) - в цифрові.

Аналого-цифрове перетворення безперервних сигналів, котрі реалізують за допомогою АЦП, це перетворення неперервної функції часу $U(t)$, що описує вихідний сигнал, в послідовність чисел $\{U_k(t_j)\}$, $j = 0, 1, 2, \dots$, $U_k = 0, 1, 2, \dots, N-1$, від несених до деяких фіксованих моментів часу. Таке перетворення складається з двох незалежних процедур. Перша з них називається **дискретизацією** і полягає в перетворенні безперервної функції часу $U(t)$ в безперервну послідовність відліків $\{U(t_j)\}$. Друга називається **квантуванням** і полягає в перетворенні безперервної за значенням послідовності відліків в дискретну $\{U_k(t_j)\}$, где $U_k = 0, 1, 2, \dots, N-1$.

В основі **дискретизації** неперервних сигналів лежить можливість їх подання у вигляді зважених сум $U(t) = \sum_j a_j f_j(t)$,

де a_j - деякі коефіцієнти чи відліки, які характеризують вихідний сигнал в дискретні моменти часу; $j = 1, 2, 3, \dots$;

$f_j(t)$ - набір елементарних функцій, використовуваних при відновленні сигналу за його відліками.

В основі найбільш поширеної рівномірної дискретизації лежить **теорема відліків Котельникова**, для якої в якості коефіцієнтів a_j слід використовувати миттєві значення сигналу $U(t_j)$ в дискретні моменти часу $t_j = j\Delta t$, а період дискретизації вибирати з умови $\Delta t = 1/2F_m$, де F_m - максимальна частота спектра перетворюваного сигналу.

При виконанні процедури **квантування** безперервна за рівнем послідовність відліків $U(t_j)$ представляється к-розрядним кодом $U_k(t_j)$, що

дорівнює $0, 1, 2, \dots, N-1$. Кількість рівнів квантування N залежить від значення розрядності АЦП $N = 2^k$.

Класифікація та принцип дії АЦП

В даний час використовується велика кількість методів перетворення напруга-код [10]. Ці методи відрізняються потенціальною точністю, швидкістю перетворення і складністю апаратної реалізації. На рис. 4.1 представлена класифікація АЦП по методам перетворення. В основі класифікації АЦП використовується ознака, що вказує на те, як в часі розгортається процес перетворення аналогової величини в цифрову. Для перетворення дискретних за часом сигналів в цифрові еквіваленти використовуються операції квантування і кодування. Вони можуть бути здійснені за допомогою як послідовної, або паралельної, або паралельно-послідовної процедур наближення цифрового еквівалента до перетворюваної величини.

Найшвидшими є паралельні АЦП, однак вони найдорожчі і споживають (розсіюють) значну потужність. Послідовно-паралельні АЦП займають проміжне положення по роздільній здатності і швидкодії між паралельними АЦП і послідовними.



Рисунок 4.1 - Класифікація АЦП

Особливості побудови та функціонування різних типів АЦП викладені, наприклад, в [9]. Розглянуті нижче АЦП послідовні рахунки і наближення

знаходить широке застосування в системах управління, контролю і цифрової обробки сигналів.

АЦП послідовного рахунку

Цей перетворювач є типовим прикладом послідовних АЦП з одиничними наближеннями і складається з компаратора, лічильника і ЦАП (рис. 4.2). На один із входів компаратора поступає перетворений аналоговий сигнал, а на іншій - сигнал зворотнього зв'язку з ЦАП.

Робота АЦП починається з приходу імпульсу запуску, який включає лічильник, що підсумовує кількість імпульсів, що надходять від генератора тактових імпульсів - ГТІ. Вихідний код лічильника подається на ЦАП, що здійснює його перетворення в напругу зворотнього зв'язку $U_{ос}$. Процес перетворення триває до тих пір, поки напруга зворотного зв'язку зрівняється з вхідною напругою і переключиться компаратор, який своїм вихідним сигналом припинить надходження тактових імпульсів на лічильник. Перехід вихідного значення компаратора з 1 в 0 означає завершення процесу перетворення.

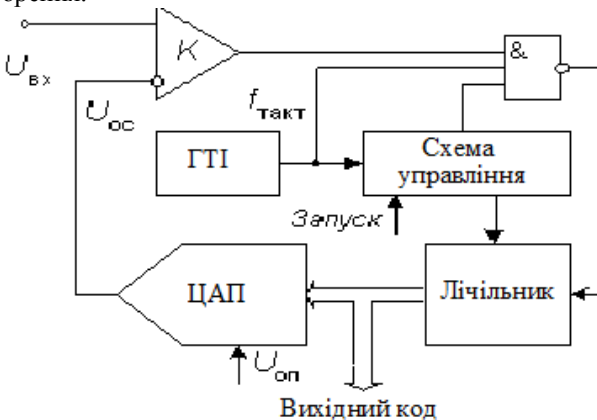


Рисунок 4.2 - Структурна схема АЦП послідовного рахунку

Отриманий вихідний код, пропорційний вхідній напрузі в момент закінчення перетворення, зчитується з виходу лічильника.

АЦП даного типу без пристрою вибірки-зберігання (ПВ/Х) придатні для роботи з постійними або повільно змінюються напругами, які за час перетворення змінюються не більше, ніж на значення кванта перетворення. Таким чином, недоліком АЦП послідовного рахунку є невелика частота дискретизації, що досягає декількох кілогерц.

Призначення і структура пристрою вибірки і зберігання

Пристрій вибірки і зберігання (ПВЗ/Х) призначений для прийому і зберігання на час перетворення входних відліків, рівних напруженням входних сигналів до відповідних дискретні моменти часу. Структурна схема ПВЗ / Х представлена на рис. 4.3. Необхідність зберігання значення входного сигналу протягом часу його перетворення пов'язана з невизначеністю значення цього сигналу в разі швидкого його зміни, що при відсутності ПВЗ/Х може привести до динамічної помилку перетворення. Крім того, ПВЗ/Х вживається в багатоканальних АЦП для запам'ятовування миттєвих значень якого-небудь каналу і його перетворення з одночасно перемиканням мультимплексора на наступний канал.

По передньому фронту тактового сигналу (ТС) пристрій здійснює відлік (вибірку) входного аналогового сигналу при замиканні ключа.

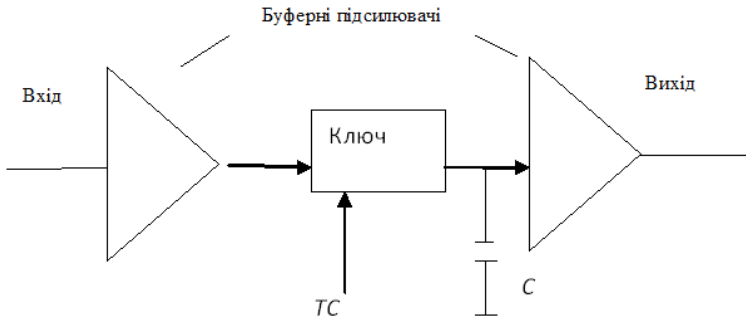


Рисунок 4.3 - Спрощена структурна схема ПВЗ / Х

Режим вибірки характеризується часом захоплення (2, 4, 6, 10 і більше мкс), коли конденсатор заряджається на величину $U_{вх}(t)$. Чим менше час захоплення (менше ємність конденсатора С), тим менше відповідна точність і збільшується падіння напруги в процесі зберігання.

По задньому фронту тактового сигналу ТС відбувається розмикання ключа, і значення відліку (напруги) зберігається, поки АЦП виконує перетворення. Слід зазначити, що збережене напруга не залишається постійною, через наявність витоку заряду конденсатора С. Зв'язок ПВЗ/Х з АЦП представлена на рис. 4.4.

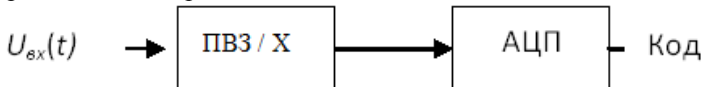


Рисунок 4.4 - Спрощена схема АЦП з ПВЗ / Х

АЦП послідовного наближення

Перетворювач цього типу, також званий АЦП з порозрядним урівнюванням, є найбільш поширеним варіантом послідовних АЦП. В основі

його роботи лежить принцип дихотомії, т. Е. Послідовного порівняння вимірюваної величини з $1/2$, $1/4$, $1/8$ і т. Д. Від можливого максимального значення її.

Розглянемо основи роботи АЦП послідовного наближення на прикладі класичної структури 4-розрядного перетворювача, що складається з трьох основних вузлів: компаратора, регістра послідовного наближення (РПН) і ЦАП (рис. 4.5).

Після подачі команди «Пуск» з приходом першого тактового імпульсу РПН примусово задає на вході ЦАП код, рівний половині його шкали (для 4-розрядного ЦАП це $1000_2 = 8_{10}$). Завдяки цьому напруга зворотного зв'язку на виході ЦАП – $U_{зв.зв} = 2^3h$, де h - квант вихідної напруги ЦАП (відповідає одиниці молодшого розряду- ОМР).

Дане значення U_{oc} становить половину можливого діапазона що перетворюються сигналів. Якщо вхідна напруга АЦП більше, ніж ця величина, то на виході компаратора встановлюється 1, якщо менше, то 0. При 0 на виході компаратора схема управління повинна переключити старший розряд d_3 АЦП назад в стан нуля. отриманий залишок ($U_{вх} - d_32^3h$) таким же чином порівнюється з найближчим молодшим (тобто предстаршим) розрядом і т. д.

Після чотирьох подібних кроків в регістрі послідовного наближення виявляється двійкове число, з якого після цифро аналогового перетворення виходить напруга, відповідним ний значенням $U_{вх}$ з точністю до ОМР. В процесі перетворення на виході компаратора (Рис. 4.5) формується вихідна число у вигляді послідовного коду старшими розрядами вперед.

Даний клас АЦП займає проміжне положення між послідовно-паралельними і інтегруючими АЦП по швидкості дії, вартості та роздільної здатності і знаходить широке застосування в системах управління, контролю і цифровий обробки сигналів.

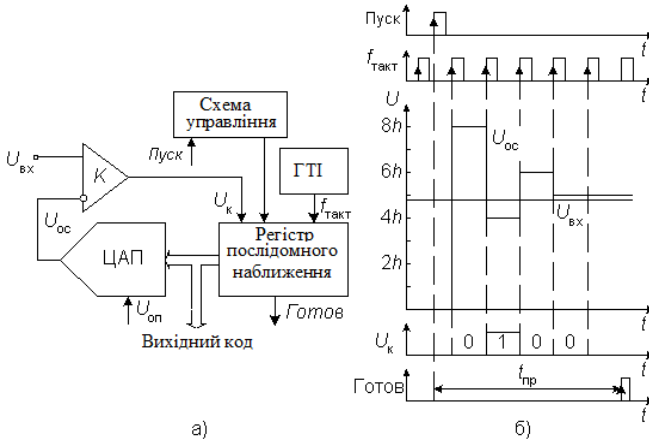


Рисунок 4.5 - Структурна схема АЦП послідовного наближення

4.1.2 Основні параметри АЦП

До основних параметрів АЦП відносять число розрядів вихідного коду, діапазон зміни вхідного напруги $U_{\text{вх. max}}$, абсолютну роздільну здатність, абсолютну похибка перетворення, максимальну частоту перетворення, час перетворення вхідного сигналу.

Число розрядів вихідного коду до відображає вихідну аналогову величину, яка формується на виході АЦП. При використанні двійкового коду $k = \log_2 N$, где N - максимальне число кодових комбінацій (рівнів квантування) на виході АЦП (0, 1, ... $N-1$).

Діапазон зміни вхідного напруги - $u_{\text{вх. max}}$. Відзначимо, що АЦП може обробляти вхідну інформацію у вигляді однополярного аналогового напруження з межами 0, ..., $u_{\text{вх. max}}$ і двополярного $\pm u_{\text{вх. max}}/2$.

Абсолютна роздільна здатність - середнє значення мінімального зміни вхідного сигналу $U_{\text{вх}}$, що обумовлює збільшення або зменшення вихідного коду на одиницю. Роздільна здатність визначається розрядністю вихідного коду і діапазоном вхідної напруги.

Абсолютна похибка перетворення Δi (в кінцевій точці шкали) є відхилення реального максимального значення вхідного сигналу $u_{\text{вх. max}}$ від максимального значення ідеальної характеристики L АЦП (рис. 4.6). Зазвичай Δi вимірюється відносно ЗМР - значення молодшого розряду (ЗМР) і становить $\Delta i \leq (1/2 \dots 1)$ ЗМР.

Максимальна частота перетворення (десятки і сотні кГц) - максимальне значення частоти вхідного сигналу.

Час перетворення вхідного сигналу $t_{\text{пр. max}} \leq (1/2) \Delta t$.

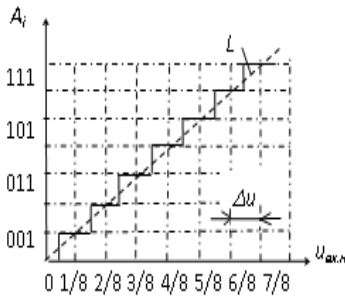


Рисунок 4.6 - Абсолютна похибка перетворення АЦП

4.1.3 Хід виконання лабораторної роботи

Завдання 1. Зібрати схему для дослідження точності перетворення АЦП [11].

Створюємо схемний проект (рис. 4.7), в якому використані елементи: віртуальний 8-розрядний АЦП - A1 і віртуальний ЦАП - A2 з групи елементів Mixed; джерела напружені (з групи елементів Sources): постійного опорного напруження V1 і V2 (підключені до входів Vref+ і Vref-, якщо у варіанті завдання зазначено різнополярних опорна напруга, або Vref+ і земля, якщо використовується тільки позитивне опорна напруга); генератор імпульсної напруги V4 для синхронізації роботи АЦП (підключений до входу SOC); джерело постійної напруги V3 для подачі вхідної напруги, яке необхідно перетворити в код; пробники X0, ..., X9 і вольтметр U1 (з групи елементів Indicators), а також інвертор U1A з групи елементів TTL, який подає перепад напруги від 5 В до 0 на вхід OE АЦП від джерела V4 в протифазі з входом SOC, дозволяючи появу цифрового коду D0 ... D7 на виходах АЦП.

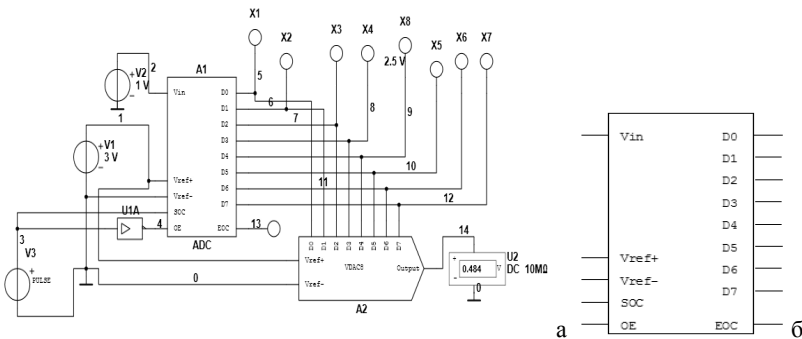


Рисунок 4.7 - Схема для дослідження роботи 8-розрядного АЦП- а; умовне графічне позначення АЦП- б

Таблиця 4.1 - Призначення основних входів і виходів віртуального АЦП

Позначення	Вхід / вихід	Призначення
V _{in}	I	Аналогова вхідна напруга
V _{ref+}	I	Опірна напруга (+)
V _{ref-}	I	Опірна напруга (-)
D0- D7	O	Цифрові поразрядні виходи
SOC	I	Начати перетворення
EOC	O	Кінець перетворення
OE	I	Дозвіл цифрового виводу даних

Призначення виводів використовуваного в схемі АЦП наводиться в табл. 4.1. Дослідження точності АЦП проводиться за наступною схемою [10].

1. Для джерел V1 і V2 встановлюються відповідно завданню (табл. 4.1) значення опорної напруги, наприклад, V1 = +2,5 В, V2 = -2,5 В.

2. По черзі на вхід V_{in} АЦП подається 10 різних напруг з діапазону зміни опорного напруги V_{ref}, наприклад, U_{вх} = 0,1; 0,2; 0,5; 1,0; 1,5; 2,0; 2,4; -0,5; -1,0; -2,0 В (використовується джерело вхідної напруги V3).

3. Для кожного значення вхідної напруги V_{in} запускається програма моделювання. Вихідний код, який визначається за світінням пробників X7, ..., X0, і значення напруги U_{вих} (ЦАП) з виходу ЦАП, вимірювані вольтметром U1, заносяться в поля таблиці виду 4.2.

4. Для 8-розрядного АЦП за значенням вхідної напруги U_{вх} можна визначити значення вихідного коду -

$$D_{(10)\text{расч}} = 256 \times u_{\text{вх}} / (V1 + | - V2 |), \quad (4.1)$$

5. Проводяться необхідні розрахунки для заповнення всіх стовп- зразків табл.

4.2. Якщо значення опорного напруги різнополярні, то код, який визначається за світінням пробників, є інверсним. При цьому, якщо отримане значення коду більше 128, то результат позитивний, а якщо менше - негативний. Десяткові інверсні сигнали D_{(10)інв} перетворюються в прямі D₍₁₀₎ за виразом: $D_{(10)} = D_{(10)\text{інв}} - 128$. (4.2)

6. Розраховуються абсолютні похибки вимірювання напруги за виразом:

$$\Delta U\% = 100\% \times (u_{\text{вих(ЦАП)}} - u_{\text{вх}}) / u_{\text{вх}}. \quad (4.3)$$

Таблиця 4.2 - Приклад дослідження АЦП

u _{вх} , В	u _{вих(ЦАП)} , В	D(2)	D(16)	D(10)інв	D(16)	D(10)розр	ΔU, %
0,1	0,09375	10000100	84	132	4	4,27	6,25
0,5	0,5156	10010101	95	149	21	21,33	3,12
1,0	0,9644	10101010	AA	170	42	42,67	3,56
2,0	2,017	11010101	D5	213	85	85,34	0,85

2,5	2,484	11101010	EA	234	106	106,67	0,64
2,9	2,906	11111011	FB	251	123	123,74	0,21
-1,0	-0,9844	01010101	55	85	-43	-42,67	3,56

Завдання 2. Виконати відповідний варіант завдання при підключенні МК до АЦП. Отримати значення вихідного коду АЦП для будь-яких трьох значень зміни $U_{вх}$ (вибирається з діапазону трансформаційних змін V_{ref} , табл. 4.1).

Створюємо схемний проект Circuit 7, встановлюємо на робочій області МК-51, досліджуваний віртуальний АЦП (8- або 16-розрядний). Подаємо з джерела постійного струму $V1$ на вхід АЦП напруга, наприклад, 1В, до висновку опорного напруги АЦП підключаємо джерело напруги $V2$, наприклад, 5В. До порту P2 МК підключаємо пробники $X0, \dots, X7$, висновок P3.6 з'єднуємо з входом АЦП- SOC. Схеми підключення МК-51 до АЦП зображена на рис. 4.8.

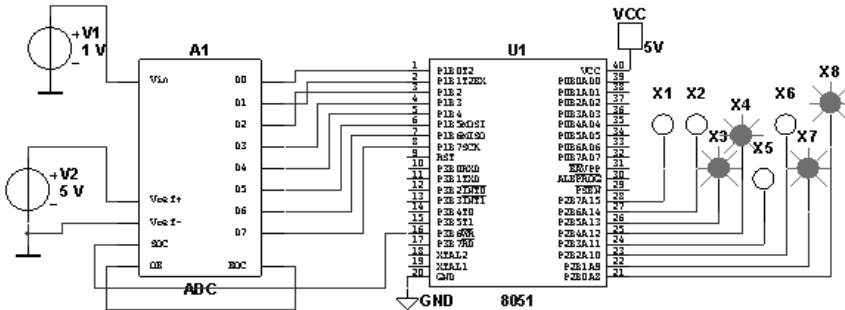


Рисунок 4.8 - Схеми підключення МК-51 до АЦП

Активуємо вкладку програмного файлу main.c і вставимо наступний програмний код:

```
#include <8051.h> // підключаємо заголовки
void main ()
{
    unsigned char val; // змінна містить цифровий код, відповідний  $U_{вх}$ .
    P1 = 0xFF; // ініціалізуємо P1 на вхід
    P2 = 0x00; // ініціалізуємо P2 на вихід
    P36 = 0; //P3.6 - вихід
    P36 = 1; // почати перетворення
    val = P1; // P1 містить цифровий код
    P2 = val; // включаємо пробники
    while (1); //безкінечний цикл //
}
```

На пробнику цифровий код, що дорівнює $51_{(10)}$ (відповідає виразу 4.1).

4.1.4 Завдання для лабораторної роботи

У відповідність до варіанта завдання (табл. 4.1) дослідити точність віртуального АЦП (завдання 1) згідно зі схемою, наведеною на рис. 4.7 і заповнити таблицю виду 4.1. Порядок виконання завдання 1 наведено в п. 4.1.3. Для виконання завдання 2 зібрати схему (рис. 4.8) і для будь-яких семи значень $U_{вх}$ з таблиці 4.2 при допомозі пробників зняти отриманий цифровий код. Використовуючи схеми рис.4.7 і 4.8, (в MS), створити її варіант/аналог вимірювання параметрів в середовищі LW (використат ВП типу осцилограф/логічний аналізатор.

Таблиця 2.14 - Варіанти завдань

№	1	2	3	4	5	6	7	8	9	10
Розрядність АЦП	8	16	8	16	8	16	8	16	8	16
E_{ref+}	+2,5	+2,5	+4	+5,0	+3,0	+1,5	+2	+3,5	+2,5	+1
E_{ref-}	-2,5	0	-1	0	-2	-2,5	-2	0	-1	-1

4.1.5 Зміст звіту

1. Назва та мета роботи, її теоретичної і практичної частини.
2. Перелік елементів, використаних у схемі, з їх короткими характеристиками.
3. Копія схемного файлу в MS і LV, діаграми вхідних і вихідних сигналів.
4. Заповнена при проведенні моделювання по схемі 4.7, табл. 4.1.
5. Зняті показання пробників при моделюванні за схемою 4.8 та заповнена табл. 4.2.
6. Висновки по роботі.

4.1.6 Питання для самоконтролю

1. Наведіть приклади використання АЦП.
2. Дайте визначення роздільної здатності АЦП.
3. Як визначається частота дискретизації аналогового сигналу?
4. У яких випадках використовується двуполярна опорна напруга?
5. Як проводиться квантування аналогового сигналу по рівню?
6. Як визначається цифровий код АЦП при відомому вхідному аналоговому сигналі і заданих опорних напружених?
7. Чим відрізняється принцип роботи АЦП послідовного рахунку від АЦП послідовного наближення?
8. В чому відмінність принципів роботи паралельних АЦП від принципів роботи послідовних АЦП?
9. Для чого в структурі АЦП необхідно використовувати пристрій вибірки та зберігання?
10. Як визначається відносна точність перетворення АЦП?

ЛИТЕРАТУРА

1. Шегал А. А. Применение программного комплекса Multisim для проектирования устройств на микроконтроллерах : лабораторный практикум / А. А. Шегал. - Екатеринбург : Изд-во Урал. ун-та, 2014. - 114с.
2. Каспер Э. Программирование на языке Ассемблера для микроконтроллеров семейства i8051 / Э. Каспер - М.: Горячая линия-Телеком, 2004. - 192 с.
3. Магда Ю. С. Микроконтроллеры серии 8051: практический подход / Ю. С. Магда - М.: ДМК Пресс, 2008. - 228 с.
4. Баррет С. Ф. Встраиваемые системы. Проектирование приложений на микроконтроллерах семейства 68HC12/HCS12 с применением языка С / С. Ф. Барретт, Д. Дж. Пак. М. : ДМК- пресс, 2007. 640с.
5. Микушин А. В. Занимательно о микроконтроллерах / А. В. Микушин. С-Пб. : БХВ-Петербург, 2006. 432 с.
6. Хернтер М. Е. Multisim 7 Современная система компьютерного моделирования и анализа схем электронных устройств / М. Е. Хернтер – М.: «ДМК - Пресс», 2006. – 488 с.
7. Загидуллин Р. Ш. Multisim, LabView, Signal Express /Р. Ш. Загидуллин. М. : Горячая линия-Телеком, 2009. 366 с.
8. Марченко А. Л. Лабораторный практикум по электротехнике и электронике в среде Multisim: учебное пособие для вузов / А. М. Марченко, С. В. Освальд. М. : ДМК Пресс, 2010. 448 с.
9. Белов А. В. Конструирование устройств на микроконтроллерах / А. В. Белов. СПб. : Наука и Техника, 2005. 256 с.
10. Угрюмов Е. П. Цифровая схемотехника: учебное пособие / Е. П. Угрюмов. СПб. БХВ. С-Пб. : 2010. 528 с.
11. Волович Г. И. Схемотехника аналоговых и аналого-цифровых электронных устройств / Г. И. Волович. М. : Издательский дом «Додэка-XXI», 2005. 528 с.