

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЗАПОРІЗЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

Інститут інформаційних та радіоелектронічних
Технологій радіоелектроніки і телекомунікацій
(повне найменування інституту, назва факультету)
Інформаційних технологій електронних даних
(повна назва кафедри)

Пояснювальна записка
до магістерської роботи

(ступінь вищої освіти (освітній ступінь))

на тему Розробка лабораторії з вивченням процесів
електронної апертури

Виконав: студент 6 курсу, групи РТ-534
спеціальності (спеціалізації)

Технології інформаційних та радіоелектронічних
(код і назва спеціальності, спеціалізації)

Лешохін К.В.

(прізвище та ініціали)

Керівник

Григорук Н.І.

(прізвище та ініціали)

Рецензент

Середобров М.П.

(прізвище та ініціали)

м. Запоріжжя
20 18 рік

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Запорізький національний технічний університет
(повне найменування вищого навчального закладу)

Інститут, факультет УІРЕ, ФРЕТ
 Кафедра інформаційних технологій електронних засобів
 Ступінь вищої освіти (освітній ступінь) _____
 Спеціальність 192 Телекомунікації та радіотехніка
(код і назва)
 Спеціалізація „Мікроелектронна техніка“
(код і назва)

ЗАТВЕРДЖУЮ

Завідувач кафедри Т.М. Мило"13" жовтня 2018 року

З А В Д А Н Н Я
НА МАГІСТЕРСЬКУ РОБОТУ СТУДЕНТУ

Лисовиху Костянтину Валерійовичу
(прізвище, ім'я, по батькові)

1. Тема проекту (роботи) Розробка лабораторії з вивченням
досліджень за електронної електроніки

керівник проекту (роботи) Гурманов Н.Г.; к.т.н., доцент
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу від "04" листопада 2018 року № 337

2. Строк подання студентом проекту (роботи) 13 жовтня 2018 року

3. Вихідні дані до проекту (роботи) визначення УДМ 4.2 за
ГОСТ 15150-69, матеріальність вивчення
досліджень

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) 1. Опис лабораторії для проведення лабораторних та інших
робот, з вивченням лабораторії з вивченням електронних засобів.
2. Проведення аналізу роботи лабораторії з вивченням електронних засобів.
3. Проведення аналізу роботи лабораторії з вивченням електронних засобів.
4. Проведення аналізу роботи лабораторії з вивченням електронних засобів.
5. Проведення аналізу роботи лабораторії з вивченням електронних засобів.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

1. Креслення лабораторії з вивченням електронних засобів
2. Креслення лабораторії з вивченням електронних засобів
3. Авторський розрахунок роботи лабораторії з вивченням електронних засобів
4. Схема електронних засобів, електронних засобів лабораторії, електронних засобів лабораторії

6. Консультанти розділів проекту (роботи)		Підпис, дата	
Розділ	Прізвище, ініціали та посада консультанта	завдання видан	прийняв виконання завдання
1-4	Гурменова Н. Г. КТН доц. каф ТТБЗ	03.03.18	09.11.18
5	Левченко Н. М. д.в.н; проф. каф. ПТ та БД		
6	Гусаровий А. О.; ст. викл каф. ОП і НС		
Математич.	Домінік Т. Е., ст. викл. каф. ТТБЗ		

7. Дата видачі завдання 03.09.18

№ з/п	Назва етапів магістерської роботи	Срок виконання етапів роботи	Примітки
1	Аналіз наукових вісників та інформаційних ресурсів	1 тиждень	
2	Аналіз інформації з питань кримінального права	2 тижні	
3	Робота над інформаційною базою з кримінального права	3 тижні	
4	Робота над інформаційною базою	4 тижні	
5	Робота над інформаційною базою	5 тижнів	
6	Робота над інформаційною базою з кримінального права	6 тижнів	
7	Завершення магістерської роботи	7 тижнів	

Студент

Керівник роботи

Левський К. В.
(ПРИЗВИЩЕ ТА ІНІЦІАЛИ)

Гурманова Н. І

РЕФЕРАТ

Магістерська робота складається: 136 с., 42 рис., 6 табл., 24 джерел, 4 додатків.

Мета роботи – розробка лабораторії з віддаленим доступом до електронної апаратури.

У даній роботі розглянуті способи електронних форм навчання, їх переваги і недоліки, запропонований сучасний і ефективний спосіб дистанційного навчання.

Створений проект лабораторії з віддаленим доступом для дисципліни «Електронні прилади». Розроблене програмне забезпечення для керуючого блоку та комп'ютера, що входить до складу системи.

В якості експериментального стенду була використана установка для зняття вольт-амперних характеристик та світлових характеристик фотоприладів для виконання лабораторної роботи «Дослідження фотопровідності напівпровідників та фотоелектричних властивостей р-n переходу». Дана установка була перетворена для цифрового управління. В роботі представлена електрична принципова схема удосконаленої установки.

ЛАБОРАТОРІЯ З ВІДДАЛЕНИМ ДОСТУПОМ, ЦИФРОВА І МІКРОПРОЦЕСОРНА ТЕХНІКА, МІКРОКОНТРОЛЕРИ, СЕРВЕР, ІНТЕРФЕЙС, ЛАБОРАТОРНІ РОБОТИ, ДИСТАНЦІЙНА ОСВІТА, МЕТОДИЧНІ ВКАЗІВКИ, ЕЛЕКТРОННИЙ СТЕНД

ЗМІСТ

ВСТУП.....	6
1 ОГЛЯД ІСНУЮЧИХ ЛАБОРАТОРІЙ ДЛЯ ПРОВЕДЕННЯ ЛАБОРАТОРНИХ ТА ПРАКТИЧНИХ РОБІТ	10
1.1 Використання програм – симуляторів	13
1.1.1 Electronics Workbench.....	13
1.1.2 Proteus.....	15
1.1.3 MPLAB	17
1.1.4 Висновок	19
1.2 Використання систем моделювання	19
1.2.1 Altium Designer.....	20
1.2.2 Micro-Cap	21
1.2.3 MATLAB.....	24
1.2.4 Висновок	26
1.3 Віртуальні лабораторії.....	27
2 ВІРТУАЛЬНІ ЛАБОРАТОРІЇ З ВИКОРИСТАННЯМ РЕАЛЬНИХ ПРИСТРОЇВ.....	32
2.1 Визначення поняття віртуальна лабораторія з використанням реального пристрою	32
2.2 Класифікація віртуальних лабораторій з віддаленим доступом	32
2.2.1 Дистанційна лабораторія GymKT	35
2.2.2 RELLE (Remote Labs Learning Environment).....	38
2.2.3 Висновок	41
3 ПРОЕКТУВАННЯ АПАРАТУРИ ЗАБЕЗПЕЧЕННЯ ДЛЯ ПРОВЕДЕННЯ ДОСЛІДЖЕНЬ.....	44
3.1 Апаратна частина лабораторії	45
3.1.1 Експериментальна установка	45
3.1.2 Керуючий пристрій.....	49

4 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ТА МЕТОЛОГІЙ ДЛЯ ПРОВЕДЕННЯ ДОСЛІДЖЕНЬ	58
4.1 Програмна частина лабораторії	58
4.1.1 Розробка програмного забезпечення та алгоритму роботи для керуючого комп'ютера.....	58
4.1.2 Розробка та алгоритм роботи програми для мікроконтролера	71
4.1.3 Розробка користувальницького інтерфейсу лабораторій	86
4.2 Методика роботи лабораторії з віддаленим доступом з дисципліни «Електронні прилади»	90
5 ЕКОНОМІЧНЕ ОБГРУНТУВАННЯ	93
5.1 Техніко-економічне обґрунтування	93
6 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ	105
6.1 Аналіз потенційних небезпечних чинників при проведенні науково- дослідної роботи.....	105
6.2 Заходи щодо забезпечення безпеки.....	106
6.3 Заходи щодо забезпечення виробничої санітарії та гігієни праці	108
6.4 Заходи з пожежної безпеки	112
6.5 Заходи з цивільного захисту	113
ВИСНОВОК.....	117
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ	119
ДОДАТОК А.....	122
ДОДАТОК Б	123
ДОДАТОК В	126
ДОДАТОК Г	130

ВСТУП

Освіта є основою інтелектуального, духовного, фізичного і культурного розвитку особистості, її успішної соціалізації, економічного добробуту, запорукою розвитку суспільства, об'єднаного спільними цінностями і культурою, та держави [1].

В рамках традиційного підходу до освітнього процесу вважається, що лабораторно практикуми в природничій області (фізика, хімія, біологія і т.п.), а також в техніці (електроніка, електротехніка, інформаційні технології і т.п.) повинні бути виконані переважно на реальному обладнанні, а навчання на моделях електронних і електротехнічних пристроїв в програмних продуктах моделювання спричинить тільки до недооцінки випускниками важливості справжніх експериментів.

Сучасний стан розвитку промисловості характеризується автоматизацією виробництва, постійною модернізацією технологічних ліній та удосконаленням обладнання. Таким чином, потреби у висококваліфікованих спеціалістах для проектування, монтажу обслуговування технологічного устаткування зростають. До основних особливостей навчання студентів технічного напрямку відносять значну частину експериментальних досліджень роботи апаратів і процесів виробництва, побудова і аналіз математичних моделей технологічних процесів. Найскладніші моменти виникають при вивченні роботи виробничого обладнання, оскільки необхідно чітко представити технологічні процеси. Для забезпечення високої якості навчання спеціалістів технічного напрямку, важливою умовою є матеріальна база кафедри, що дозволяє максимально відобразити технологічні умови виробництва, змодельовати дослідні та експериментальні процеси на установках .

З метою формування у студентів практичних вмінь і навичок у сучасній освіті сформувалися такі навчальні напрямки для виконання експериментальних досліджень: традиційні локальні лабораторії, віртуальні лабораторії (Virtual Laboratories), лабораторії з віддаленим доступом (Remote

Laboratories).

В даний час у всьому світі вже існує великий досвід із використання віртуальних навчальних лабораторій. Особливий акцент встановлюється на лабораторіях, що демонструють технологічні процеси у технічній освіті.

Активно розвиваються мережеві програмні системи управління навчальним процесом. Ці системи включають основні функції організації електронного навчання – реєстрацію, підтримку самостійної роботи, організацію індивідуальної та групової взаємодії студентів і викладачів, проміжне і підсумкове тестування та ряд інших функцій, що забезпечують високоякісне навчання та засвоєння матеріалу під час навчального процесу.

Традиційно навчальні лабораторії оснащуються певним набором технічних засобів, що дозволяють проводити найпростіші досліди. Тому для більшої наочності навчального процесу використовуються наступні засоби: відеофільми, комп'ютерні моделі процесів, експериментальні моделі технологічних апаратів, математичні моделі процесів, досліди на реальному технологічному об'єкті. Відзняті на виробництві відеофільми демонструють роботу устаткування, технологічних ліній. Для демонстрації процесів, важких для сприйняття й теоретичного аналізу, використовується моделювання фундаментальних завдань за допомогою комп'ютерних моделей, що потребує значних вкладень в закордонні ліцензійні програми. При розробці віртуальних лабораторій віддаленого доступу на базі імітаційних моделей і реального устаткування виникає проблема дистанційного керування лабораторним стендом і відображення пульта управління на екрані студента. Середовище програмування надає широкі можливості для створення програмного забезпечення по управлінню реальними об'єктами, зокрема з можливістю віддаленого доступу через мережу Internet.

Використання в навчальному процесі віртуальних лабораторій дозволяє забезпечити:

- можливість модульного поділу лабораторної роботи;
- можливість індивідуальне навчання без перерв;

- можливість паралельного використання на лекційних, практичних і лабораторних заняттях;
- поетапне вивчення технологічних процесів;
- можливість аналізу експериментальних даних одночасно з експериментом;
- можливість модифікації і вдосконалення, внесення коректив до існуючої моделі.

Переваги від використання віртуальних лабораторних робіт, нівелюють такі недоліки, як необхідність для розробки спеціального програмного забезпечення, перегляд запрограмованих технологічних процесів, відсутність студента біля технологічного об'єкту під час експериментів.

Використання віртуальних лабораторій у навчальному процесі вищого технічного закладу дає змогу, з одного боку, отримати практичні навички проведення експериментів, ознайомитися детально з комп'ютерною моделлю сучасного обладнання, досліджувати пожежо- і вибухонебезпечні процеси і явища, не побоюючись за можливі наслідки. З іншого боку, є можливість організувати взаємодію віртуального лабораторного комплексу із реально діючим обладнанням для збору експериментальних даних, що забезпечить відповідний рівень розвитку наукових розробок і технологічного керування процесом.

Аналіз переваг і недоліків різних видів лабораторних занять показав доцільність використання віртуальних лабораторних робіт для виконання дослідів. При створенні віртуального лабораторного комплексу для технічно-технологічних спеціальностей дотримуємося ряду основних принципів:

- забезпечити масовість розповсюдження практикуму в Інтернет мережі, що передбачає виконання дослідів кожним студентом окремо, із різним темпом і глибиною вивчення технологічних процесів;
- забезпечити комплексно-модульну структуру практикуму, що дозволить використовувати наявні матеріали під час практичних, лабораторних і лекційних занять;

- можливість застосовувати віртуальні лабораторні заняття для різних форм навчання: денної, заочної, дистанційної.

Отже, розробка віртуального навчального стенду є досить суттєвим та фундаментальним внеском при проведенні освітніх програм [2].

Крім того, завдяки віртуальним лабораторіям вирішується проблема доступу до освіти людей які перебувають закордоном та людей з обмеженими можливостями, що визначає територіальний ще і соціально-значимий характер даної роботи.

Таким чином, метою магістерської роботи є розробка проекту віртуальної лабораторії з віддаленим доступом для дистанційної освіти студентів, які навчаються за напрямом підготовки бакалаврів «Радіотехніка та телекомунікації» з дисципліни «Електронні прилади».

1 ОГЛЯД ІСНУЮЧИХ ЛАБОРАТОРІЙ ДЛЯ ПРОВЕДЕННЯ ЛАБОРАТОРНИХ ТА ПРАКТИЧНИХ РОБІТ

Завдання підготовки конкурентоспроможних фахівців у різних сферах діяльності є актуальною для навчальних закладів вищої професійної освіти, головною метою яких є підготовка бакалаврів, які володіють стрункою системою знань, що володіють здатністю до вирішення професійних завдань в різних ситуаціях.

Для досягнення цієї мети особливого значення набуває організація практичної навчальної діяльності майбутніх ІТ-фахівців, оскільки навіть необхідний теоретичний об'єм знань не зможе повністю забезпечити вироблення у студента тих умінь, які були б оцінені сучасними роботодавцями.

Інакше кажучи, при вивченні технічних, особливо інженерних, дисциплін невід'ємною складовою частиною навчального процесу є лабораторний практикум, завданням якого є формування у студентів практичних навичок роботи з обладнанням, отримання та обробки експериментальних даних, умінь планувати експеримент, аналізувати і зіставляти отримані результати з літературними даними.

Одним з головних напрямків інформатизації в навчальному закладі є поширення різних електронних видів і форм навчання. Все більш затребуваним способом отримання нових знань в світі стає система електронного навчання, яка в Україні також називається дистанційним навчанням, навчанням із застосуванням комп'ютерів, віртуальним навчанням і т.д.

Дистанційне навчання - це віддалений спосіб навчання, при якому викладач і студент знаходяться в різних місцях фізично. Історично, дистанційне навчання вважалося заочним. Але зараз - це засіб навчання, яке передбачає використання аудіо, відеотехніки, Інтернету і супутникових каналів зв'язку [3].

Однак, з огляду на важливість розвитку у студентів-інженерів практичних навичок роботи з пристроями і процесами, виникають складності об'єднання таких двох понять, як «дистанційне навчання» і «лабораторні практикуми».

Проте, в «Законі України про освіту» наголошується на необхідності застосування електронних систем навчання в навчальному процесі, незважаючи на те, що їх складно реалізувати для виконання лабораторних практикумів з інженерних предметів.

Потреба у створенні електронних практичних робіт пояснюється такими факторами, як:

- отримання доступу до цінного , унікального , важкодоступного , потужного лабораторного обладнання, яке може бути використане не тільки для проведення наукових досліджень, а й в освітніх цілях;
- забезпечення нерозривності процесу освіти і наукових досліджень, дистанційний доступ до електронного практикуму з будь-якої точки світу;
- можливість забезпечення доступу до освіти людей з обмеженими здібностями.

Порівняння лабораторії з віддаленим доступом та традиційної локальної лабораторій представлено у табл.1.1, де показано їх основні переваги та недоліки [2].

На сучасному етапі розвитку науки і техніки інформаційні технології дозволяють розробити дистанційні лабораторні практикуми для студентів-інженерів, використовуючи вже існуючі в науці механізми реалізації [4]. Електронні системи навчання, засновані на застосуванні цих механізмів можна розділити на три групи:

- 1) електронні лабораторні роботи з використанням програм - стимуляторів;
- 2) електронні лабораторні роботи з використанням систем моделювання;
- 3) віртуальні лабораторії.

Зупинимося докладніше на кожній з них.

Таблиця 1.1 – Порівняльна характеристика лабораторій

	Лабораторія з віддаленим доступом	Традиційна локальна лабораторія
Переваги	– Виконання дослідів в он-лайн режимі на коштовному, часто унікальному устаткуванні; – Можливість використання у будь-якій формі навчального процесу; – Немає потреби в установці спеціального програмного забезпечення; – Об'єднання дослідників із різних організацій, університетів, обговорення результатів роботи.	– Висока наочність процесів; – Засвоєння техніки безпеки і вміння працювати із обладнанням; – Сприймаються більшістю видів почуттів.
Недоліки	– Відсутність біля технологічного об'єкту під час експериментів; – Узгодження роботи із технологічних обладнанням з іншими обслуговуючим персоналом.	– Значні фінансові витрати на створення стенду; – Забезпечення функціонування стенду спеціальним персоналом; – Проведення занять за узгодженим графіком; – Заняття колективні, тому обмежений доступ студентів до устаткування; – Неможливість дослідження екстремальних і аварійних режимів.

1.1 Використання програм – симуляторів

Під програмою - симулятором в даному випадку розуміється комп'ютерна програма, робота якої заснована на симуляції роботи пристроїв і процесів і аналізі роботи з ними. Від інших подібних програм вони відрізняються тим, що побудовані лише на відображенні заздалегідь закладених в моделях пристроїв алгоритмів, не проявляючи всіх тонкощів моделювання, тому що користувач не має можливості задати або змінити форму самої моделі. В процесі навчання найчастіше такі програми застосовуються для демонстрації основ електроніки та електротехніки, схемотехніки і мікропроцесорних систем, тому що дозволяють більш наочно і точно показати принцип роботи електричної схеми, пристрої тощо.

Розглянемо найбільш відомі програми - стимулятори, які підходять для дистанційного навчання студентів з технічних дисциплін.

1.1.1 Electronics Workbench

Electronics Workbench (EWB) - система схемо-технічного моделювання, призначена для моделювання та аналізу електричних схем. програма ElectronicsWorkbench дозволяє моделювати аналогові, цифрові та цифро-аналогові схеми великій мірі складності. Наявні в програмі бібліотеки включають в себе великий набір широко поширених електронних компонентів. Інтерфейс програми представлений на рис. 1.1.

Параметри компонентів можна змінювати в широкому діапазоні значень. Прості компоненти описуються набором параметрів, значення яких можна змінювати безпосередньо з клавіатури, активні елементи - моделлю, яка представляє собою сукупність параметрів і описує конкретний елемент або його ідеальне уявлення. Модель вибирається зі списку бібліотек компонентів, параметри моделі також можуть бути змінені користувачем. Широкий набір приладів дозволяє проводити вимірювання різних величин, задавати вхідні

впливу, будувати графіки. Всі прилади зображуються у вигляді, максимально наближеному до реального, тому працювати з ними просто і зручно. Результати моделювання можна вивести на принтер або імпортувати в текстовий або графічний редактор для їх подальшої обробки. Програма Electronics Workbench сумісна з програмою P-SPICE, тобто надає можливість експорту та імпорту схем і результатів вимірювань в різні її версії.

Незважаючи на все це, програма не дозволяє створювати нові або змінювати вже існуючі моделі компонентів. Вона відмінно підійде для налагодження роботи простий схеми, але використовувати EWB для проведення наукових досліджень неможливо.

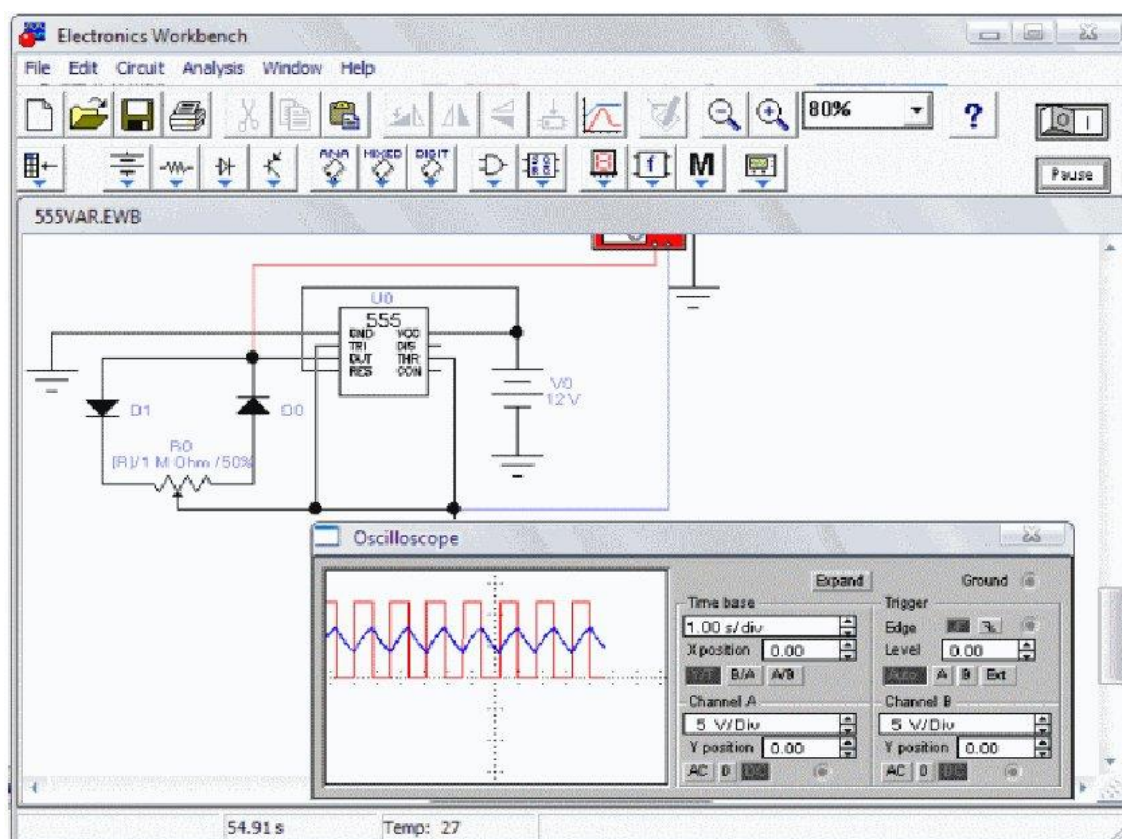


Рисунок 1.1 - Програма Electronics Workbench [5]

1.1.2 Proteus

Proteus - програма для автоматичного проектування електричних схем, являє собою потужну систему схемо технічного моделювання, зроблену на основі віртуальних моделей електронних елементів. Специфічною рисою даного програмного пакету (Proteus) - є відмінна можливість моделювання різної роботи програмованих пристроїв: мікропроцесорів, контролерів, DSP і т.д.

Крім цього в пакет Proteus закладена спеціальна система проектування і моделювання друкованих плат. Програма вміє симулювати роботу таких контролерів, як ARM7, 8051, PIC, AVR, Motorola, Basic Stamp. Внутрішня бібліотека компонентів має різні довідкові дані. Вона підтримує МК: 8051, PIC, HC11, AVR, ARM7 / LPC2000 і інші широко поширені процесори. Додатково до цього в програмі містяться більше 6000 цифрових і аналогових моделей всіляких пристроїв. Інтерфейс програми представлений на рис. 1.2.

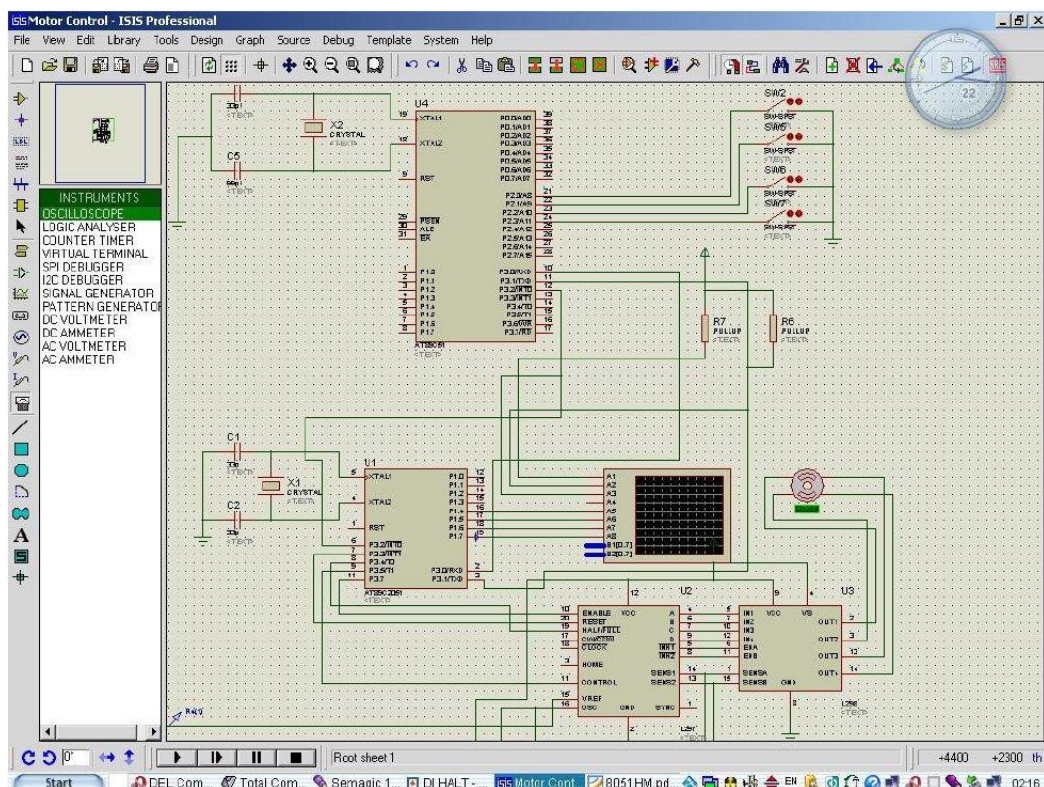


Рисунок 1.2 - Інтерфейс програми Proteus [6]

Програма Proteus прекрасно працює з більшістю компіляторів і асемблер. PROTEUS VSM робить досить достовірно моделювання та налагодження вельми складних пристроїв, в яких може перебувати кілька МК різних сімейств в одному пристрої.

Програма Proteus складається з 2 основних модулів:

1) ISIS - це графічний редактор принципів електронних схем, який служить для введення проектів з подальшою імітацією і передачею на розробку різних друкованих плат в ARES. Після загальної налагодження пристрою можна розвести наявну друковану плату в ARES, яка має підтримку автоматичного розміщення і трасування за існуючою схемою;

2) ARES - це графічний редактор друкованих плат з вбудованим авто трасувальником ELECTRA, автоматичної розстановкою компонентів на друкованій платі і відмінним менеджером бібліотек (рис. 1.3).

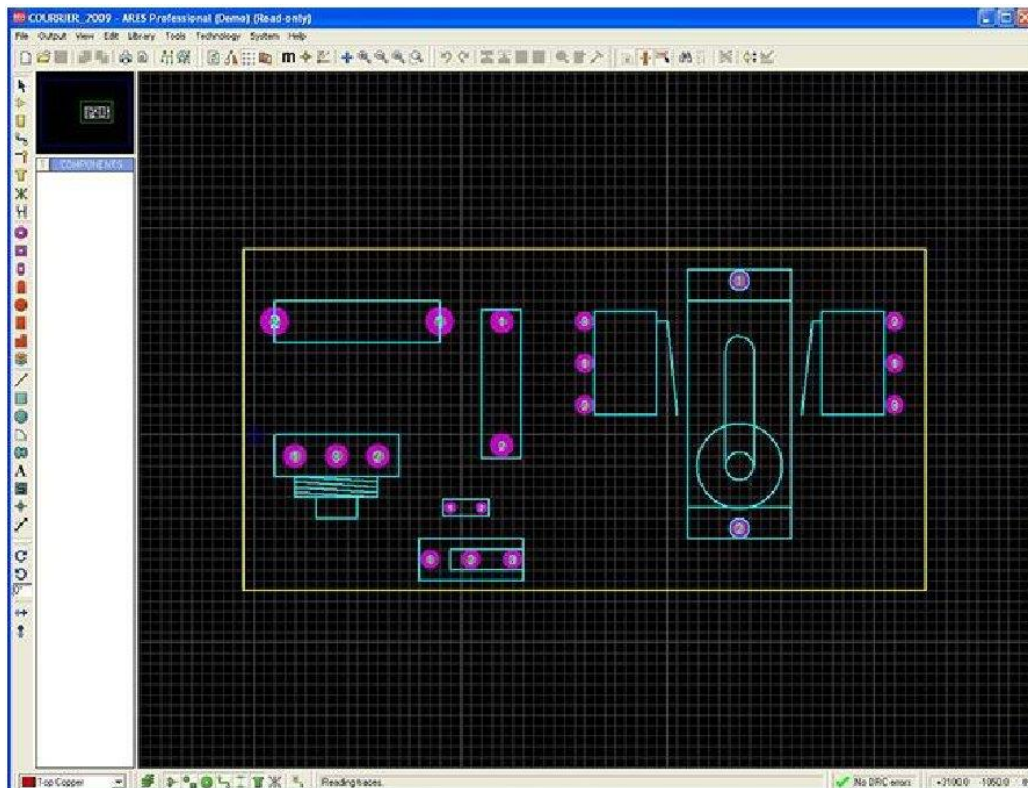


Рисунок 1.3 - Графічний редактор друкованих плат ARES програми Proteus

Програма PROTEUS має відмінні можливості:

1. COMPIM - цей компонент дозволяє вашого віртуального пристрою підключитися до реального COM-порту вашого ПК;
2. USBCONN - цей інструмент дозволяє підключитися до реального USB порту комп'ютера.

Програма PROTEUS VSM буде ідеальним рішенням для новачків, які вирішили почати вивчення роботи мікроконтролерів [7].

1.1.3 MPLAB

MPLAB - це інтегроване середовище розробки (IDE) для сімейства мікроконтролерів PICmicro фірми Microchip Technology Incorporated (рисунок 1.4). MPLAB дозволяє писати, налагоджувати та оптимізувати програми для Ваших розробок. MPLAB включає текстовий редактор, симулятор (віртуальний отладчик), менеджер проектів та підтримує емулятори (внутрішні отладчики) MPLAB-ICE і PICMASTER, програматори PICSTART Plus і PRO MATE II і інші засоби, і інструменти розробок фірми Microchip і інших фірм.

Інструментальні засоби MPLAB, організовані як спадаючі меню і визначаються швидкі клавіші, дозволяють:

- асемблювати, компілювати вихідний текст;
- налагоджувати логіку роботи, спостерігаючи за допомогою симулятора або, в реальному часі, з емулятором MPLAB-ICE;
- переглядати змінні в вікнах перегляду;
- програмувати кристали за допомогою програматорів PICSTART Plus або PRO MATE II і багато іншого.

MPLAB працює під Microsoft Windows 3.1x, Windows 95, 98, NT, 2000 (починаючи з версії 5.00.00). Правда не всі додаткове обладнання, зокрема внутрішньосхемні емулятори та програматори, буде функціонувати під усіма операційними системами.

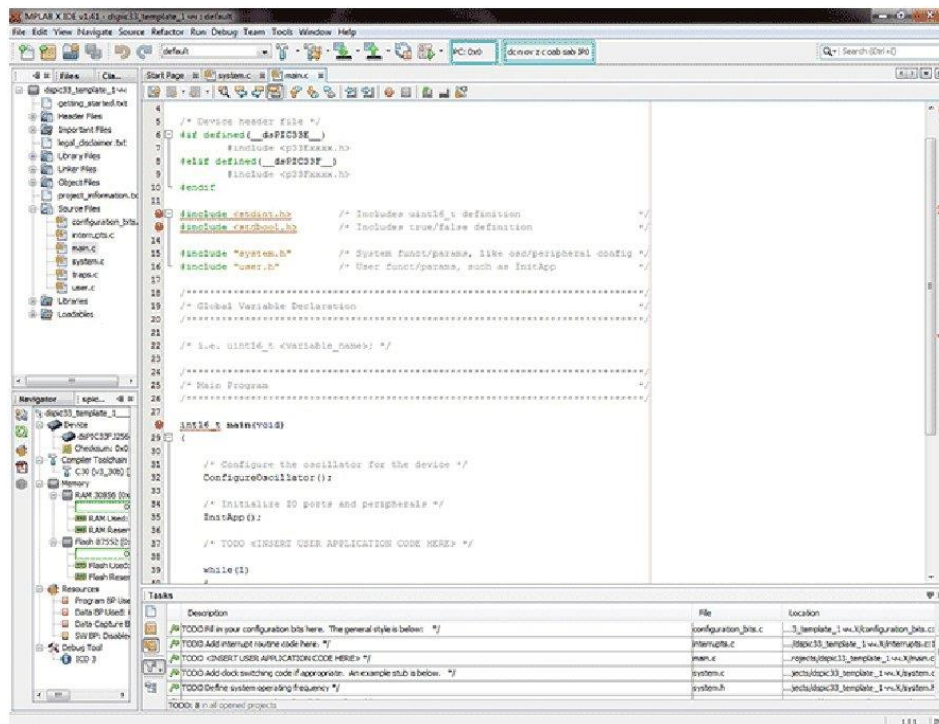


Рисунок 1.4 - Інтерфейс програми MPLAB [8]

MPLAB є закінченою середу розробки, інтегруючи кілька інструментальних засобів:

- 1) MPLAB менеджер проектів (project Manager). Використовується для створення проектів і роботи зі зв'язаними файлами;
- 2) MPLAB редактор (Editor). Використовується для створення і редагування текстових файлів, таких як вихідні і файли сценаріїв компонування;
- 3) MPLAB-SIM симулятор (Simulator). Дозволяє моделювати виконання команд і вхідні / вихідні сигнали мікроконтролерів;
- 4) MPLAB-ICE внутрісхемний емулятор (Emulator). Дозволяє, використовуючи додаткові апаратні засоби і комп'ютер, замінювати мікроконтролер в розробному пристрої в реальному часі. MPLAB-ICE самий новий емулятор від Microchip;
- 5) MPASM універсальний асемблер;

- 6) MPLINK компоновщик (linker). створює закінчений додаток, пов'язуючи переміщувані модулі MPASM, MPLAB-C17, і MPLAB-C18;
- 7) MPLIB управляє користувацькими бібліотеками;
- 8) MPLAB-CXX компілятор C. Дозволяє включати в проект вихідні тексти, написані на мові високого рівня C і на асемблері;
- 9) PRO MATE II і PICSTART Plus програматори;
- 10) PICMASTER і PICMASTER-CE внутрісхемні емулятори.

1.1.4 Висновок

Програми - симулятори можуть бути використані в якості платформи для проведення віддалених лабораторних роботи для навчання базовим знанням студентів технічних спеціальностей. Вони дозволяють наочно продемонструвати принципи роботи схем, процесорів, мікроконтролерів і тому подібне.

Однак студент не може керувати моделлю пристрою, змінювати її параметри і значення, тому отримані в ході виконання роботи знання і навички будуть недостатньо повними. Крім того, пристрої, побудовані в таких програмах, що будуть працювати тільки за заданим алгоритмом, що для якісного навчання фахівців-практиків абсолютно не підходить, тому що студентам необхідно вміти працювати з реальним обладнанням, а не з його симуляцією.

1.2 Використання систем моделювання

Іншим механізмом реалізації дистанційних практикумів є побудова лабораторних робіт на основі застосування систем моделювання. В даному контексті під системою моделювання розуміється комп'ютерна програма, що моделює роботу пристроїв і процесів. Від програм - симуляторів системи моделювання відрізняються тим, що в останніх крім використання вже готових

моделей пристроїв присутній можливість створювати свою форму моделі або змінювати вже готову. Такі програми, як правило, допомагають розвинути навички в розробці і конструюванні нових схем і пристроїв, налагодженні їх принципів роботи і визначенні їх властивостей [9].

Найпопулярнішими рішеннями в даній сфері є такі програми, як Altium Designer, Micro-Cap, MATLAB та інші.

1.2.1 Altium Designer

AltiumDesigner - комплексна система автоматизованого проектування (САПР) радіоелектронних засобів, розроблена австралійською компанією Altium (рис. 1.5). Спочатку програма мала назву Protel. Сьогодні Altium Designer - це система, що дозволяє реалізовувати проекти електронних засобів на рівні схеми або програмного коду з наступною передачею інформації проектувальнику ПЛІС або печатні плати. Відмінною особливістю програми є проектна структура і наскрізна цілісність ведення розробки на різних рівнях проектування. Іншими словами, зміни в розробці на рівні плати можуть миттєво бути передані на рівень ПЛІС або схеми і так само назад. Даний пакет складається з двох продуктів, ба базуються на єдиній інтегрованій платформі DXР, можливість роботи з тим або іншим з них залежить від типу придбаної ліцензії:

- Altium Designer Custom Board Front-End Design - Проектування ПЛІС, схемотехнічне проектування і моделювання;
- Altium Designer Custom Board Implementation - Проектування друкованих плат і ПЛІС.

До складу програмного комплексу Altium Designer входить весь необхідний інструментарій для розробки, редагування та налагодження проектів на базі електричних схем і ПЛІС.

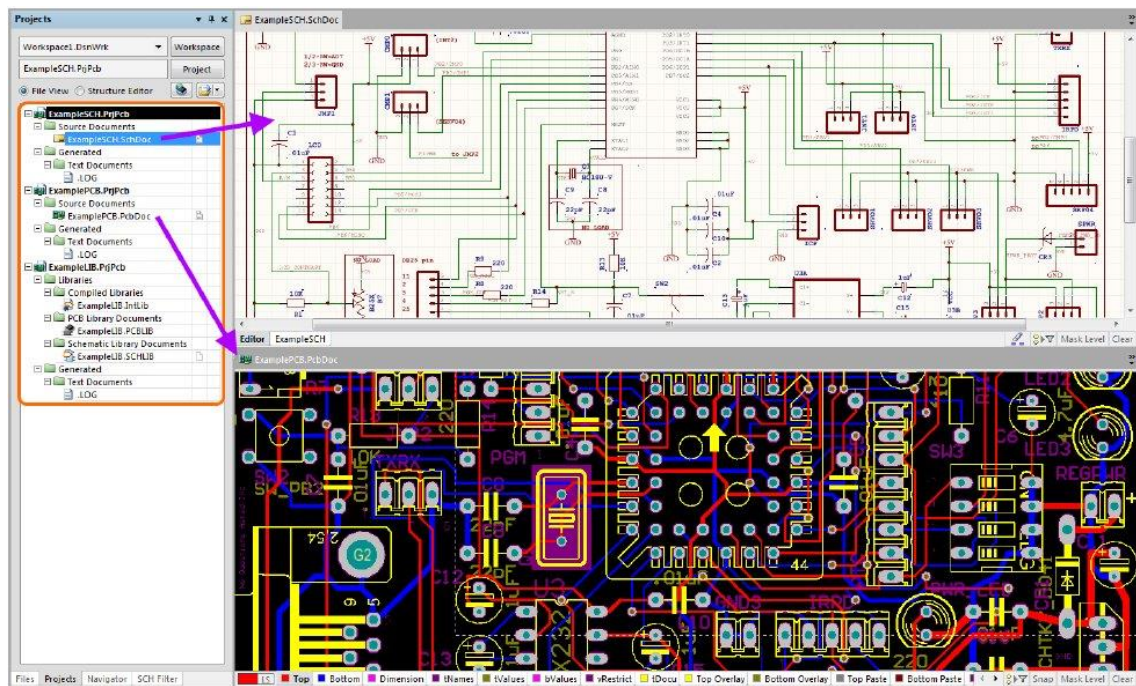


Рисунок 1.5 - Програма Altium Designer [10]

Редактор схем дозволяє вводити багатоієрархічні та багатоканальні схеми будь-якої складності, а також проводити змішане цифро-аналогове моделювання. Бібліотеки програми містять понад 90 тисяч готових компонентів, у багатьох з яких є моделі посадочних місць, SPICE і IBIS-моделі, а також тривимірні моделі. Будь-яку з перерахованих вище моделей можна створити внутрішніми засобами програми [11].

Програма Altium Designer дозволяє не тільки використовувати свої моделі компонентів, але і створювати нові, унікальні, що дуже корисно при розробці нових схем, і стане в нагоді не тільки в освіті, а й у науці.

1.2.2 Micro-Cap

Micro-Cap - професійна програма аналогового, цифрового та змішаного моделювання та аналізу ланцюгів електронних пристроїв середнього ступеня складності (рис. 1.6).

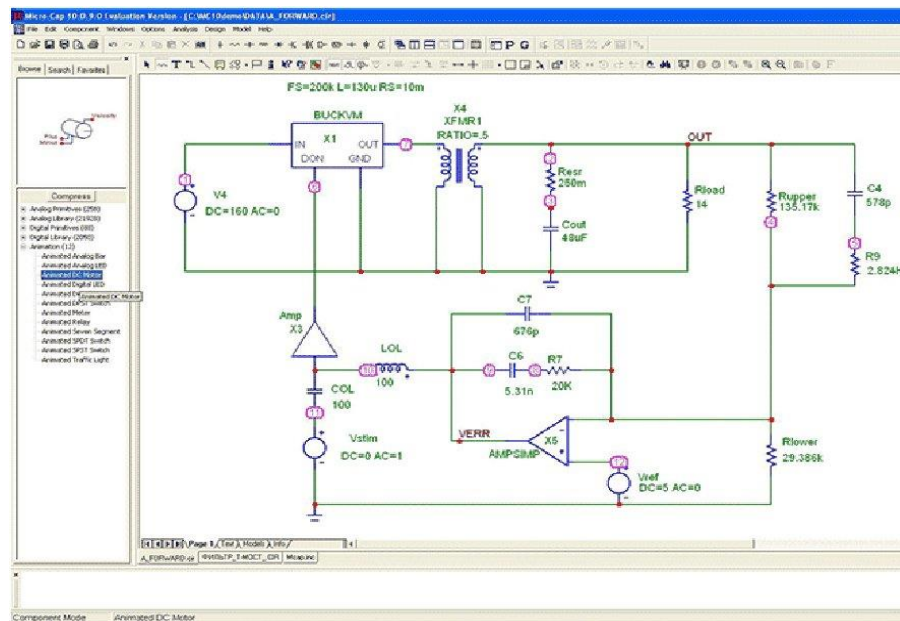


Рисунок 1.6 - Робоче вікно програми Micro-Cap

Інтуїтивно зрозумілий інтерфейс, невимогливість до обчислювальних ресурсів персонального комп'ютера і великий спектр можливостей послужили основою популярності Micro-Cap серед радіоаматорів, студентів і викладачів мікроелектроніки. Алгоритм роботи включає в себе створення електричного кола в графічному редакторі, завдання параметрів аналізу і вивчення отриманих даних (рис. 1.7). Програма самостійно складає рівняння кола і проводить моментальний розрахунок. Будь-яка зміна схеми або параметрів елементів призводить до автоматичного оновлення результатів. Графічний редактор спирається на бібліотеки електронних компонентів, які можна поповнювати на основі експериментальних або довідкових даних за допомогою вбудованого модуля Shape Editor. Всі номінали та параметри елементів можуть бути як постійними, так і залежними від температури, часу, частоти, стану схеми, параметрів інших компонентів. Анімовані деталі (світлодіоди, реле, семисегментний індикатори і деякі інші елементи) змінюють стан відповідно до надходять на них сигналами. Моделювання включає в себе цілий набір різних аналізів: перехідних процесів, передавальних характеристик по постійному струму, малосигнальних частотних характеристик, чутливості по постійному

струму, нелінійних спотворень, методу Монте-Карло і багатьох інших. Досвідчені користувачі можуть створювати свої макромоделі, які полегшують імітаційне моделювання без втрат інформації. Допускається одночасно використовувати різні стандарти елементів схеми. Повна підтримка SPICE-моделей дозволяє застосовувати проекти з інших програм (DesignLab, OrCAD, P-CAD). З недоліків можна відзначити лише необхідність установки додаткових елементів, так як обсяг бібліотек Micro-Cap (Навіть в повній версії) явно недостатній.

Програма була написана в 1982 році фірмою Spectrum Software, з тих пір вона постійно розширюється і вдосконалюється. Фірма, в свою чергу, була заснована Енді Томпсоном в лютому 1980 року, спочатку позиціонуючись на написанні програм для Apple. Вона розташована в одному з основних міст Силіконової долини – Саннівейл (Штат Каліфорнія, США).

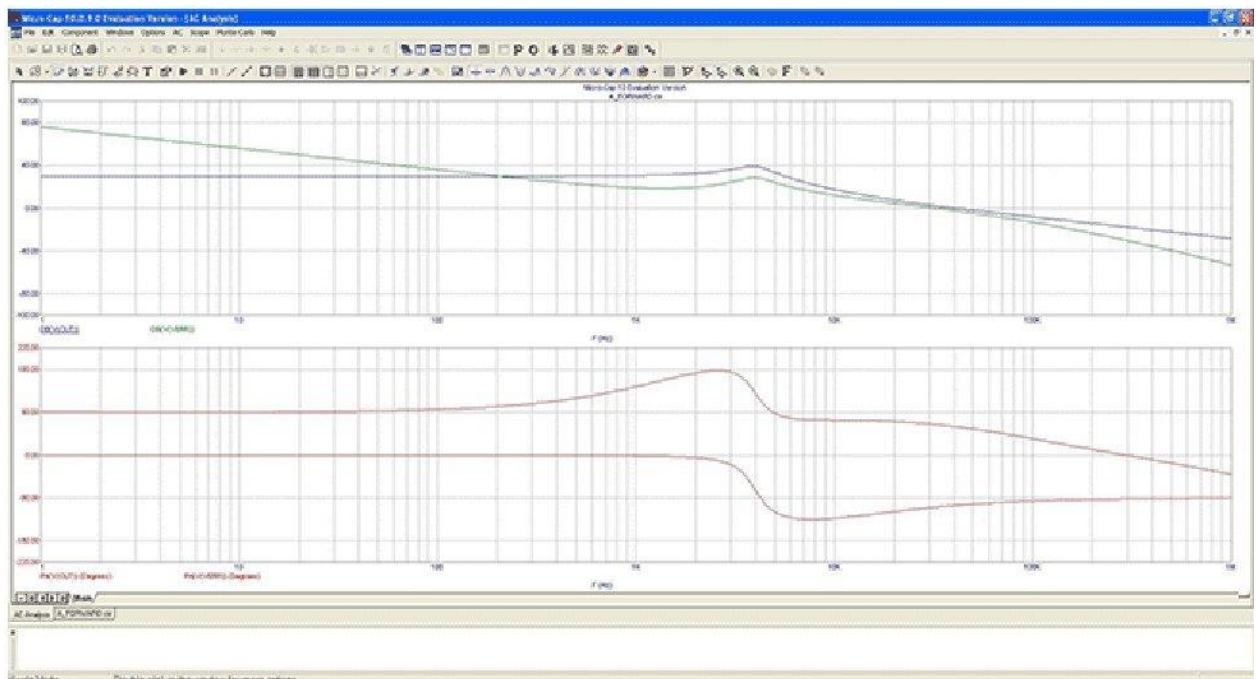


Рисунок 1.7 - Графічне представлення аналізу схеми

Вартість Micro-Cap становить кілька тисяч доларів, проте на сайті розробника можливо скачати вільно поширювану Evaluation Version, яка

володіє багатьма можливостями повнофункціональної. Основні відмінності - це не більше 50 елементів в схемі, урізана бібліотека компонентів, обмеження на побудову ряду графіків і повільна швидкість роботи [12].

1.2.3 MATLAB

Крім програм, що моделюють роботу електронних схем, існує безліч ПО для вирішення обчислювальних завдань. Яскравим прикладом такої програми є MATLAB.

MATLAB - високорівнева мова та інтерактивне середовище для програмування, чисельних розрахунків і візуалізації результатів (рис. 1.8). За допомогою MATLAB можна аналізувати дані, розробляти алгоритми, створювати моделі і додатки.

Мова, інструментарій та вбудовані математичні функції дозволяють досліджувати різні підходи і отримувати рішення швидше, ніж з використанням електронних таблиць або традиційних мов програмування, таких як C / C++ або Java.

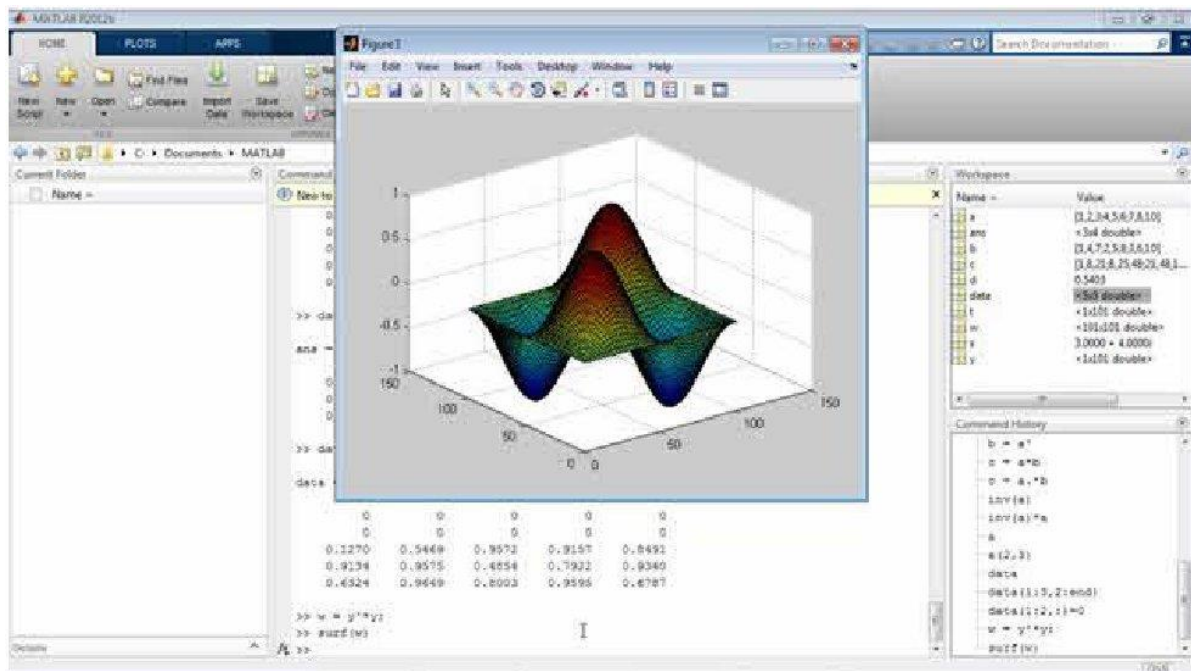


Рисунок 1.8 – Інтерфейс програми MATLAB

MATLAB широко використовується в таких областях, як:

- обробка сигналів і зв'язок;
- обробка зображень і відео;
- системи управління;
- автоматизація тестування і вимірювань;
- фінансовий інжиніринг;
- обчислювальна біологія і тому подібне.

MATLAB являє собою основу всього сімейства продуктів MathWorks і є головним інструментом для вирішення широкого спектра наукових і прикладних задач, в таких областях як: моделювання об'єктів і розробка систем управління, проектування комунікаційних систем, обробка сигналів і зображень, вимірювання сигналів і тестування, фінансове моделювання, обчислювальна біологія та інші.

Доступні наступні операції:

- 1) інтерполяція і регресія;
- 2) диференціювання та інтегрування;
- 3) системи лінійних рівнянь;
- 4) Фур'є аналіз;
- 5) власні значення і сингулярні числа матриць;
- 6) звичайні диференціальні рівняння;
- 7) розріджені матриці.

Розширення MATLAB надають спеціалізований функціонал в таких областях як статистика, оптимізація, обробка сигналів, машинне навчання.

Ядро MATLAB дозволяє максимально просто працювати з матрицями реальних, комплексних і аналітичних типів даних і зі структурами даних і таблицями пошуку.

MATLAB містить вбудовані функції лінійної алгебри (LAPACK, BLAS), швидкого перетворення Фур'є (FFTW), функції для роботи з поліномами, функції базової статистики та чисельного рішення диференціальних рівнянь; розширені математичні бібліотеки для Intel MKL.

Ключовими особливостями MATLAB є:

- платформи-незалежна високорівнева мова програмування, орієнтована на матричні обчислення і розробку алгоритмів;
- інтерактивне середовище для розробки коду, управління файлами та даними;
- функції лінійної алгебри, статистики, аналіз Фур'є, рішення диференціальних рівнянь та ін;
- багаті засоби візуалізації, 2-D і 3-D графіка;
- вбудовані засоби розробки користувацького інтерфейсу створення закінчених додатків на MATLAB[13].

Величезним достатком MATLAB є те, що широкий функціонал даної програми дозволяє створити абсолютно будь-яку модель, яку може побажати користувач.

1.2.4 Висновок

Дистанційні лабораторні роботи, засновані на застосуванні систем моделювання, надають можливість не тільки наочно продемонструвати принципи роботи пристроїв, але і проводити безліч дослідів без шкоди для реальних пристроїв.

Однак вони також, як і програми - симулятори, лише схематично за алгоритмами описують відбуваються на пристрої процеси, моделювання електронних схем не може абсолютно точно повторити роботу реального пристрою. Студент не має можливості спостерігати ні сигналів, ні контактів, хоча випускники технічних і природничих спеціальностей повинні мати досвід роботи з реальними об'єктами і явищами, а не з їх ідеалізованими моделями. До того ж більшість систем моделювання не є навчальними. Вони мають занадто багато функцій, що значно затягує процес освоєння програми, і вимагають серйозної попередньої підготовки користувача.

1.3 Віртуальні лабораторії

На сучасному етапі розвитку науки і техніки широкого поширення набули, так звані, автоматизовані віртуальні лабораторії з віддаленим доступом. В даному випадку під цим визначенням мається на увазі програмний комплекс, що дозволяє проводити досліди з допомогою моделі, що замінює реальний пристрій. Іншими словами, справжня лабораторна установка замінена на відповідну модель, а всі процеси управління моделюються за допомогою комп'ютера [14].

Найчастіше такі віртуальні лабораторії застосовуються в якості навчального матеріалу для школярів середніх і старших класів з таких предметів, як фізика, хімія, біологія, математика та інші. Прикладом такої лабораторії є VirtuLab [15].

VirtuLab (Віртуальні освітні лабораторії) - це освітні інтерактивні роботи, які дозволяють учням проводити віртуальні експерименти з фізики, хімії, біології, екології та інших предметів, як в тривимірному просторі, так і в двовірному. Інтерактивні уроки для школярів дозволяють розповісти і показати про будь-яких явищах природи просто і зрозуміло. Ефективне застосування інтерактивних тестів і уроків в освітньому процесі сприяє не тільки підвищенню якості шкільної освіти, а й економії фінансових ресурсів, створюють безпечну, екологічно чисте середовище.

На рис. 1.9 зображено приклад віртуальної лабораторної роботи в Virtulab по хімії «Витіснення одного металу іншим з розчину солі».



Рисунку 1.9 – Лабораторна робота «Витіснення одного металу іншим з розчину солі»

Застосування подібного роду лабораторій несе за собою ряд переваг [16].

1) Захоплюючі інтерактивні уроки та лабораторні роботи, можна проводити з дитиною в домашніх умовах по різних науках - фізика, біологія, хімія, екологія.

2) Віртуальні лабораторні роботи можна демонструвати в класі під час лекції як доповнення до лекційних матеріалів.

3) Проводити в комп'ютерному класі по мережі, з подальшим аналізом успішності учня.

4) Змінюючи параметри в інтерактивній лабораторії, користувач бачить зміни в 3D середовищі як результат своїх дій.

Дана лабораторія знаходиться у вільному доступі, дозволяючи учням виконувати завдання не тільки в класі, але і в домашніх умовах. У нижній частині робочого вікна транслюється рядок з описом дій, які необхідно виконати, щоб провести досвід. В кінці пропонується описати результат.

Істотним недоліком даної лабораторії є те, що викладач не має можливості відстежити результат роботи школяра. До того ж, як зазначалося раніше, VirtuLab розрахований на аудиторію шкільного віку і ніяк не може бути застосований для навчання студентів вищих навчальних закладів.

Проте існують віртуальні лабораторні роботи, спрямовані і на студентів технічних спеціальностей. Наприклад, університет Amrita Vishwa Vidyapeetham пропонує свої віртуальні лабораторії з таким областям науки і освіти, як:

- електроніка та зв'язок;
- комп'ютерні науки та техніка;
- електротехніка;
- машинобудування;
- хімічна інженерія;
- біотехнологія і біомедична інженерія;
- цивільне будівництво;
- фізичні науки;
- хімічні науки.

Дані віртуальні лабораторії дозволяють студентам не тільки отримувати знання на рівні бакалаврату, а й проводити наукові дослідження в цих областях. Перебуваючи за межами університету, студент має можливість отримувати базові знання з перерахованих вище областей.

Наприклад, в розділі фізичні науки можна знайти віртуальні лабораторні роботи, пов'язані з вивченням властивостей елементарних електричних ланцюгів (рис. 1.10).

Перед початком роботи користувачеві необхідно увійти в систему. У кожній лабораторній роботі є методичні вказівки для успішного виконання самої роботи.

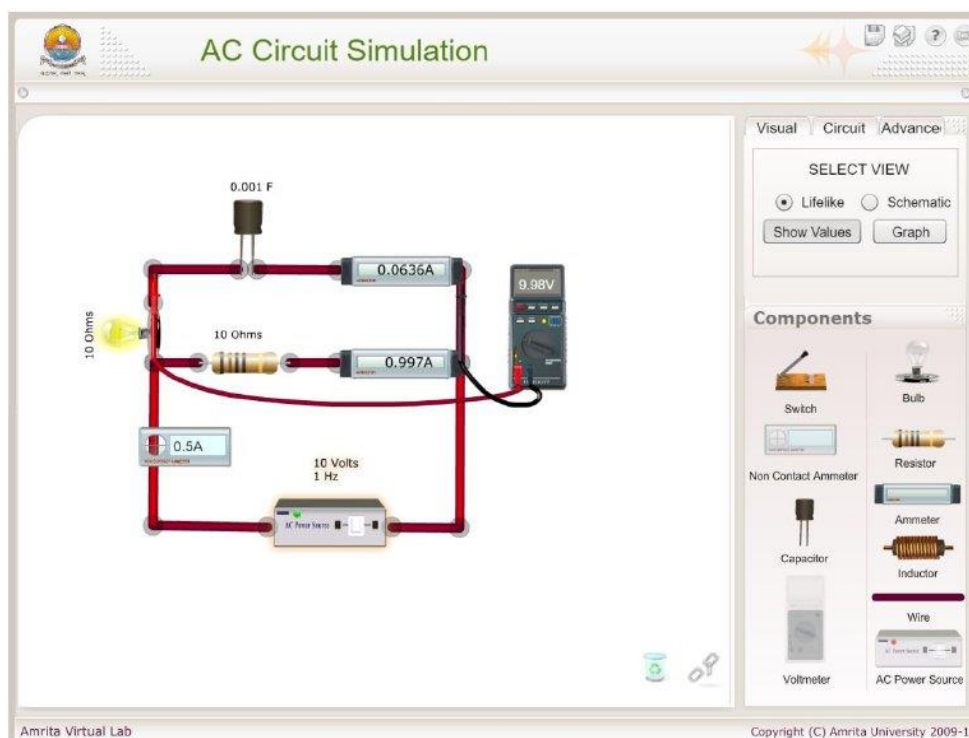


Рисунок 1.10 – Віртуальна лабораторна робота «Паралельні RC ланцюга» [17]

Відразу після виконання роботи можна пройти тестування і перевірити свої знання.

Недоліком віртуальних лабораторій університету Amrita є те, що інтерфейс виконаний повністю англійською мовою. Крім того, сам курс занадто обмежений, щоб застосовувати його для навчання студентів - інженерів старших курсів, а можливості реорганізувати його під необхідний навчальний план не передбачено. До того ж, незважаючи на те, що зовнішній вигляд досліджуваних пристроїв максимально наближений до реального, вони все одно є моделями. Майбутні інженери не можуть повноцінно вивчити властивості і принципи роботи пристроїв і приладів, засновані на виконанні лабораторних робіт з моделями, а не реальними пристроями.

Однак віртуальні лабораторії покликані виправити недоліки програм - симуляторів і систем моделювання. Даний вид віддалених лабораторних робіт дозволяє працювати з моделями дорогого устаткування, ставити наукові експерименти з будь-якої точки земної кулі.

Але студенти технічних спеціальностей повинні вміти працювати з реальними пристроями і сигналами. Тому в даній роботі пропонується розробити проект віртуальної лабораторії для студентів кафедри «Інформаційні технології електронних засобів» (ІТЕЗ) спеціальності «Радіотехніка та телекомунікації» з використанням реальної експериментальної установки, а не її моделі.

2 ВІРТУАЛЬНІ ЛАБОРАТОРІЇ З ВИКОРИСТАННЯМ РЕАЛЬНИХ ПРИСТРОЇВ

2.1 Визначення поняття віртуальна лабораторія з використанням реального пристрою

Перш ніж перейти до розробки віртуальної лабораторії з дисципліни «Електронні прилади», необхідно для початку розібратися, що саме мається на увазі під цим словосполученням.

Отже, віртуальна лабораторія з віддаленим доступом являє собою програмно-апаратний комплекс, що дозволяє проводити дослідження без безпосереднього контакту з реальною установкою. Інакше кажучи словами, це лабораторна установка з віддаленим доступом, до складу якої входить реальна лабораторія, програмно-апаратне забезпечення для управління установкою та оціфровки отриманих даних, а також засоби комунікації [18].

Віртуальні лабораторні роботи з віддаленим доступом вирішують більшу частину проблем, пов'язаних з реалізацією дистанційних лабораторних практикумів із застосуванням програм – симуляторів, систем моделювання або віртуальних лабораторій з комп'ютерними моделями, які замінюють реальні пристрої, наділяючи майбутніх інженерів необхідними практичними навичками і знаннями. Вони дозволяють працювати з реальним обладнанням, не піддаючи його будь-яким пошкодженням.

2.2 Класифікація віртуальних лабораторій з віддаленим доступом

Класифікувати лабораторії віддаленого доступу можна за кількома ознаками.

1) За типом інтерфейсу користувача:

а) додаток, для доступу до лабораторії необхідно завантажити додаток;

б) Web-сайт, доступ організований через веб-інтерфейс.

2) За типом доступу:

а) обмежені, для отримання доступу до віртуальних лабораторій у кожного користувача повинен бути свій логін і пароль;

б) частково-обмежені, у кожного користувача є свій логін і пароль, але також можливий гостьовий режим з обмеженим лабораторним практикумом;

в) вільні, немає логінів і паролів, тому і профілів у користувачів немає. В основному зустрічається в лабораторіях, навчальних базовим навичкам.

3) За вартістю:

а) платні, купується доступ до курсу на певний період;

б) частково платні, доступ до лабораторії безкоштовний тільки для студентів ВНЗ, в якому він розроблений, для всіх інших - платний;

в) умовно безкоштовні, безкоштовний доступ відкритий тільки для деяких лабораторних робіт, щоб відкрити інші, потрібно оплатити доступ;

г) безкоштовні.

4) Бази даних:

а) ведуться БД, щоб викладач мав можливість спостерігати за прогресом учнів, результат виконаних робіт записується в базу даних;

б) чи не ведуться БД, не можна відстежити результати виконаних робіт;

5) Наявність оператора:

а) з оператором, для проведення деяких віддалених лабораторних робіт з використанням реальних експериментальних установок необхідний оператор, який буде знаходитися в цей час в реальній лабораторії;

б) без оператора.

6) Реалізація відповідної реакції:

а) присутній, крім зміни стану на реальному пристрої, відповідна реакція дублюється і на екран монітора, наприклад, у вигляді текстового повідомлення;

б) відсутнє, реакцію пристрою на зміну будь-яких параметрів можна спостерігати тільки на відео зображенні реального пристрою в режимі реального часу.

7) Час роботи лабораторії:

а) обмежено, не кожна лабораторія може підтримувати можливість доступу до неї цілодобово, якщо реальна лабораторія розміщена в аудиторії навчального закладу, то час роботи віртуальної буде відповідати його режиму роботи;

б) необмежено, віртуальна лабораторія працює цілодобово.

8) Час доступу до одній установці:

а) обмежено, оскільки до однієї лабораторної установці в один момент може підключитися тільки одна людина, то допустимий час роботи з установкою може бути обмежена;

б) необмежено.

9) Методичні вказівки:

а) присутні, до лабораторного практикуму, крім інструкції, описує як саме працювати з цією установкою, додаються методичні вказівки, в яких позначений теоретичний матеріал по даній темі, приклади та завдання до даної лабораторної роботи;

б) відсутні.

10) Спрямованість:

а) загальні, найчастіше такі лабораторії навчають базовим знанням в певної теми;

б) спеціалізовані, створені під конкретний навчальний план конкретного ВНЗ.

Незважаючи на настільки різноманітну класифікацію, більшість лабораторій мають обмежений доступ, тому проаналізувати їх, а тим більше перебудувати їх для своїх цілей, не надається можливості [3].

Розглянемо кілька існуючих доступних рішень в даній області.

2.2.1 Дистанційна лабораторія GYMKT

Дистанційна Інтернет - лабораторія GYMKT була створена під керівництвом Комісії з фізики в чеських гімназіях Ярослава Врчліцького. Лабораторія включає в себе реальні фізичні експерименти, які можуть проводитися через браузер користувача. Дані віддалені експерименти можна проводити абсолютно безкоштовно і без будь-якої реєстрації. Виміряні дані можна завантажувати на персональний комп'ютер, де потім можна продовжити роботу з ними. Завдяки тому, що лабораторія доступна цілодобово, студенти можуть проводити експерименти як на заняттях в навчальному закладі, наприклад, в комп'ютерному класі, так і вдома в будь-який час доби. Лабораторний практикум може бути застосований не тільки в школах, але і в вищих навчальних закладах для навчання основам фізики і електроніки. Крім того, роботи, представлені в даній лабораторії, можуть бути використані для проведення експериментів, які складно реалізувати в інших закладах через складність реалізації досвіду, відсутність необхідного обладнання або його вартості.

Лабораторія GYMKT вже реалізувала наступні роботи:

- температура, залежність опору металів і напівпровідників;
- визначення горизонтальної складової земного магнітного поля;
- дистанційне керування роботизованою рукою;
- ВАХ світлодіодів;
- основні характеристики біполярного транзистора;
- вольтамперна характеристика лампи розжарювання та інші.

Розглянемо принцип роботи лабораторії на прикладі досліду «Температура, залежність опору металів і напівпровідників» (рис. 2.1).

Суть досвіду полягає в вимірі опору двох базових електронних компонентів - датчиків і платиного термістора з негативною характеристикою - при різних температурах. Для нагріву (або охолодження)

двох компонентів використовуються Пельтьє (термоелектричне явище, при якому відбувається виділення або поглинання тепла при проходженні електричного струму в місці контакту (спаяний) двох різнорідних провідників), вихід якого контролюється імпульсною модуляцією. Опір обох частин визначається операційним підсилювачем, а пара AD-перетворювачів перетвориться в цифрову форму. Поточне значення температури, відображене на дисплеї термометра, в реальному часі зчитується веб-камерою через USB.



Рисунок 2.1 – Зовнішній вигляд лабораторної установки «Температура, залежності опору металів і напівпровідників»

Системні операції, такі як зміна полярності комутаційних засобів елемента Пельтьє в режимі очікування, забезпечуються посиленням цифрових сигналів від керуючого комп'ютера. Пристрої зв'язку і ПК з'єднані через USB. В якості пристрою управління використовується універсальна USB-

експериментальна плата K8055. Час роботи з установкою обмежена, користувачеві дається 15 хвилин, але якщо в черзі більше нікого немає, то можна заново підключитися до установки, і результат роботи буде збережений.

Вікно користувача віртуальної лабораторної роботи «Температура, залежність опору металів і напівпровідників» представлено на рис. 2.2.

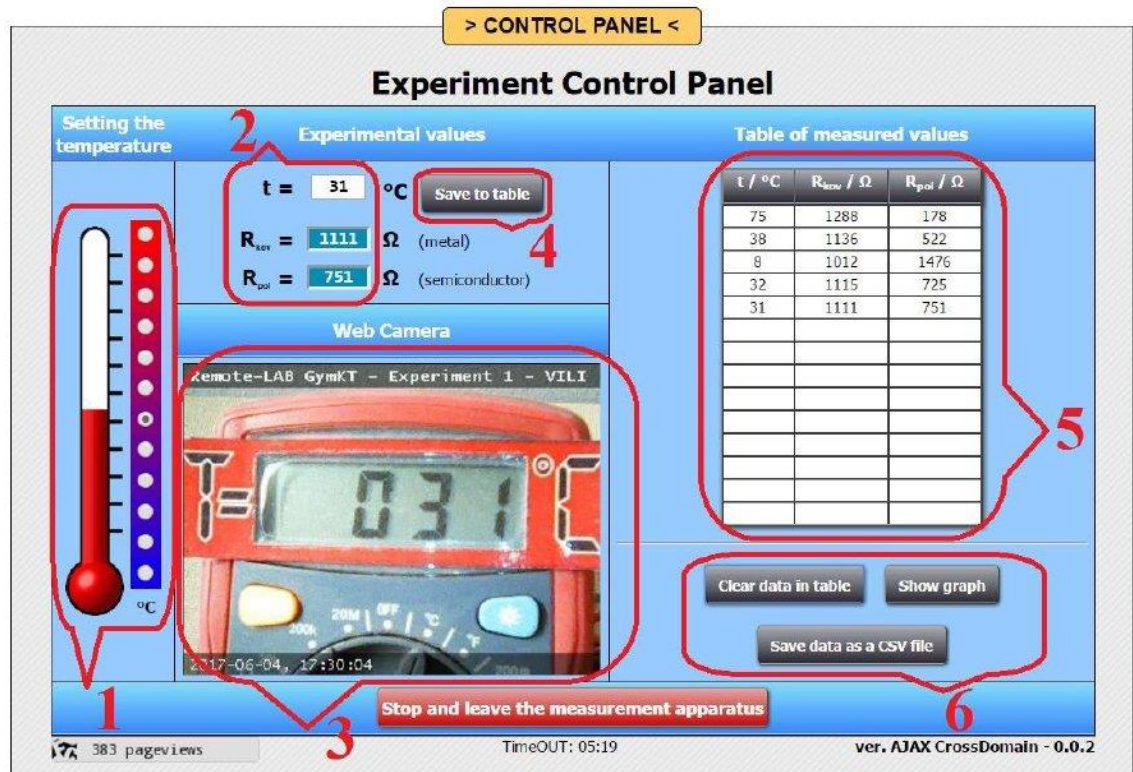


Рисунок 2.2 - Інтерфейс користувача [19]

Як показано на малюнку робоче вікно користувача розділене на 6 областей:

1) налаштування температури - температура регулюється за допомогою дванадцяти значень;

2) експериментальні значення - значення резисторів завантажуються автоматично, температура повинна зчитуватися з веб-камери;

3) зображення з веб-камери, камера спостерігає за відображенням цифрового термометра з температурою вимірюваного елемента;

4) запис значень в таблицю - виміряні значення заносяться в таблицю.

5) таблиця експериментальних значень - t , R_k і R_p ;

6) робота з даними в таблиці - є можливість подивитися графік залежності, зберегти дані в файл *.csv, видалити таблицю.

У порівнянні зі звичайними лабораторними роботами віддалена Інтернет - лабораторія GYMKT володіє декількома важливими перевагами [19]:

1) вільний доступ до лабораторії (в будь-який час і в будь-якому місці);

2) користувачеві не потрібні ніякі фізичні допомоги;

3) досвід може повторюватися кілька разів;

4) користувачі працюють з реальними вимірювальними приладами, дані вимірювань є реальними;

5) немає небезпеки отримання травми при роботі з небезпечним обладнанням;

6) може бути використана в якості домашньої роботи в традиційній навчальній лабораторній практиці;

7) сучасний підхід збільшує зацікавленість студентів;

8) економія часу для викладачів;

9) швидка графічна обробка вимірюваних значень.

Однак дана лабораторія не може бути використана для навчання студентів кафедри «Інформаційні технології електрон засобів» за спеціальністю «Радіотехніка та телекомунікації» з дисципліни «Електронні прилади», тому що запропоновані досліди підходять лише для вивчення найпростіших основ фізики і електроніки. Також не передбачена можливість перестроювання лабораторії під інший навчальний план. Крім того, недоліком є іноземний інтерфейс, неадаптованих для зміни на інші мови.

2.2.2 RELLE (Remote Labs Learning Environment)

RELLE (навчальне середовище віддалених лабораторій) є середовищем для маніпулювання і управління віддаленими лабораторіями.

RELLE розроблений робочою групою мобільних експериментів (GT-MRE) в Лабораторії віддалених експериментів (RExLab) в Федеральному університеті Санта-Катаріна в Бразилії. GT-MRE - це проект, спонсований Координацією щодо поліпшення якості персоналу вищої освіти (CAPES) і Бразильської національної науково-освітньої мережею (RNP) через тематичні програми досліджень і розробок. Всі віддалені лабораторії відкриті для будь-якого користувача.

RELLE включає в себе наступні лабораторні роботи:

- середовище розробки для Arduino: середовище, яке дозволяє перевіряти і завантажувати код, управляти датчиками і виконавчими механізмами в Arduino;
- способи поширення тепла: вивчення способів поширення тепла шляхом конвекції і випромінювання;
- віддалений мікроскоп: мікроскопія листових пігментів;
- похила площина: вивчення другого закону Ньютона і розкладання сили;
- диск Ньютона: вивчення колірної композиції;
- оптична лава: вивчення поведінки світла в найбільш поширених лінзах і різних форматах;
- теплопровідність в металевих стрижнях: ця лабораторія вивчає модель поширення тепла в металевих стрижнях;
- Block.ino: середовище розробки для Arduino з використанням візуальної мови.

Крім того, частина лабораторних практикумів знаходиться в розробці.

Розглянемо принцип роботи лабораторії на прикладі лабораторної роботи «Середовище розробки для Arduino» (рис. 2.3).

Час роботи з пристроєм обмежено в 10 хвилин.

Робоча область розділена на шість полів, кожне з яких виконує свої функції:

1) редактор функцій, у цьому редакторі є функції, аналогічні тим, які існують в Arduino IDE;

2) панель управління, користувач може скомпілювати і завантажити свій код в Arduino, а також відкрити, завантажити файли «.ino» і відкрити послідовний термінал;

3) меню, в якому знаходиться кілька прикладів і бібліотек, доступних для використання в коді;

4) модель схеми, за допомогою якої можна ідентифікувати датчики і контакти, які будуть використовуватися під час розробки програми;

5) зворотній зв'язок: результат компіляції і завантаження з'явиться в даному вікні, якщо вони були успішно виконані;

6) реальне відео реальної плати Arduino в режимі реального часу.



Рисунок 2.3 – Лабораторна робота «Середовище розробки для Arduino» RELLE

Для того щоб використовувати лабораторію, необхідно написати код, (можна використовувати приклад або завантажити вже готовий файл), скомпілювати його, завантажити в Arduino, спостерігати результат через відеопоток [20].

Крім віддалених віртуальних лабораторій середу RELLE включає в себе ще кілька курсів для перевірки своїх знань, але для їх проходження вже необхідно увійти до системи.

Однак, незважаючи на багатобічність лабораторні практикуми, RELLE не може бути використана для студентів ІТЕЗ напряму «Електронні апарати». У даній лабораторії не тільки не передбачено перестроювання лабораторних робіт, але і весь інтерфейс повністю англійською, або португальською мовою, що значно ускладнює процес навчання.

2.2.3 Висновок

Виходячи з усього вище перерахованого можна визначити, що необхідність в розробці та реалізації лабораторій і лабораторних практикумів, які повинні будуть відповідати навчальним планом підготовки студентів за спеціальністю «Радіотехніка та телекомунікації» кафедри ІТЕЗ Запорізького національного технічного університету все ще залишається актуальною. Незважаючи на різноманітність існуючих рішень в даній області, їх вбудовування в необхідний освітній процес зовсім не надається можливим, тому що велика частина з них створені виключно для певного ВНЗ або навчального плану і мають ряд недоліків, які можна уникнути, реалізувавши власну віртуальну лабораторію.

У зв'язку з цим було висунуто ряд загальних критеріїв, яким повинна відповідати віртуальна лабораторія, розроблена для отримання навичок роботи з різними електронними схемами відповідно до навчального плану з дисципліни «Електронні прилади».

1) Глобальний доступ. Користувач повинен мати можливість отримувати доступ з будь-якої точки планети через web-інтерфейс.

2) Інтуїтивно зрозумілий інтерфейс.

3) Наявність системи аутентифікації. Доступ мають тільки зареєстровані студенти навчального закладу. У кожного користувача є свій пароль. Є можливість реєстрації сторонніх користувачів, але за певну плату.

4) Ведення баз даних зі збереженням результатів роботи і можливістю подальшої перевірки викладачем.

5) Відсутність оператора. Під час виконання лабораторних робіт немає необхідності в допомозі оператора.

6) Реалізація відповідної реакції з навчальної плати у вигляді повернення відповіді з мікроконтролера, який зчитав інформацію з навчальної плати.

7) Наявність методичних вказівок для виконання лабораторної роботи.

8) Час роботи лабораторії. Доступ до лабораторії не може бути цілодобовим, він залежить від фізичного часу лабораторної аудиторії або навчального закладу.

9) Час доступу до однієї установці. Розробляється розклад, в якому прописано час роботи кожного користувача з навчальною платою.

10) Наявність зворотного зв'язку. При виникненні будь-яких питань по ходу виконання завдань студент повинен мати можливість отримати роз'яснення по виниклих труднощах.

11) Наявність опису принципу роботи віртуальних лабораторних робіт.

Проаналізувавши готові віртуальні лабораторії було вирішено, що загальна структурна схема віртуальної лабораторії з віддаленим доступом з дисципліни «Електронні прилади» повинна виглядати наступним чином (рис. 2.7).

Принцип роботи полягає в наступному.

1) Клієнт отримує доступ до комплексу віртуальних лабораторних робіт через Інтернет - з'єднання. Це може або вільний, або обмежений доступ. Інтерфейс представлений у робочого столу комп'ютера. Далі на екрані користувача з'являється область для роботи. Вона представляє із себе панель віртуального управління пристроєм, в якому знаходяться: керуючі елементи,

блок - діаграма. Виконуючи будь-які завдання, клієнт на екрані може спостерігати результати змін стану навчальної плати.

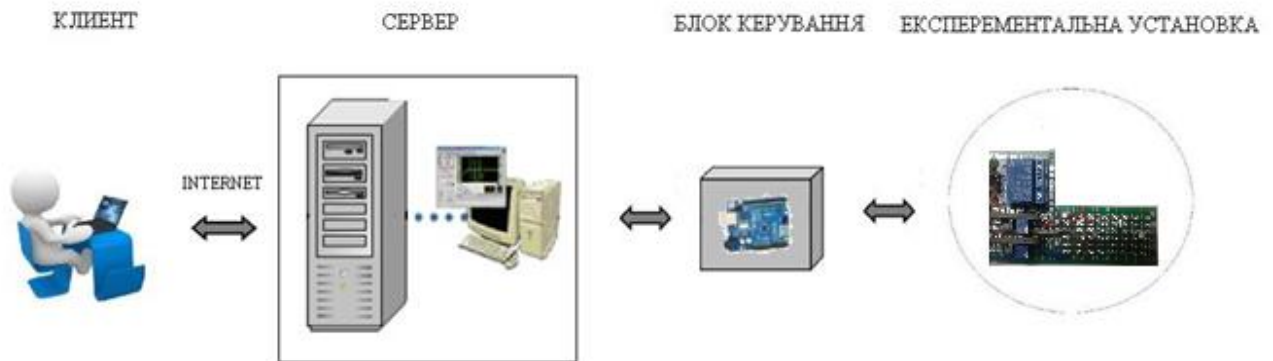


Рисунок 2.7 – Загальна структурна схема віртуальної лабораторії з віддаленим доступом

2) Всі зміни, що здійснюються користувачем, потрапляють на сервер через Інтернет - з'єднання.

3) Потім інформація про змінені параметри надходить у керуючий блок. Даний блок обробляє її в зв'язку з встановленими принципами роботи реального пристрою і передає сигнал на експериментальну установку.

4) Експериментальна установка являє собою навчальну плату підключену до керуючого блоку. Клієнт змінює стан перемикачів на екрані монітора, і відповідно до принципів роботи реального пристрою відбуваються відповідні зміни також на екрані користувача блок - діаграма відображає внесенні зміни.

Виходячи з усього вище перерахованого, можна виділити завдання, які необхідно вирішити для розробки віртуальної лабораторії:

- розробка програмного комплексу;
- розробка апаратного комплексу;
- розробка методології виконання лабораторних робіт з використанням розробленого програмно-апаратного комплексу.

3 ПРОЕКТУВАННЯ АПАРАТУРИ ЗАБЕЗПЕЧЕННЯ ДЛЯ ПРОВЕДЕННЯ ДОСЛІДЖЕНЬ

Віртуальна лабораторія з дисципліни «Електронні прилади» дозволить майбутнім інженерам отримати навички роботи з реальними електронними схемами в будь-який час, перебуваючи за межами навчального закладу.

Структурна схема даної лабораторії представлена на рис. 3.1.



Рисунок 3.1 - Структурна схема віртуальної лабораторії

Як видно зі схеми віртуальна лабораторія складається з трьох модулів:

- експериментальна установка;
- керуючий пристрій;
- інтерфейс користувача.

З точки зору розробки проектування лабораторії складається з двох рівнів:

- 1) апаратний;
- 2) програмний.

Розглянемо кожен з них детальніше.

3.1 Апаратна частина лабораторії

З боку апаратної реалізації лабораторія складається з експериментальної установки та керуючого пристрою.

3.1.1 Експериментальна установка

Експериментальна установка - це реальна навчальна плата. На даній платі реалізовано пристрій для зняття вольт-амперних характеристик фото приладів (фотодіода), яке відповідає одному із завдань методичних вказівок для виконання лабораторних робіт з дисципліни «Електронні прилади» (рис. 3.2), перетворена для цифрового управління .

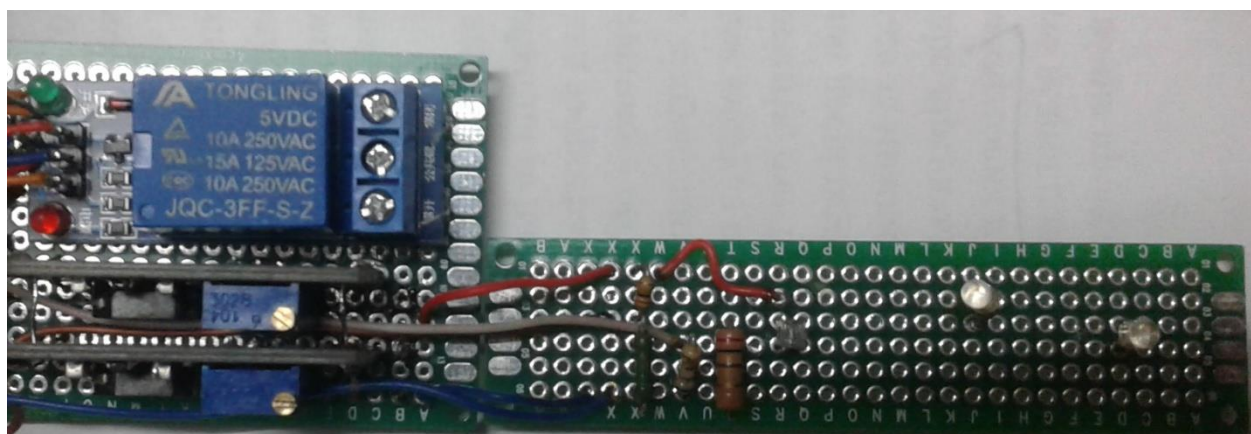


Рисунок 3.2 - Реальна навчальна плата для виконання лабораторних робіт з дисципліни «Електронні прилади»

Електрична принципова схема перетвореної реальної навчальної плати представлена на рис. 3.3.

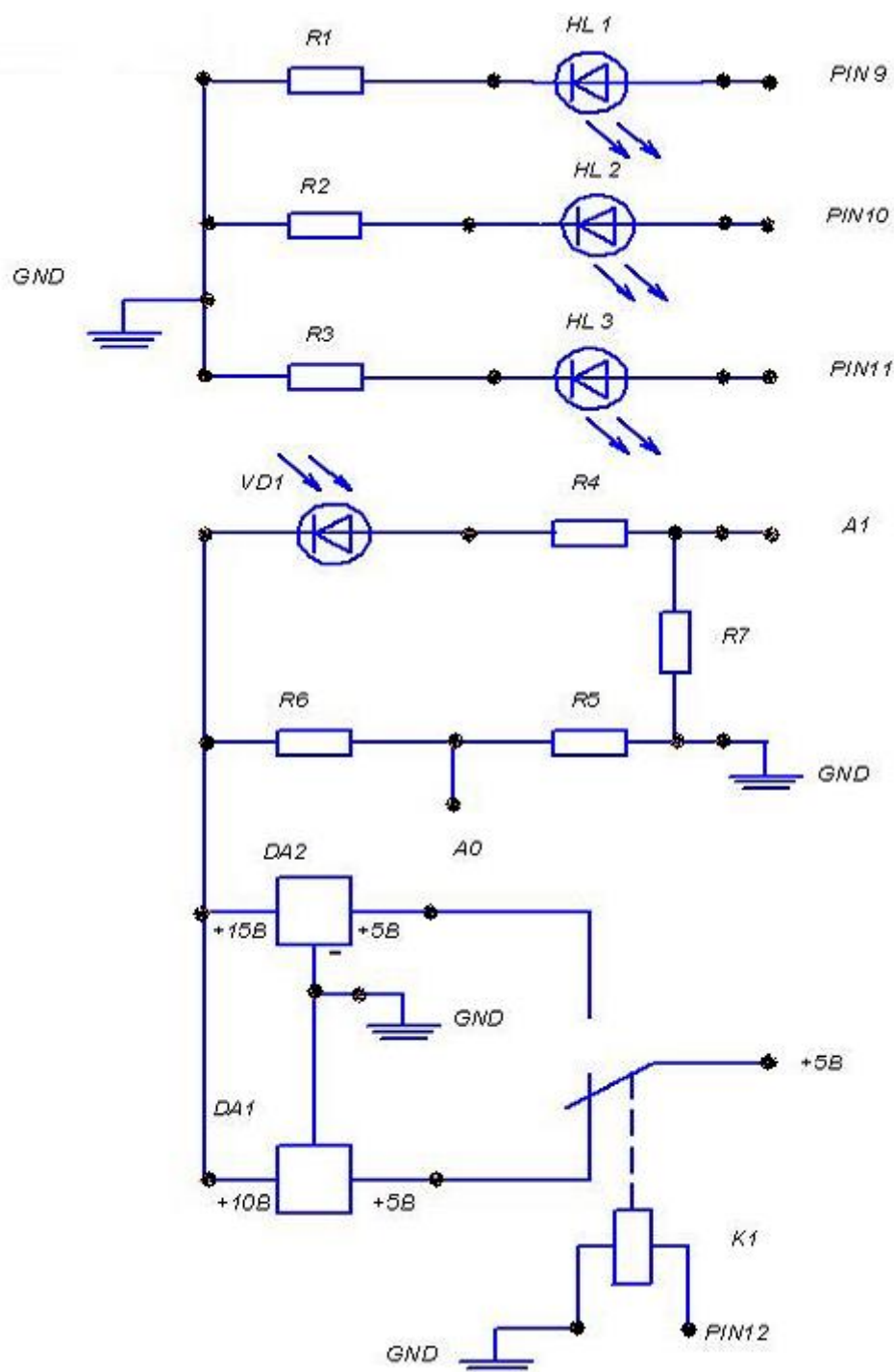


Рисунок 3.3 - Електрична принципова схема перетвореної реальної навчальної плати

На платі встановлені два послідовно з'єднаних резистора R5 та R6, які виконують функцію дільника напруги. Показання знімаються з точки з'єднання між ними. Також R7 використаний в якості «шунту» для зняття сили струму. В якості регулюючого блоку живлення використовується модуль реле K1 PIC ARM AVR (електрична принципова схема на рис. 3.4) та регульовані

перетворювачі напруги DA1 та DA2 (електрична принципова схема на рис. 3.5). Модуль реле K1 використовується для перемикання. Для включення першого або другого перетворювача напруги, зазначених на платі як DA1 та DA2 відповідно, на них подається напруга 5 В з плати Arduino UNO на вхід. На виході перетворювачів встановлена напруга: на першому 10 В, на другому 15 В. R4 - навантажувальний опір для фотодіода VD1. HL1, HL2 та HL3 - світлодіоди, які знаходяться на різній відстані від фотодіода VD1 та вмикаються окремо. Резистори R1, R2 та R3 є обмежувальними для світлодіодів. Виводи (Pin 9, Pin10, Pin11, Pin12, A0, A1, +5V та GND), підключені на плату Arduino UNO відповідно найменуванню виводів. Перелік елементів, необхідних для виготовлення перетвореної реальної навчальної плати, представлений у табл. 3.1. Усі використані резистори було узято з урахуванням більш точного обчислення та калібрування пристрою.

Таблиця 3.1 - Перелік елементів

Позначення	Найменування	Кількість	Примітка
Резистори			
R1,R2,R3	C2-33-100 Ом $\pm$ 5% 0,125 Вт	3	
R4	C2-33-10 кОм $\pm$ 5% 0,125 Вт	1	
R5	C2-33-10 кОм $\pm$ 5% 0,125 Вт	1	
R6	C2-33-100 кОм $\pm$ 5% 0,125 Вт	1	
R7	C2-33-10 Ом $\pm$ 5% 0,125 Вт	1	
Світлодіоди			
HL1, HL2 ,HL3	C503C-WAN	3	
Фотодіод			
VD1	HPDB1B-48D	1	
Покупні деталі			
DA1, DA2	Перетворювач напруги mt3608	2	
K1	Модуль реле 1-канальний 5V PIC ARM AVR	1	

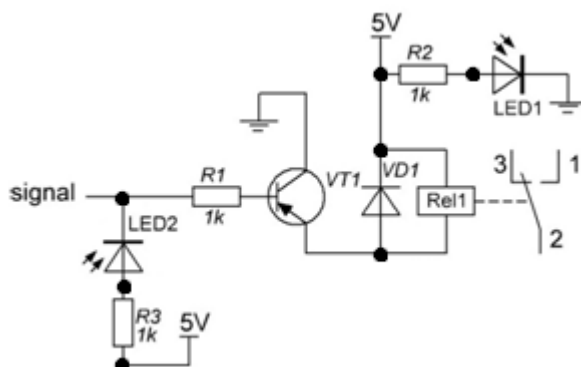


Рисунок 3.4 – Електрична принципова схема модуль реле 1-канальний 5В PIC ARM AVR

Параметри модуля реле:

- VCC - інтерфейс живлення - 5 В;
- GND - земля;
- струм приводу не менше - 4 мА;
- COM: Заземлення кола управління, підключення до GND;
- розмір: приблизно 41x16x16 мм / 1,61x0,62x0,62 "
- вага: приблизно - 12 г

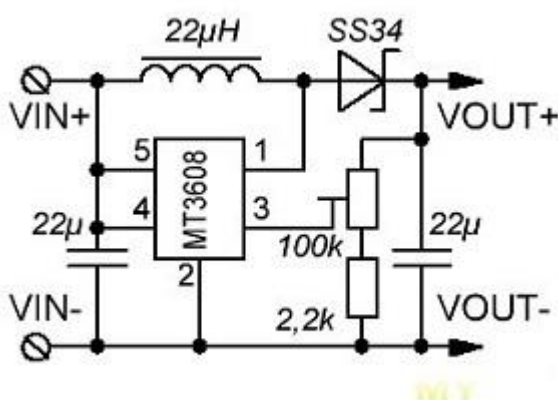


Рисунок 3.5 - Електрична принципова схема перетворювача mt3608

Характеристики перетворювача mt3608:

- максимальний вихідний струм: 2 А;

- вхідна напруга: 2 В ~ 24 В;
- максимальна вихідна напруга: 28 В;
- ефективність: макс 93%;
- розмір продукту: 36мм * 17мм * 14мм;
- регулюючи потенціометр можна налаштувати вихідну напругу, вихідна напруга більше вхідної напруги;
- вхідна напруга не повинна перевищувати максимальну вихідну напруги.

Дана схема (рис. 3.3) містить в собі завдання лабораторної роботи «Дослідження фотопровідності напівпровідників та фотоелектричних властивостей р-n переходу». Реальна навчальна плата зроблена згідно експериментальної установки. У цьому завданні від студента вимагається зняти вольтамперні характеристики та світлові характеристики фотодіода. Плата та керуючий пристрій вміщені у світлонепроникний корпус.

3.1.2 Керуючий пристрій

Керуючий пристрій пов'язує користувача з експериментальною установкою. Інакше кажучи, це блок управління навчальною платою, що перетворює цифрові сигнали, які подає користувач, в аналогові, які відображаються на навчальній платі в результаті виконуваних користувачем дій.

До складу керуючого пристрою входить:

- керуюча плата з мікроконтролером;
- керуючий ПК з ПЗ для роботи з підключеною керуючою платою через порти USB.

Керуюча плата з мікроконтролером - пристрій управління навчальною платою, який відповідно до заданими значеннями сигналів на вході виставляє відповідні даному типу навчальної плати значення на виходах та входах. Та подає живлення на реальну плату. У якості керуючої плати з мікроконтролером,

проаналізувавши кількість необхідних для роботи портів, було вирішено використовувати плату Arduino UNO.

Плата Arduino UNO – це пристрій на основі мікроконтролера ATmega328 (рис. 3.6, рис. 3.7, рис. 3.8, рис. 3.9). До його складу входить все необхідне для зручної роботи з мікроконтролером: 14 цифрових входів / виходів (з них 6 виводів можуть використовуватися в якості ШІМ - виходів), 6 аналогових входів, кварцовий резонатор на 16 МГц, роз'єм USB, роз'єм живлення, роз'єм для внутрішньосхемного програмування (ICSP) і кнопка скидання. Для початку роботи з МПС необхідно подати живлення від AC/DC-адаптера, або підключити його до комп'ютера за допомогою USB- кабелю. Для роботи з платою Arduino UNO в операційній системі Windows необхідно встановити на комп'ютер інтегроване середовище розробки Arduino – Arduino IDE (Integrated Development Environment).



Рисунок 3.6 - Плата Arduino UNO

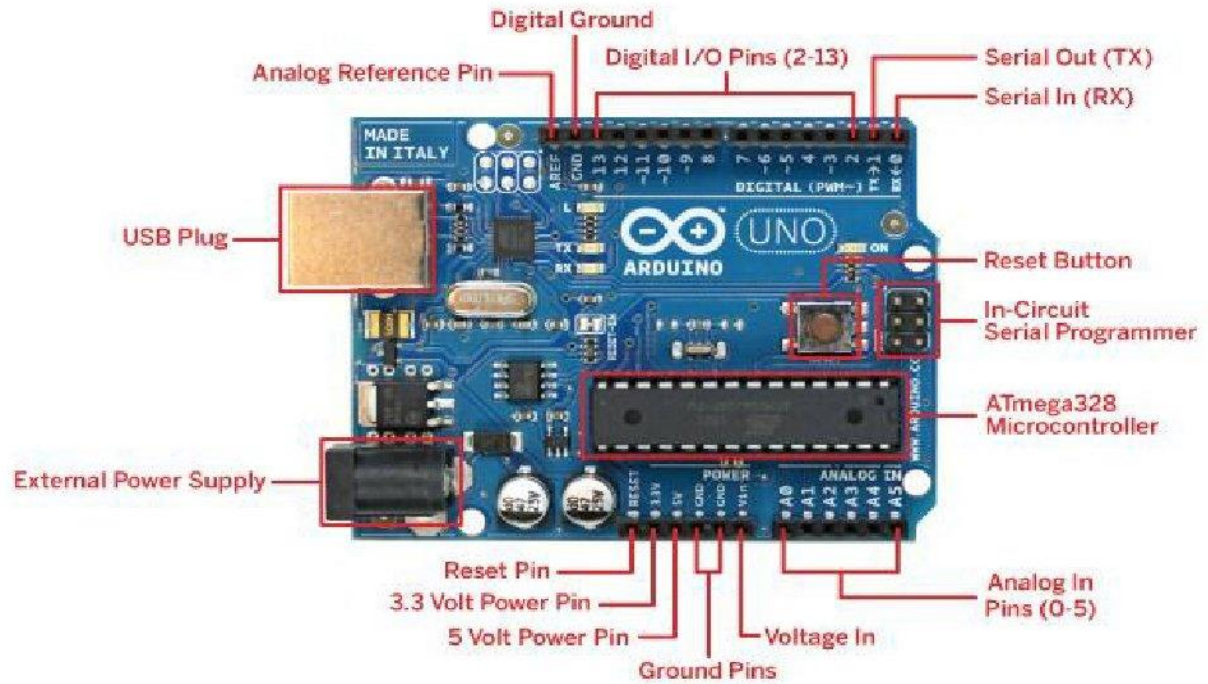


Рисунок 3.7 - Загальний опис елементів плати Arduino UNO

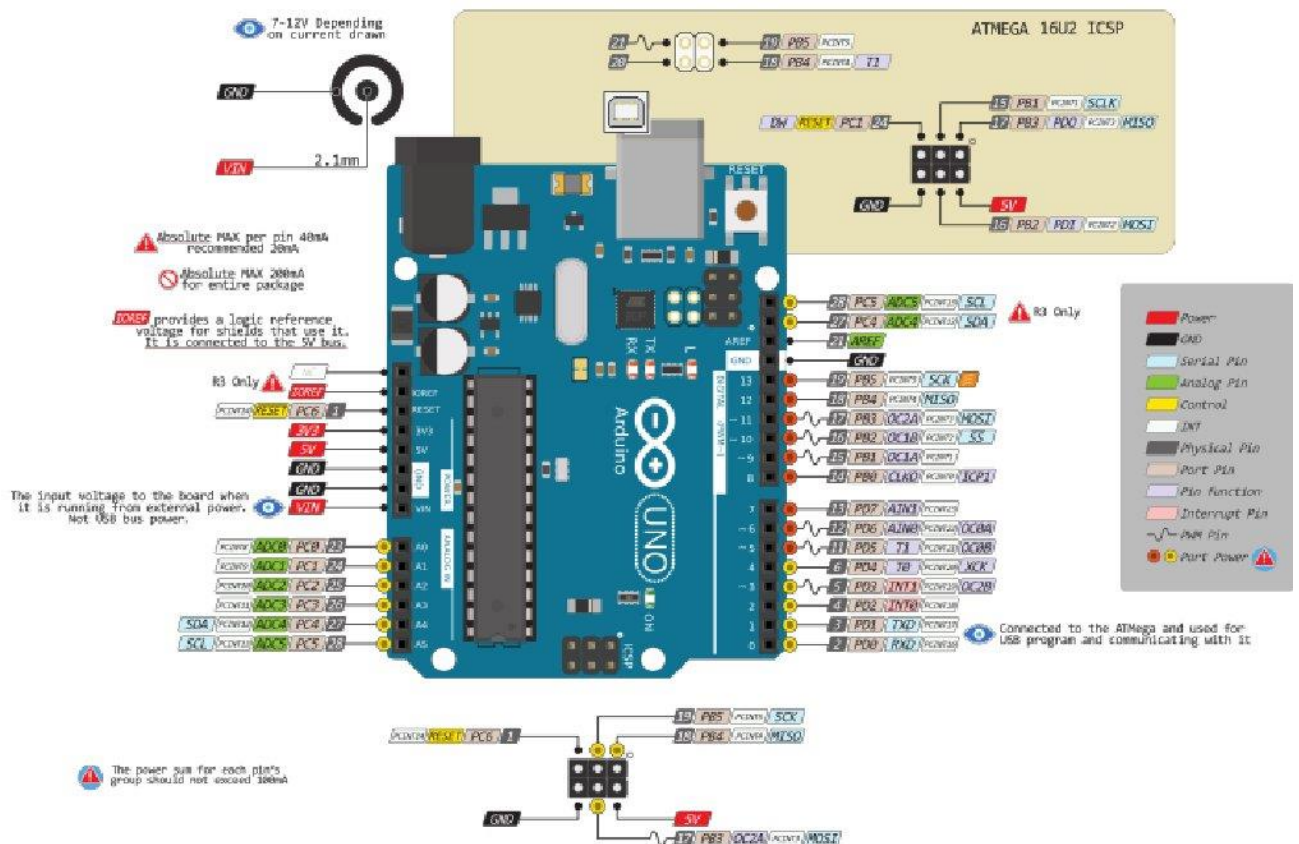


Рисунок 3.8 - Загальне призначення виводів плати Arduino UNO

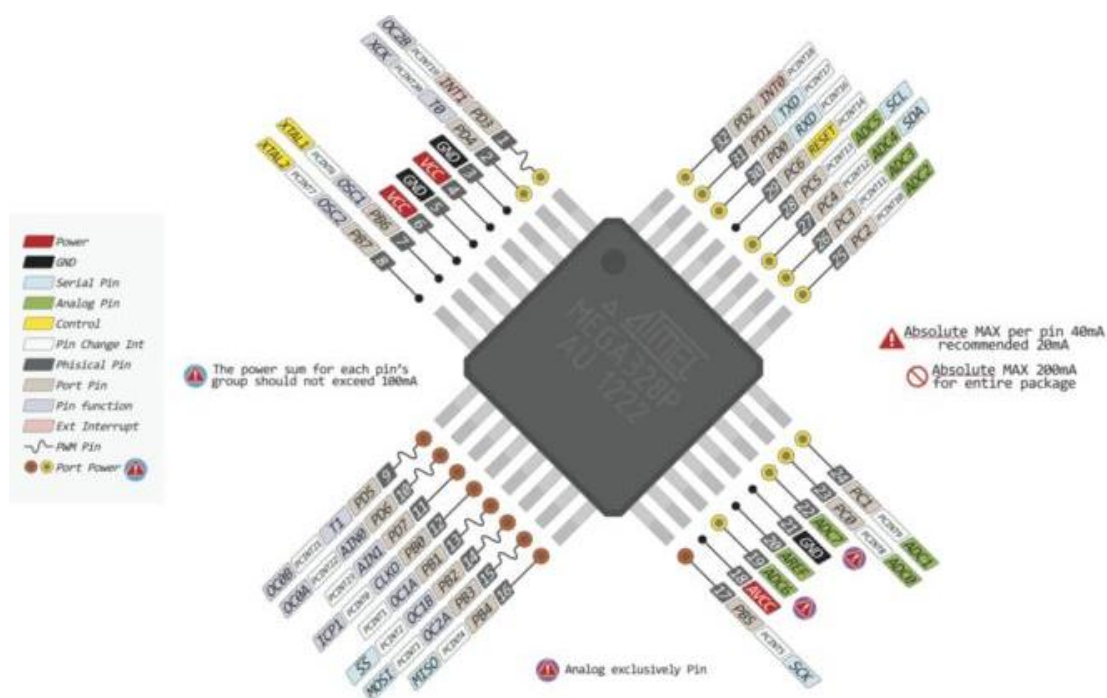


Рисунок 3.9 - Призначення виводів мікроконтролера ATmega328

Характеристики плати Arduino UNO:

- мікроконтролер - ATmega328;
- робоча напруга - 5В;
- напруга живлення (рекомендована) - 7-12В;
- напруга живлення (гранична) - 6-20В;
- цифрові входи / виходи - 14 (6 у якості ШІМ-виходів);
- аналогових входів – 6;
- максимальний струм одного виведення - 40 мА;
- максимальний вихідний струм виводу - 3.3В 50 мА;
- Flash-пам'ять - 32 КБ (ATmega328) з яких 0.5 КБ загрузчик;
- SRAM - 2 КБ (ATmega328);
- EEPROM - 1 КБ (ATmega328);
- тактова частота - 16 МГц.

Входи і виходи плати Arduino UNO:

- Послідовний інтерфейс: виходи 0 (RX) і 1 (TX), що використовуються для отримання (RX) і передачі (TX) даних по послідовному інтерфейсу. Ці виводи з'єднані з відповідними виводами мікросхеми ATmega16U2 на платі Arduino UNO, яка виконує роль перетворювача USB-UART.

- Зовнішні переривання: виводи 2 і 3, які можуть служити джерелами переривань, що виникають при фронті, спаді або при низькому рівні сигналу на цих виводах.

- ШІМ-виводи 3, 5, 6, 9, 10 і 11 – інтерфейс SPI: виводи 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK).

- Світлодіод: 13 представляє вбудований світлодіод, приєднаний до вивода 13. При відправці значення HIGH світлодіод включається, при відправці LOW - вимикається.

- 6 аналогових входів (A0 - A5), кожен з яких може уявити аналогову напругу у вигляді 10-бітного числа (1024 різних значення). За замовчуванням, вимір напруги здійснюється щодо діапазону від 0 до 5 В.

На платі Arduino UNO розташовані ще кілька додаткових виводів.

- Reset: формування низького рівня (LOW) на цьому виводу призведе до перезавантаження мікроконтролера. Зазвичай цей висновок служить для функціонування кнопки скидання на платах розширення.

- 3V3: напруга на даному виводі +3.3 В, що генерується вбудованим регулятором на платі. Максимальне споживання струму 50 мА. Від цього виводу можуть житися деякі апаратні модулі.

- 5V: напруга на даному виводу +5 В, що генерується вбудованим регулятором на платі Arduino UNO.

- GND: земля, або загальний мінус. Цей вивід є другим обов'язковим виводом для будь-якого іншого пристрою.

- Vin: використовується для подачі живлення від зовнішнього джерела в відсутності живлення 5 В від роз'єму USB, наприклад, коли необхідно запуснути плату Arduino окремо від комп'ютера.

Рівень напруги на виводах обмежений 5В, максимальний струм, який може віддавати або споживати один вивід, становить 40 мА. Всі виводи пов'язані з внутрішніми налаштовуються резисторами номіналом 20-50 кОм.

Зовнішнє живлення (не від USB) може подаватися через роз'єм Vin або роз'єм XP14 плати розширення. Платформа може працювати при зовнішній напрузі від 6 В до 20 В, але рекомендується використовувати напругу в діапазоні 7 - 12 В для запобігання перегріву або нестабільної роботи.

Деякі виводи плати Arduino UNO можуть виконувати додаткові функції:

- з використанням функцій `pinMode()`, `digitalWrite()` і `digitalRead()` кожен з 14 цифрових виходів може працювати в якості входу або виходу;
- за допомогою функції `analogWrite()` можуть виводитися 8-бітові аналогові значення в вигляді ШІМ-сигналу;
- опорна напруга 5В для аналогових входів AREF може бути задіяна функцією `analogReference()`;
- зовнішні переривання на виводах 2 і 3 можуть бути задіяні функцію `attachInterrupt()`;
- із застосуванням бібліотеки SPI ШІМ-виводи можуть здійснювати зв'язок по інтерфейсу SPI;
- із використанням бібліотеки Wire дані виводи можуть здійснювати зв'язок по інтерфейсу TWI [21].

3.1.2.1 Програмування

Arduino UNO програмується за допомогою програмного забезпечення Ардуіно. ATmega328 в Arduino UNO випускається з прошитим завантажувачем, що дозволяє завантажувати в мікроконтролер нові програми без необхідності використання зовнішнього програматора. Взаємодія з ним здійснюється за оригінальним протоколу STK500. Тим не менш, мікроконтролер можна прошити і через роз'єм для внутрішньосхемного програмування ICSP (In-Circuit Serial Programming), не застосовуючи завантажувач.

Мікроконтролер ATmega16U2 забезпечує зв'язок мікроконтролера ATmega328P з USB-портом комп'ютера. При підключенні до комп'ютера

Arduino UNO визначається як віртуальний COM-порт. Прошивка мікросхеми 16U2 використовує стандартні драйвера USB-COM, тому установка зовнішніх драйверів не потрібна.

3.1.2.2 Автоматичне (програмне) скидання

Щоб кожен раз перед завантаженням програми не було потрібно натискати кнопку скидання, Arduino UNO спроектований таким чином, що дозволяє здійснювати його скидання програмно з підключеного комп'ютера. Один з виводів ATmega8U2 / 16U2, який бере участь в управлінні потоком даних (DTR), з'єднаний з виводом RESET мікроконтролера ATmega328 через конденсатор номіналом 100 нФ. Коли на лінії DTR з'являється нуль, вивід RESET також переходить в низький рівень на час, достатній для перезавантаження мікроконтролера. Дана особливість використовується для того, щоб була можливість прошивати мікроконтролер всього одним натисненням кнопки в середовищі програмування Ардуіно. Така архітектура дозволяє зменшити тайм-аут завантажувача, оскільки процес прошивки завжди синхронізований зі спадом сигналу на лінії DTR.

На платі Uno існує доріжка (зазначена як "RESET-EN"), розімкнувши яку, можна відключити автоматичне скидання мікроконтролера. Для повторного відновлення функції автоматичного скидання необхідно спаяти між собою виводи, розташовані по краях цієї доріжки. Автоматичне скидання також можна вимкнути, підключивши резистор номіналом 110 Ом між висновком RESET і 5В.

3.1.2.3 Захист USB від перевантажень

В Arduino UNO є відновлювані запобіжники, що захищають USB-порт комп'ютера від коротких замикань і перевантажень. Незважаючи на те, що більшість комп'ютерів мають власний захист, такі запобіжники забезпечують додатковий рівень захисту. Якщо від USB-порту споживається струм більше 500 мА, запобіжник автоматично розірве з'єднання до усунення причин короткого замикання або перевантаження.

3.1.2.4 Фізичні характеристики

Максимальна довжина і ширина друкованої плати UNO становить 6.9 см і 5.4 см відповідно, з урахуванням роз'єму USB і роз'єму живлення, які виступають за межі плати. Чотири кріпильних отвори дозволяють прикріплювати плату до поверхні або корпусу. Відстань між цифровими виводами 7 і 8 не кратне традиційним 2.54 мм і становить 4 мм.

Висновок: Arduino UNO - відмінний варіант плати для створення пристрою. 14 цифрових і 6 аналогових пінів дозволяють підключати різноманітні датчики, світлодіоди, та інші зовнішні пристрої. USB-роз'єм допоможе підключитися до комп'ютера для перепрошивки скетчу без додаткових зовнішніх пристроїв, та управління реальної установки. Вбудований стабілізатор дозволяє використовувати різні елементи живлення з широким діапазоном напруги, від 6-7 до 12-14 В. У Arduino UNO досить зручно реалізована робота з популярними протоколами: UART, SPI, I2C.

Принципова схема плати Arduino UNO представлена на рис. 3.10.

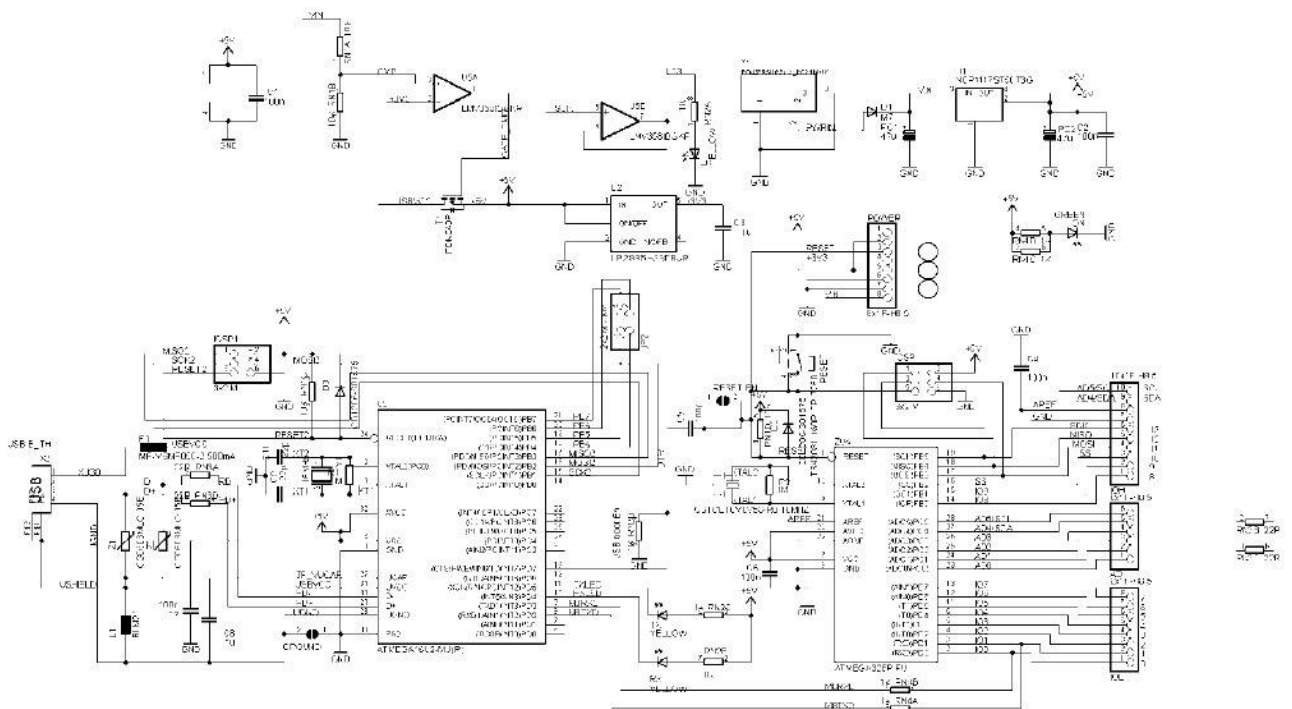


Рисунок 3.10 - Принципова схема плати Arduino UNO

Інтерфейс у всіх керуючих плат однаковий, тому може використовуватися для будь-якого пункту лабораторної роботи. Для економії простору і ресурсів до одного керуючого комп'ютера одночасно є можливість підключати декілька керуючих плат, відповідно доведеться підключити стільки ж і реальних навчальних плат. Програмний код на керуючих платах та елементи на реальній навчальній платі залишаються без змін, змінюється тільки напруга в перетворювачах напруги на інших реальних навчальних платах, що надає можливість зміни показників вимірювання та різниці варіантів завдань для студентів.

4 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ТА МЕТОЛОГІЙ ДЛЯ ПРОВЕДЕННЯ ДОСЛІДЖЕНЬ

З боку програмної реалізації для розробки віртуальної лабораторії необхідно вирішити такі завдання:

- розробка програмного забезпечення для керуючого комп'ютера;
- розробка програми для мікроконтролера;
- розробка користувальницького інтерфейсу лабораторій.

Зупинимося докладніше на кожній з них.

4.1 Програмна частина лабораторії

4.1.1 Розробка програмного забезпечення та алгоритму роботи для керуючого комп'ютера

Важливою складовою частиною автоматизованого лабораторного практикуму є поняття віртуального інструменту (VI). Віртуальним інструментом називають сукупність програмних і апаратних засобів, що реалізують функції різних вимірювальних приладів. Для реалізації VI доцільно використовувати програмний пакет LabVIEW (Laboratory Virtual Instrumentation Engineering Workbench) фірми National Instruments. Для програмування задач вводу-виводу та обробки сигналів, які не потребують високої швидкодії яка обслуговує програми, система LabVIEW є надзвичайно зручною. Переваги використання системи LabVIEW для створення віртуальних інструментів полягають в наступному:

- пакет створювався, перш за все, для інтеграції вимірювального обладнання і програмних засобів;
- різноманітні стандарти (шаблони) передніх панелей приладів - осцилографів, мультиметрів, ручок управління, перемикачів і т.п.;
- потужна математична бібліотека для обробки даних;

- провідні виробники вимірювального обладнання постачають свою продукцію драйверами під LabVIEW; таке обладнання можна використовувати в режимі нульового порогу входження, не займаючись низькорівневим програмуванням;

- фірма National Instruments підтримує ринок готових програм в середовищі LabVIEW і надає суттєві знижки освітнім установам.

Клієнтська частина являє собою ПЗ, що забезпечує користувачеві доступ до проведення експерименту. Залежно від виду лабораторного експерименту клієнтське ПЗ відправляє запит лабораторному серверу для виконання тих чи інших дій.

В даному проекті для візуалізації процесу вимірювань і управління платою Arduino UNO, було використане програмне забезпечення LabVIEW. Програмне забезпечення LabVIEW використовується на кафедрі ІТЕЗ з дисципліни «Технологія віртуальних приладів та системи аналізу даних».

LabVIEW (англ. Laboratory Virtual Instrumentation Engineering Workbench) - це середовище розробки і платформа для виконання програм, створених на графічній мові програмування «G» фірми National Instruments (США). Перша версія LabVIEW була випущена в 1986 році для Apple Macintosh, в даний час існують версії для Unix, Linux, Mac OS та ін., а найбільш розвиненими та популярними є версії для Microsoft Windows [21].

LabVIEW використовується в системах збору та обробки даних, а також для управління технічними об'єктами та технологічними процесами, та орієнтована на вирішення завдань в області автоматизованої системи наукових досліджень.

Створення програм в графічному середовищі програмування LabVIEW проводиться в двох вікнах:

- Front Panel (лицьова панель);
- Block Diagram (блок-діаграма).

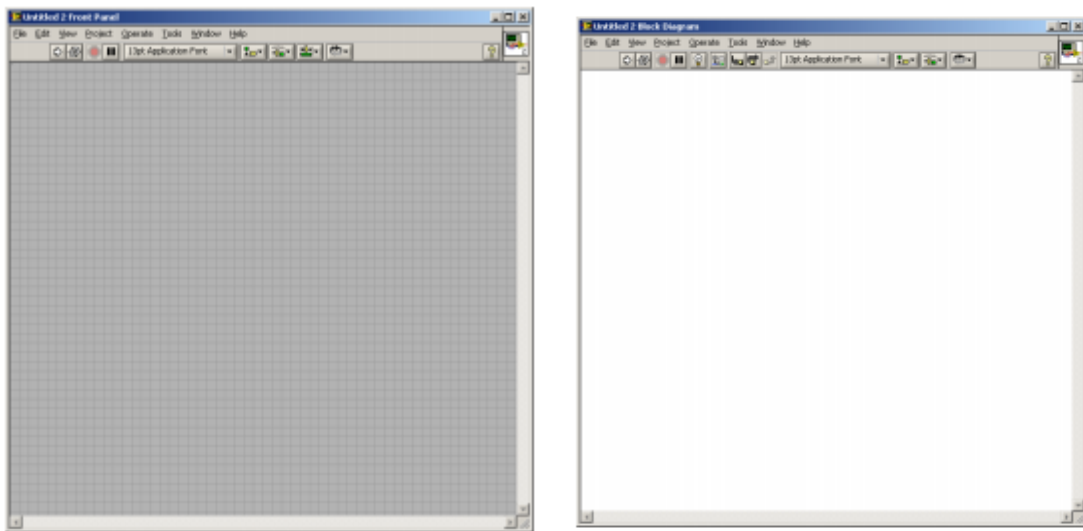
На лицьовій панелі розробляється зовнішній вигляд майбутнього віртуального приладу (всі програми розроблені на LabVIEW називаються

віртуальними приладами (Virtual Instruments) (рис. 4.1,а). На ній створюються елементи управління та відображення, які є інтерактивними засобами введення і виведення даних цього віртуального приладу.

Елементи управління (Controls) - кнопки та інші пристрої введення даних.

Елементи відображення (Indicators) - графіки, світлодіоди та інші індикатори.

Елементи управління моделюють пристрої введення даних і передають дані на блок-діаграму ВП. Елементи відображення моделюють пристрої виведення та відображення даних, які отримує або генерує блок-діаграма.



а

б

Рисунок 4.1 - Вікна:

а - Front Panel (передня панель), б - Вікно Block Diagram (блок-схема)

На блок-діаграмі розробляється вихідний код майбутнього віртуального приладу. На відміну від класичних мов програмування, таких як C++ / C#, Pascal, FORTRAN, вихідний код LabVIEW є блок-діаграмою, де всі команди, оператори циклів і порівняння зображуються графічними позначеннями (рис.4.1, б). Блок-діаграма складається з вузлів, терміналів і провідників даних.

Вузли (Node) - це об'єкти на блок-діаграмі, які мають одне або більше полів вводу-виводу даних і виконують алгоритмічні операції віртуального

приладу. Вони аналогічні операторам, функціям і підпрограмам текстових мов програмування. Вузли включають в себе функції (functions), підпрограми (SubVI) і структури (Structures). Підпрограма - віртуальний прилад, який можна використовувати на блок-діаграмі іншого ВП як підпрограми. Структури (Structures) - це елементи управління процесом, такі як структура Послідовності (Flat sequence), структура Варіанта (Case), цикл за умовою (While) і тому подібне.

Об'єкти лицьовій панелі на блок-діаграмі відображаються у вигляді терміналів даних (Data terminals). Термінали даних забезпечують обмін даними між лицьовою панеллю і блок-діаграмою. Розрізняють термінали даних наступних типів: термінали управління і відображення даних, термінали вузлів. Термінали управління і відображення відносяться до засобів управління і відображення даних на лицьовій панелі. Дані, введені в елементи управління на лицьовій панелі, надходять на блок-діаграму через термінали управління. Дані між об'єктами блок-діаграми передаються по з'єднувальним лініям – по провідникам даних (Wires). Провідник даних аналогічний змінним в текстових мовах програмування. Кожен провідник даних має єдине джерело даних, але може передавати їх до багатьох функцій. Провідники даних розрізняються кольором, стилем і товщиною лінії, в залежності від типу переданих даних.

У середовищі LabVIEW об'єкти з'єднуються провідниками даних після їх приміщення на блок-діаграму.

Перемикатися між вікнами можна декількома способами:

- клацнувши мишкою по відповідному вікну;
- комбінацією клавіш <Ctrl> <E>.

Ряд кнопок, розташований під головним меню, називається інструментальною панеллю.

Кнопка одноразового запуску Run - запускає віртуальний прилад.

Кнопка безперервного запуску Run Continuously - віртуальний прилад виконується багаторазово до моменту примусової зупинки.

Під час виконання віртуальний прилад активується кнопка Abort Execution. Ця кнопка використовується для негайної зупинки виконання віртуального приладу.

Для розміщення елементів управління і відображення даних на лицьову панель Використовується панель елементів управління і відображення (рис. 4.2). Палітра елементів управління і відображення доступна тільки з лицьової панелі. Для виведення палітри на екран слід натиснути правою кнопкою миші в робочому просторі лицьовій панелі.

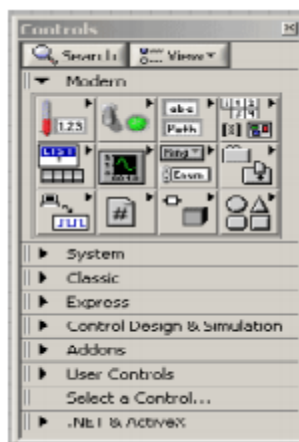


Рисунок.4.2 - Палітра елементів управління і відображення

Всі елементи управління і відображення на палітрі згруповані за розділами:

- Modern - елементи управління і відображення мають сучасний стиль;
- System - елементи управління і відображення мають стиль, що відповідає даній операційній системі (кнопки, смуги прокрутки);
- Classic - елементи управління і відображення мають класичний стиль (альтернатива стилю Modern);
- Express - поширені елементи управління і відображення.

Кожен розділ може ділитися, в свою чергу, на підрозділи (числові, строкові індикатори, кнопки).

На блоці - діаграм розробляється вихідний текст програми.

Для цього використовується палітра функцій (Functions palette) (рис.4.3). Для виведення палітри на екран слід натиснути правою кнопкою миші в робочому просторі блоку-діаграм.



Рисунок4.3 - Палітра функцій

Всі елементи на палітрі згруповані за розділами:

- Programming - тут зібрані всі основні функції, структури циклу, порівняння, оператори додавання, віднімання, необхідні для створення більшості додатків;
- Instrument I/O - тут зібрані функції для роботи з різними пристроями вводу-виводу (COM-порт і ін.);
- Mathematics - тут зібрані функції для математичних обчислень (рішення системи рівнянь, рішення диференціальних рівнянь та ін.);
- Signal Processing - тут зібрані функції пов'язані з цифровою обробкою і аналізом дискретних сигналів (цифрові фільтри, швидке перетворення Фур'є і ін.);
- Express - поширені функції, пов'язані як з обчисленнями, так і з обробкою даних;
- Select a VI ... – функції, створені користувачем.

Створювати, змінювати і налагоджувати ВП можна, використовуючи палітру інструментів (Tools Palette) (рис.4.4). Палітра інструментів доступна як на лицьовій панелі, так і на блок-діаграмі. Термін «інструмент» означає спеціальний операційний режим курсору миші. При виборі певного інструменту значок курсору змінюється на значок даного інструменту. Палітра інструментів викликається одночасним натисканням клавіші <Shift> і правої клавіші миші. Палітру Інструментів можна розміщувати в будь-якій області робочого простору блок-діаграми і лицьовій панелі. Якщо включений автоматичний вибір інструменту, то при наведенні курсору на об'єкт лицьовій панелі або блок-діаграми LabVIEW автоматично вибирає відповідний інструмент з палітри інструментів.



Рисунок 4.4 - Палітра інструментів

Автоматичний вибір інструменту включається натисканням на кнопку Automatic Tool Selection палітри інструментів (прямокутник в правому верхньому кутку палітри: зелений - включений, чорний - вимкнений) або натисканням клавіш <ShiftTab>.

– Інструмент УПРАВЛІННЯ (<палець>) використовується для зміни значення елементів управління або введення тексту. При наведенні курсору на такий елемент, як рядковий елемент управління, значок інструменту змінюється.

– Інструмент ПЕРЕМІЩЕННЯ (<стрілка>) використовується для вибору, переміщення або зміни розмірів об'єктів. При наведенні інструменту на об'єкт змінюваного розміру значок інструменту змінюється.

– Інструмент ВВЕДЕННЯ ТЕКСТУ (<A>) використовується для редагування тексту і створення вільних міток. При створенні вільних міток значок інструменту змінюється.

– Інструмент З'ЄДНАННЯ (<котушка>) створює провідники даних, поєднуючи об'єкти на блок-діаграмі.

– Інструмент виклику контекстного меню (<меню>) викликає контекстне меню відповідного об'єкта після клацання лівої кнопки миші.

– Інструмент швидкої прокрутки екрану (<рука>) використовується для перегляду вікна без використання смуги прокрутки.

– Інструмент ВВЕДЕННЯ КОНТРОЛЬНОЇ ТОЧКИ (<контрольна точка>) дозволяє розставляти контрольні точки на ВП, функціях, вузлах, провідниках даних, структурах і призупиняти в них виконання програми.

– Інструмент ВСТАНОВЛЕННЯ ВИПРАВНИХ ІНДИКАТОРІВ (<пробник>) дає можливість досліджувати потік даних в провідниках блок-діаграми. Використовується для перегляду проміжних значень при наявності сумнівних або несподіваних результатів роботи ВП.

– Інструмент КОПІЮВАННЯ КОЛЬОРУ (<піпетка>) призначений для копіювання кольору з подальшої вставкою за допомогою інструменту розфарбовування (<кисть>).

– Інструменту розфарбовування (<кисть>) дозволяє змінити колір об'єкта. він також відображає поточний передній план і параметри настройки кольору фону.

Якщо автоматичний вибір інструменту вимкнений, можна змінювати інструменти палітри за допомогою клавіші <Tab>.

Мікропроцесорна система на основі плати Arduino UNO підтримує зв'язок мікроконтролера ATmega328P через USB-порт комп'ютера з інтерфейсом у LabVIEW. Мета інтерфейсу в LabVIEW полягає в підтримці послідовного терміналу зв'язку, що дозволяє користувачеві відправити команду читання чи запису в контролер Arduino для обміну даними .

Створення каналу зв'язку в LabVIEW для контролера Arduino UNO

починається з використання візуального компоненту VI «VISA Configure Serial Port», що призначений для налаштування послідовного порту (рис. 4.5).

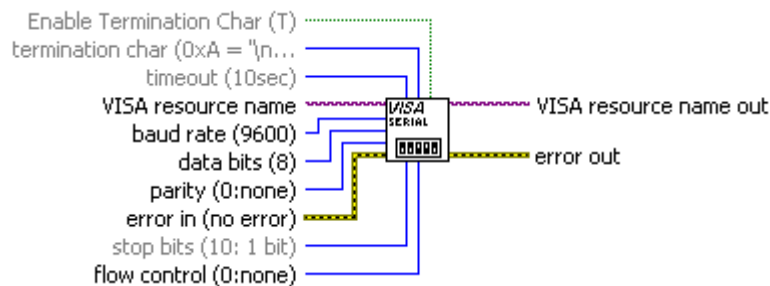
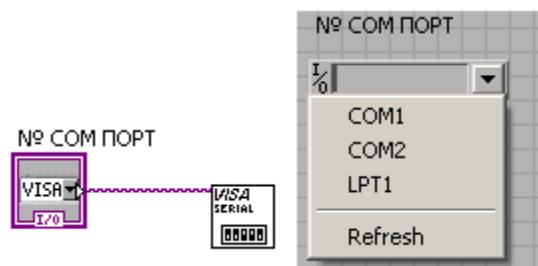


Рисунок 4.5 - Візуальний компонент VI «VISA Configure Serial Port»

Для вибору того який COM-порт буде займати плата Arduino UNO створюється візуальний компонент (БК) «VISA resource name». Блок вибору має вигляд (рис. 4.6 а,б).



а

б

Рисунок 4.6 - Блок вибору має вигляд:

а – на блок-діаграмі; б – на фронтальній панелі

Оператори потоку управління, які використовуються для виконання блоків коду, це структури «Event Structure» та «While Loop».

Структура подія «Event Structure» має одну або більше піддіаграм або варіантів подій, з яких тільки один виконується при зверненні до структури.

Структура цикл за умовою «While Loop» де код програми, розміщений в структурі, виконується до подачі на термінал умови логічної змінної.

Візуальний компонент «VISA Close» VI (рис. 4.7), який закриває послідовний комунікаційний порт, коли буде натискатися кнопка зупинки «Stop»



Рисунок 4.7 – Візуальний компонент VI «VISA Close»

Кнопки і перемикачі в LabVIEW мають логічний тип даних значення (TRUE / FALSE), для передачі потрібного значення використовується «Select» (рис .4.8).



Рисунок 4.8 – Компонент «Select»

«Select» повертає значення, підключений до входу t або f, в залежності від значення. Якщо s TRUE, ця функція повертає значення, підключений до t. Якщо s FALSE, ця функція повертає значення, пов'язане з f. Ці значення потім необхідно форматувати в строку «Format Into String» (рис. 4.9).

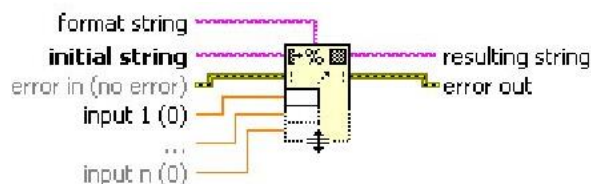


Рисунок 4.9 - Компонент «Format Into String»

Отриману строку необхідно далі відправити на мікроконтролер через компонент VI «VISA Write» (рис. 4.10).



Рисунок 4.10 – Візуальний компонент VI «VISA Write»

Зчитування даних з серійного порту підключеного пристрою, здійснюються за допомогою компоненту «VISA Read» (рис.4.11).

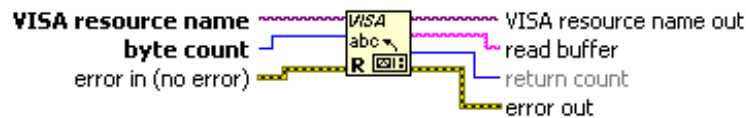


Рисунок 4.11 - Візуальний компонент VI «VISA Read»

Сканування вхідного рядка і перетворення рядка у відповідності зі рядком формату здійснюються компонентом «Scan From String» (рис .4.12).

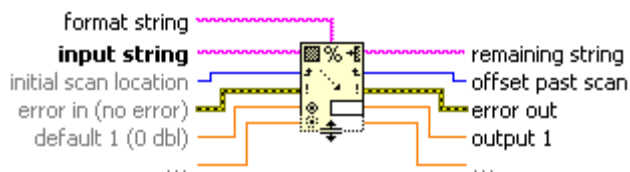


Рисунок 4.12 - Компонент «Scan From String»

Перевірка того, чи доступні якісь дані на серійному порту, та порівнюється з 0,здійснюються компонентом «Bytes at Serial Port» (рис.4.13).

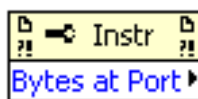


Рисунок 4.13 - Компонент «Bytes at Serial Port».

У проекті були використані вище перераховані компоненти, в якості керуючих елементів на фронтальній панелі встановлені кнопки та тумблери. Блок-схема та фронтальна панель віртуальної лабораторії знаходиться у Додатку А та Додатку Б.

Алгоритм роботи полягає у наступному. Клієнтська програма посилає послідовність вхідних значень на керуючий комп'ютер. Далі ця інформація надсилається через USB на керуючу плату з мікроконтролером. Мікроконтролер встановлює на виходах передані значення, і результат з'являється на екрані користувача.

Керуюча плата підключена до певного СОМ порту, коли користувач обирає цей порт, з ПК в свою чергу надсилається запит на мікроконтролер, який готовий до використання і чекає команд від користувача.

Алгоритм програми керуючого ПК представлений на рис. 4.14.

- 1) Запуск програми.
- 2) Подача живлення на керуючу плату з мікроконтролером.
- 3) Користувач обирає СОМ- порт, з яким хоче працювати. До кожного СОМ-порту підключена керуюча плата та реальна плата . Програма очікує інформацію про вибір порту .
- 4) Якщо користувач обрав, то відправляється запит на керуючу плат . Інакше до п.3.
- 5) Якщо підключена обрана плата, то переходимо до п.6, інакше відправляється запит на іншу підключену керуючу плату (перехід до п.4).
- 6) Користувач починає роботу з платою. Якщо стану перемикачів були змінені користувачем, то переходимо до п.7, інакше очікуємо змін далі.
- 7) Формування послідовності вхідних значень для передачі на МК.
- 8) Передача отриманої послідовності на МК.
- 9) Зробити запит на МК про зміну стану виходів реальної плати.
- 10) Передача послідовності вихідних значень на сервер для відображення відповідної реакції на реальній платі.

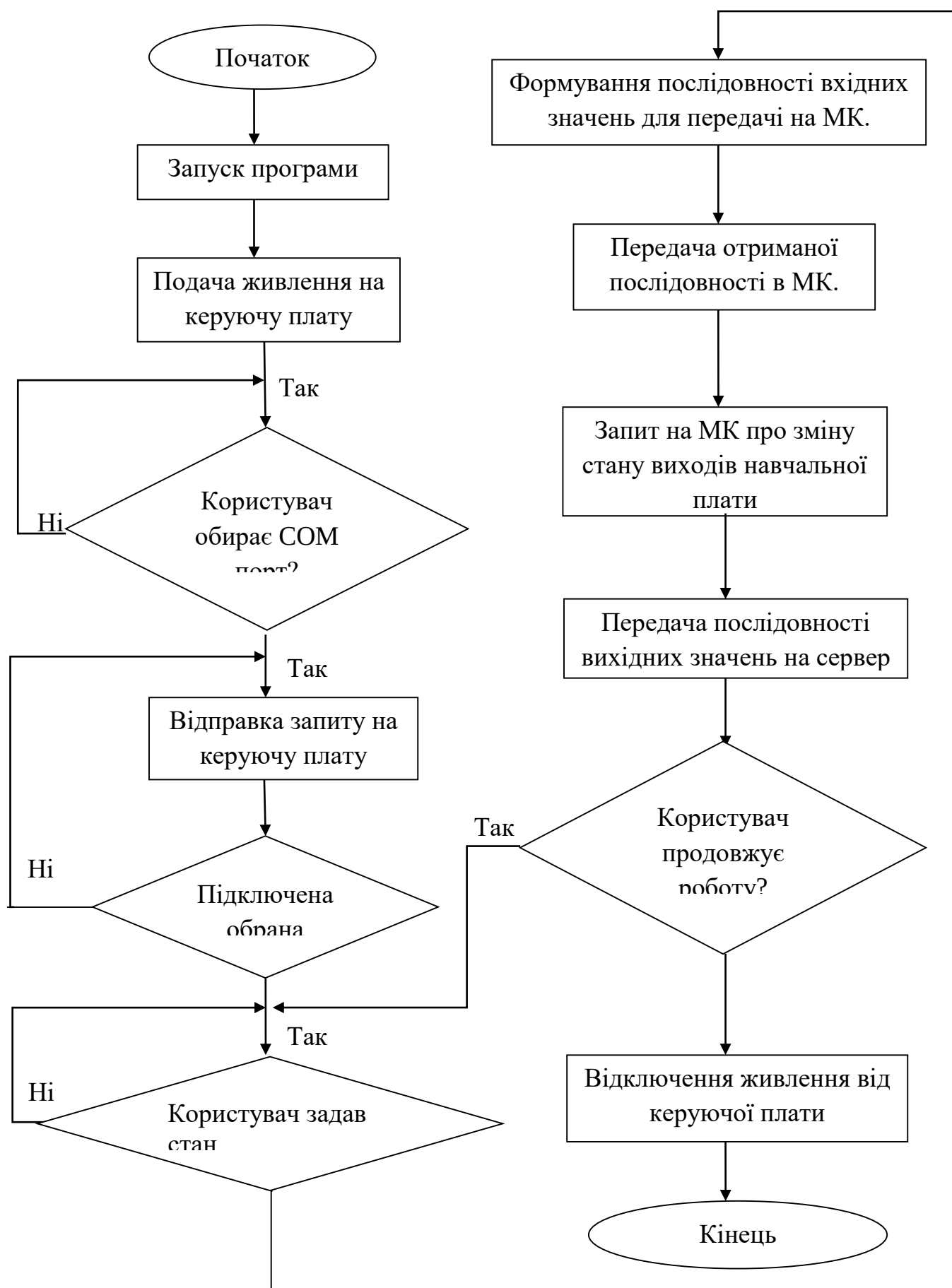


Рисунок 4.14 - Алгоритм програми керуючого ПК

11) Якщо користувач продовжує роботу з даною платою, то перехід до п.6, інакше переходимо до п.12.

12) Відключення живлення від керуючої плати.

13) Кінець програми.

4.1.2 Розробка та алгоритм роботи програми для мікроконтролера

Всі функції середовища розробки програм Arduino IDE (Integrated Development Environment – інтегроване середовище розробки) [22] доступні для використання, якщо приєднана плата Arduino (рис.4.15).

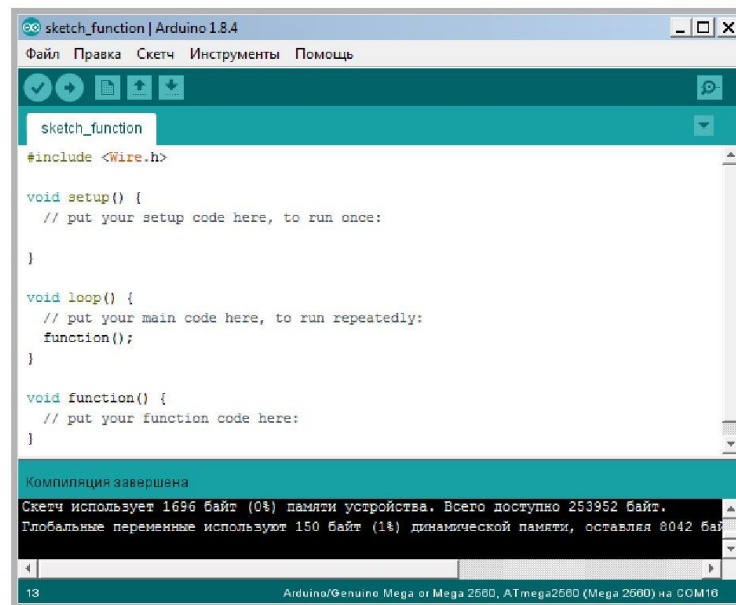


Рисунок 4.15 – Области вікна середовища розробки Arduino

Пункти меню редактора програм Arduino IDE (рис.4.15) включає в себе наступні основні елементи: файл, правка, скетч, сервіс і довідка.

Кожне з меню складається з пунктів:

1) Меню «Файл» містить команди, що відповідають за створення нової програми, завантаження з диска вже існуючої програми, збереження змін, а також команди для завантаження програми на мікроконтролер:

- «Створити» – створити нову програму (в даному середовищі програми називаються скетчами);

- «Відкрити» – відкрити існуючу програму;

- «Папка зі скетчами» – відкрити програму із заданої папки;

- «Приклади» – відкрити приклад програми;

- «Закрити» – закрити поточне вікно;

- «Зберегти» – зберегти зміни;

- «Зберегти як» – зберегти програму в новому файлі;

- «Завантажити» - завантажити програму в Arduino;

- «Завантажити за допомогою програматора» – завантажити програму за допомогою програматора;

- «Налаштування друку» – настройка принтера;

- «Друк» – висновок на друк коду програми;

- «Налаштування» – настройки редактора;

- «Вихід» – вихід з Arduino IDE.

2) Меню «Правка» містить команди, пов'язані з редагування тексту програми, що включають в себе: копіювання, вставку, настройку відступів і пошук.

3) Меню «Скетч» містить команди для контролю за процесом компіляції програми:

- «Перевірити / Компілювати» – компілювати програму;

- «Показати папку скетчей » – відкриває системну папку з програмами;

- «Додати файл» – додати до проекту файл з даними або програмою;

- «Імпортувати бібліотеку» – підключити до програми бібліотеку зі списку встановлених.

4) Меню «Інструменти» включає в себе допоміжні функції для роботи з самим мікроконтролером:

- «Автоформатування» – автоматична розстановка відступів, переносів рядків і т.п.;

- «Архівувати скетч» – архівація папки з програмою та збереження архіву у вказане місце;
- «Монітор порту» – відкрити вікно для обміну даними з мікроконтролером;
- «Плата» – вибір поточної плати (в нашому випадку Arduino UNO);
- «Послідовний порт» – вибір порту, до якого підключений;
- «Програматор» – вибір програм (не використовується);
- «Записати завантажувач» – запис програми завантажувача в мікроконтролер (також нами не використовується).

5) Меню «Довідка» містить докладний опис всіх функцій самого редактора Arduino IDE, а також різні команди і прийоми роботи з платформою.

На панель інструментів винесені у вигляді кнопок найбільш часто використовувані функції редактора програм Arduino IDE. Призначення кнопок на панелі інструментів описано в табл. 4.1.

Таблиця 4.1 - Кнопки на панелі інструментів

Кнопки	Призначення
	Перевірити / Компілювати програму
	Завантажити програму в Arduino
	Створити нову програму
	Відкрити існуючу програму
	Зберегти програму
	Монітор послідовного порту

Безпосередньо текст програми створюється і редагується у вікні редактора коду. Це вікно є типовим текстовим редактором з підсвічуванням синтаксису програми. У нижній частині вікна Arduino IDE є область, що служить для виведення повідомлень про помилки, що виникають у процесі компіляції програми або під час завантаження програми в мікроконтролер.

Програмування в середовищі Arduino здійснюється на мові Сі з мінімальним набором елементів С++. Ця мова дуже схожа на мову С++, але має деякі відмінності. Програма для Arduino повинна обов'язково бути написана в одному файлі, який називають скетч або програма «скетч для Arduino». В структурі скетчу є відмінності від структури звичайної програми на С++, що пов'язано з відмінністю між роботою програми на звичайному комп'ютері та в мікропроцесорних системах.

При виконанні програми на звичайному комп'ютері, яка написана на мові Сі, за її виконання відповідає функція `main()`. При запуску програми запускається ця функція, з якої можна викликати інші функції програми. Коли функція `main()` повертає управління за допомогою `return`, програма завершує роботу.

Для програми, що управляє мікроконтролером, така поведінка не має сенсу. Поки на МПС подано живлення, мікроконтролер повинен управляти пристроєм, а значить, повинен постійно виконувати свою програму. Ця програма ніколи не завершується сама, вона перестає працювати, тільки якщо вимкнути мікроконтролер. Тому структура програми для Arduino відрізняється від структури звичайної програми на мові Сі. У програмі для Arduino немає функції `main()`, Замість неї в мікроконтролері є дві функції: `setup()` і `loop()`.

Функція `setup()` викликається один раз у самому початку виконання програми відразу після того, як мікроконтролер Arduino включається. Ця функція налаштовує мікроконтролер, зокрема задає, які його виводи будуть використовуватися як входи, а які – як виходи і так далі. У цій функції можна якимось іншим чином приготувати пристрій до роботи, наприклад, включити додаткові можливості, або навпаки, переконатися, що нічого зайвого поки не включено.

Функція `loop()` виконується весь час, поки працює мікроконтролер Arduino. Якщо вона завершається, то середовище Arduino запускає її заново. Саме у цій функції потрібно реалізувати алгоритм роботи пристрою. Зазвичай у функції `loop()` опитують підключені до мікроконтролеру сенсори, або самі

входи мікроконтролера, аналізують отриману інформацію і видають команди виконавчим пристроям.

В бібліотеці Arduino вже є функція `main()`, яка може бути влаштована наступним чином:

```
void main () {  
    setup ();  
    while (true)  
        loop ();  
}
```

Така функція `main()` дійсно є в бібліотеці Arduino, але вона влаштована складніше. У середовищі Arduino як і в звичайних програмах, можна створювати будь-які інші функції користувача і викликати їх з функцій `setup()` і `loop()`. Якщо розробляти просту програму, то можна написати весь код просто всередині функцій `setup()` і `loop()`. Проте, якщо робити щось складне, то варто розбити код програми на окремі функції.

В програмі Arduino можна викликати стандартні функції, які є в бібліотеці. Для цього не потрібно додавати директив `#include`. У звичайній мові Сі це потрібно робити навіть для стандартних функцій. Одна з функцій, яка не потребує додавання директив `#include` є функція `delay(time)`, яка виконує затримку на зазначений час.

Приклад функції `delay(time)`, що виконує затримку на зазначений час, що задається в мілісекундах, має вигляд:

```
delay (1000);
```

В програмі Arduino є ще кілька стандартних функцій, які можна відразу писати в програмі. Ці функції керують базовими можливостями мікроконтролера.

Великою перевагою середовища Arduino є велика кількість плат розширення різних виробників, які можна підключати до Arduino і використовувати разом з МПС. Для того, щоб такі плати було зручно використовувати, для них написані бібліотеки, що містять зручні функції для

роботи з цими пристроями. Такі бібліотеки представляють окремі компоненти, які можна підключити до плати Arduino.

Щоб використовувати бібліотеку, потрібно на початку програми помістити директиву `#include`:

```
#include<ім'я_бібліотеки.h>
```

Наприклад, бібліотека для роботи з дисплеєм називається `LiquidCrystall`, і щоб її використовувати, треба спочатку написати:

```
#include <ім'я_бібліотеки.h>
```

У мові Cі в середовищі Arduino для роботи з бібліотеками використовують елементи мови C++. Фактично, самі бібліотеки пишуть на C++, але в скетчі можна використовувати тільки дуже прості можливості цієї мови. Мова C ++ дозволяє створювати класи. Класи містять призначені для користувача типи даних, що описують набір полів і операції, які можна з ними робити. З допомогою цього механізму в C++ можна зробити, наприклад, клас «`NewLiquidCrystall`», що наслідує властивості класу «`LiquidCrystall`» і об'єкт, який належить цьому класу. Щоб зробити такий об'єкт, треба його оголосити, як оголошують звичайні змінні. Наприклад, для роботи з дисплеєм треба зробити змінну типу `LiquidCrystall`:

```
LiquidCrystal lcd (4, 5, 6, 7, 8, 9);
```

У цьому прикладі `LiquidCrystall` – це назва типу, яка описана в документації у бібліотеці «`LiquidCrystall`». Змінна `lcd` – це назва об'єкту, який можна використовувати як будь-яку змінну. Цифри – це параметри, які передаються об'єкту при створенні. Про те, які параметри треба вказати, також описано в документації. Для дисплея це, номери виводів мікроконтролера, до яких він підключений.

Для роботи з об'єктами використовують спеціальні функції, які доступні тільки для цієї змінної та представляють методи класу. Викликають методи класу наступним способом:

```
ім'я_змінної . ім'я_функції (параметри);
```

Наприклад, текст на дисплей виводить функція `print()`, яку можна викликати наступним чином:

```
lcd.print ( "Hello!");
```

У прикладі спочатку вказується ім'я змінної, тому функція `print()` викликається саме для цієї змінної. Таким чином, можна підключити кілька дисплеїв і зробити для кожного з них свою змінну.

Інші особливості синтаксису мови Cі для Arduino UNO, знання яких необхідно для написання програм.

Кожен вираз на мові Cі закінчується символом крапки з комою. Наприклад вираз:

```
a = b + c;
```

Тіло функцій і складових операторів (`if`, `else`, `for`, `while`) виділяються фігурними дужками, наприклад:

```
if (a > 0) {  
    b = a - c; }
```

Строкові змінні виділяються подвійними лапками:

```
lcd.print ( "some text")
```

Символи виділяються одинарними лапками:

```
symbol = 'a';
```

Підключення бібліотек здійснюється за допомогою конструкції:

```
#include<math.h>
```

Коментарі в програмі починаються двома прямими виду слеш:

```
// це коментар
```

Оголошення змінної в мові Cі здійснюється за допомогою конструкції виду: тип_змінної ім'я_змінної;

Наприклад, оголошення змінної має вигляд:

```
int x, y; // оголошені змінні x і y, мають цілий тип
```

У програмах для Arduino можна використовувати такі типи для числових даних: `byte` – ціле число від 0 до 255;

`int` – ціле число від -32768 до +32767;

word – ціле число від 0 до 65535;

long – ціле число від -2147483648 до 2147483647;

float і double – числа з плаваючою точкою, відповідають типу float для звичайних комп'ютерів, тип double еквівалентний float.

Масиви в мові Сі оголошуються конструкцією виду:

тип_елемента ім'я_масиву [розмір];

Наприклад, оголошення масиву має вигляд:

int values [10]; // масив з десяти цілих чисел

У мові Сі передбачений тип для зберігання символів, це тип – char. У векторі та масиві, що мають тип char останній символ має код 0, який позначає кінець масиву.

Наприклад, оголошення змінної типу char має вигляд:

char data_str; // вектор довжиною не більше дев'яти символів

Тип boolean використовується для представлення логічних значень true() і false(). Крім того, будь-яке ціле значення може використовуватися як логічне значення, при цьому 0 означає – false(), а будь-яке не нульове значення – true().

Тип const перед оголошенням змінної свідчить про те, що її значення не може бути змінено у процесі виконання програми.

Останній тип даних – це тип void, який використовується при оголошенні функцій, що не мають аргументів або які не повертають результат.

У мові Сі використовується стандартний набір математичних операторів, що включає +, -, * і /, для обчислення залишку від ділення використовується %. Крім того, є їх спеціальні комбінації з оператором присвоювання, що дозволяють скоротити запис:

x += 4; // означає x = x + 4

Такі форми є для всіх операторів. Щоб збільшити або зменшити задане число на одиницю, є інкремент і декремент:

x ++; // означає x = x + 1

y--; // означає y = y – 1

Набір операторів порівняння є стандартним і включає оператори `, <=, > =`. Порівняння записується за допомогою двох знаків рівності (`==`), а нерівність записується `!=`. Логічне заперечення записується за допомогою знаку оклику `!`. Логічні вирази можна записувати за допомогою операторів, що представлені в таб.4.2.

Умовний оператор має наступний синтаксис:

`if (умова)`

код, що виконується, якщо умова істинна.

Таблиця 4.2 - Логічні операції на мові C++

Операція	Визначення	Приклад	Опис складної умови
і	<code>&&</code>	<code>a == 3 && b > 4</code>	істинно, якщо істинні обидві прості умови
або	<code> </code>	<code>a == 3 b > 4</code>	істинно, якщо істинна, хоча б одне з простих умов
ні	<code>!!</code>	<code>(a == 3)</code>	істинно, якщо а не дорівнює 3

Якщо в разі істинності умови необхідно виконати декілька операторів, то вони об'єднуються у один блок за допомогою фігурних дужок:

`if (умова) {`

код, що виконується, якщо умова істинна;

`}`

Повна форма умовного оператора включає блок `else`, Код в якому виконується інша умова, якщо істина умова помилкова має вигляд:

`if (умова) {`

код, що виконується, якщо умова істинна;

`}`

`else {`

код, що виконується, якщо умова помилкова;

```
}
```

Оператор switch дозволяє здійснити вибір однієї з гілок коду в залежності від значення числової змінної, наприклад:

```
switch (mode) {
  default:
  case 0:
    код для режиму 0;
    break;
  case 1:
    код для режиму 1;
    break;
  case 2:
    код для режиму 2;
    break;
}
```

При цьому значення змінної mode визначає, в яку точку коду перейде виконання програми після виконання оператора switch. Далі будуть виконуватися всі дії, поки не зустрінеться break, який перерве подальше виконання коду всередині switch. Якщо значення виразу не відповідає жодній з міток, то виконується код, починаючи з мітки default. В даному прикладі через відсутність break після default управління потрапить в «код для режиму 0».

Цикл while дозволяє виконувати програмний код до тих пір, поки умова вірна. Він має синтаксис:

```
while (умова)
  код, що виконується в циклі.
```

Більш складним є цикл for:

```
for (ініціалізація; умова; крок)
  код, що виконується в циклі;
```

У цьому циклі вираз, що вказаний в блоці «ініціалізація», виконується один раз перед початком циклу. Зазвичай в ньому задають початкові значення для змінних циклу. Умова перевіряється кожен раз на самому початку чергової ітерації циклу і якщо вона стане помилково, то цикл зупиняється.

Вираз «крок» виконується після виконання тіла циклу і зазвичай збільшує (або зменшує) змінні циклу. У блоках «ініціалізація» і «крок» можна задати кілька змінних, якщо написати відповідні вирази через кому.

Приклад циклу, що обчислює суму арифметичної прогресії:

```
int i, sum;
for (i = 0, sum = 0; i <= 10; i++)
    sum += i;
```

Код програми на мові Сі повинен бути обов'язково розбитий на функції. Для оголошення функції використовується наступний синтаксис:

```
тип_функції ім'я_функції (параметри)
{
    код, що виконується в рамках функції;
    return результат_функції;
}
```

Тип функції описує тип значення, яке буде повертати функція. Наприклад, стандартна функція `cos`, що обчислює косинус, повертає значення типу `float`.

Ім'я функції може мати будь-яку форму, що починається з букви, містить тільки літери, цифри і символи підкреслення. Параметри – це перелік змінних, які приймають значення, передані в функцію. Змінні вказуються через кому, спочатку пишуть тип параметра, а потім – його назву.

Щоб повернути значення з функції використовується оператор `return`. Він негайно перериває виконання функції, а результатом виконання функції стає значення зазначеного після `return` вираження.

Якщо функція нічого не повертає, то тип_функції вказується як void. Якщо функція не передається ніяких аргументів, то можна нічого не писати всередині круглих дужок або можна написати там слово void.

Наприклад функція, що підсумовує два числа типу int має вигляд:

```
int sum (int a, int b) {
    int result;
    result = a + b;
    return result;
}
```

Для виклику функції необхідно написати її ім'я і в круглих дужках вказати через кому значення аргументів, наприклад:

```
r = sum (3, 10);
```

У програмах на мові Cі можна оголосити змінні за межами будь-яких функцій. Такі змінні є глобальними і їх можна використовувати в будь-яких функціях, що розташовані після оголошення змінної. Наприклад, якщо змінна led використовується у функціях setup() і loop() вона описується як глобальна:

```
const int led = 2;
void setup () {
    pinMode (led, OUTPUT);
}
void loop () {
    digitalWrite (led, HIGH);
    delay (1000);
    digitalWrite (led, LOW);
    delay (1000);
}
```

Одна з найпростіших дій, яку може виконати мікроконтролер, це перемикання стану контактів цифрових входів-виходів на платі Arduino UNO. Будь-який з цифрових виходів мікроконтролера можна програмним способом переключити між високим рівнем (+5 В) і низьким рівнем (0 В).

За допомогою такого перемикання відбувається управління простими пристроями:

1) Перемикання стану контактів цифрових входів-виходів на платі Arduino UNO здійснюється функцією `digitalWrite()`, виклик якої має вигляд: `digitalWrite (номер_контакта, рівень_сигналу)`.

2) Перемикання стану контактів аналогових входів-виходів на платі Arduino UNO здійснюється функцією `analogWrite()`, виклик якої має вигляд: `analogWrite (номер_контакта, рівень_сигналу)`.

3) Читання стану контактів цифрових входів-виходів на платі Arduino UNO здійснюється функцією `digitalRead()`, виклик якої має вигляд: `ім'я_змінної = digitalRead (номер_контакта)`.

4) Читання стану контактів аналогових входів-виходів на платі Arduino UNO здійснюється функцією `analogRead()`, виклик якої має вигляд:

`ім'я_змінної = analogRead (номер_контакта)`.

Параметр рівня сигналу може приймати два значення: HIGH (високий, +5 В) або LOW (низький, 0 В). Параметр номер_контакта – це число, що відповідає номеру контакту на платі Arduino. Параметр ім'я_змінної – це змінна, до якої записується числовий еквівалент сигналу, що встановлений для заданого номеру контакту на платі Arduino UNO.

Одні й ті ж виводи мікроконтролера можуть виступати як в ролі цифрових виходів, так і в ролі цифрових входів. Тому перед їх використанням необхідно вказати, в якій ролі буде використовуватися контакт: вхід або вихід. Це робиться за допомогою функції `pinMode()`:

`pinMode (номер_контакта, режим_контакту);`

Аргумент цієї функції режим_контакту може приймати значення: OUTPUT (вихід) і INPUT (вхід). Таким чином, щоб на цифровому виході (2) встановити спочатку високий рівень сигналу (HIGH), а потім з затримкою (Delay) 1000 мс встановити низький рівень сигналу (LOW), достатньо виконати наступну програму:

`const int pin = 2;`

```

void setup () {
  pinMode (pin, OUTPUT);
  digitalWrite (pin, HIGH);
}

void loop () {
  digitalWrite (led, HIGH);
  delay (1000);
  digitalWrite (led, LOW);
  delay (1000);
}

```

Контролер Arduino працює на частоті 16 МГц, тому він може включати і вимикати світлодіод, що підключений до виводу (2) тисячі разів у секунду. У сучасній мікроелектроніці застосовуються мініатюрні світлодіоди для поверхневого монтажу. Такі індикатори, наприклад, є на Arduino UNO для інформування користувача про стан системи.

Робоча напруга живлення світлодіода варіюється від 1,85 В до 2,5 В при рекомендованій силі струму не більше 20мА. Щоб сила струму не перевищила це значення, при підключенні світлодіода до виходу мікроконтролера, який видає напругу 5 В, у ланцюг слід додати послідовний обмежуючий резистор з опором від 200 до 500 Ом.

Мікропроцесорна система на основі плати Arduino UNO дозволяє реалізувати вимірювальні прилади з мікропроцесорним управлінням [21].

Скетч з коментарями до керуючої плати Arduino UNO знаходиться у Додатку В.

На рисунку 4.16 зображений алгоритм роботи керуючої плати з мікроконтролером.

Керуюча плата з мікроконтролером (МК) забезпечує зв'язок між керуючим ПК і навчальної платою. Мікроконтролер відповідає за управління сигналами, що подаються на навчальну плату, і отримання відповідної реакції.

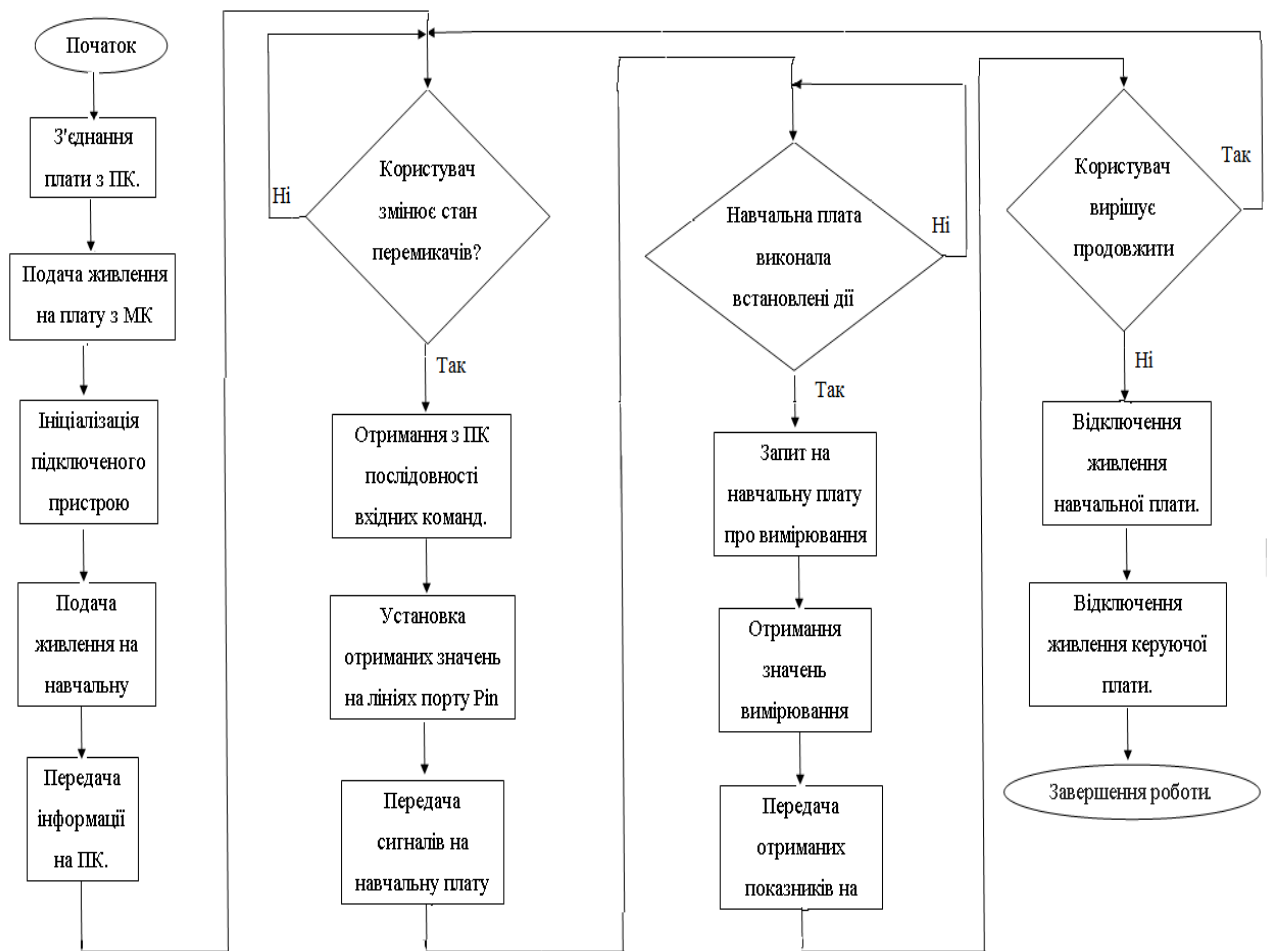


Рисунок . 4.16 - Алгоритм роботи керуючої плати.

- 1) Початок.
- 2) З'єднання плати з ПК.
- 3) подача живлення на плату через роз'єм USB та на мікроконтролер ATmega328.
- 4) Ініціалізація підключеного пристрою, ПК запитує інформацію з МК про тип пристрою.
- 5) подача живлення на навчальну плату, шляхом відкриття ключа подачі напруги живлення.
- 6) Передача інформації на ПК.
- 7) Якщо користувач змінює стан перемикачів, то переходимо до п.8, інакше знову очікуємо дій користувача.
- 8) Отримання з ПК послідовності вхідних команд .

- 9) Установка отриманих значень на лініях порту Pin-x (вхід) МК.
- 10) Передача сигналів на навчальну плату .
- 11) Якщо навчальна плата виконала встановлені дії, то переходимо до п.12, інакше очікуємо виконання.
- 12) Запит на навчальну плату про вимірювання показників при зміні стану світло діодів.
- 13) Отримання значень вимірювання показників на лінії порту A0 та A1 на МК.
- 14) Передача отриманих показників на ПК для відображення зворотної реакції з навчальної плати на екрані користувача.
- 15) Якщо користувач вирішує продовжити роботу, то переходимо до п. 7, інакше - до п.16.
- 16) Відключення живлення навчальної плати.
- 17) Відключення живлення керуючої плати.
- 18) Завершення роботи.

4.1.3 Розробка користувацького інтерфейсу лабораторій

Інтерфейс користувача відповідає за зовнішній вигляд віртуальної лабораторії та розрахований на студентів і викладачів кафедри ІТЕЗ. Віртуальна лабораторія матиме обмежений доступ, у кожного користувача є свій пароль. Для зручності а також з майбутнім розширенням практики дистанційного навчання на кафедрі ІТЕЗ з різних дисциплін , потрібно використовувати систему віддаленого доступу ,яка дозволяє швидко і безпечно отримати віддалений доступ через Інтернет до комп'ютера та керувати ними в режимі реального часу.

Оптимальним рішенням для віддаленого доступу до комп'ютера через Інтернет та дистанційного навчання є програмне забезпечення Ammuu Admin.

Віддалене управління придбало значну популярність за останні кілька років. Завдяки різним додатковим можливостям, віддалений доступ

використовується в багатьох сферах надаючи комфортну роботу при управлінні серверів, допомоги клієнтам в техобслуговуванні, дистанційному навчання, а також в рядових ситуаціях звичайних користувачам.

Додаток Ammyu Admin є одним з найбільш популярних додатків для встановлення віддаленого доступу між комп'ютерами. Завдяки стабільній роботі, а також низьким вимогам до обладнання комп'ютера додаток широко використовується як в корпоративному сегменті, так і звичайними користувачами. У додатку реалізований повний набір функцій, які найбільш часто використовуються при роботі з віддаленим комп'ютером, включаючи файловий менеджер, і голосовий зв'язок.

Ammyu Admin відрізняється простим і зрозумілим інтерфейсом, працювати з яким зручно не тільки професіоналам, але також і початківцям користувачам. Програма не вимагає інсталяції, користувачеві досить завантажити Ammyu Admin запустити ярлик програми. Вікно середовища системи віддаленого доступу Ammyu Admin при запуску на рис.4.17.

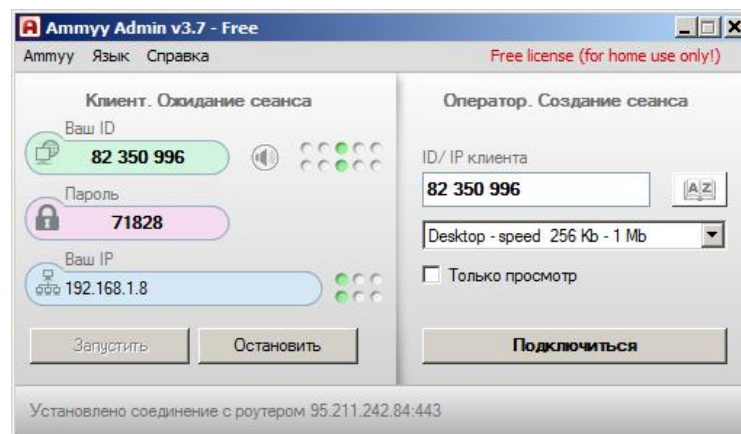


Рисунок4.17 - Вікно середовища при запуску Ammyu Admin

Для вставлення з'єднання між комп'ютерами користувачеві необхідно виконати елементарні дії: достатньо лише надати свій ID або IP іншому користувачеві. Користувач до якого підключаються може обмежити доступ до тих або інших функцій. Вікно для налаштування обмежень на рис. 4.18.

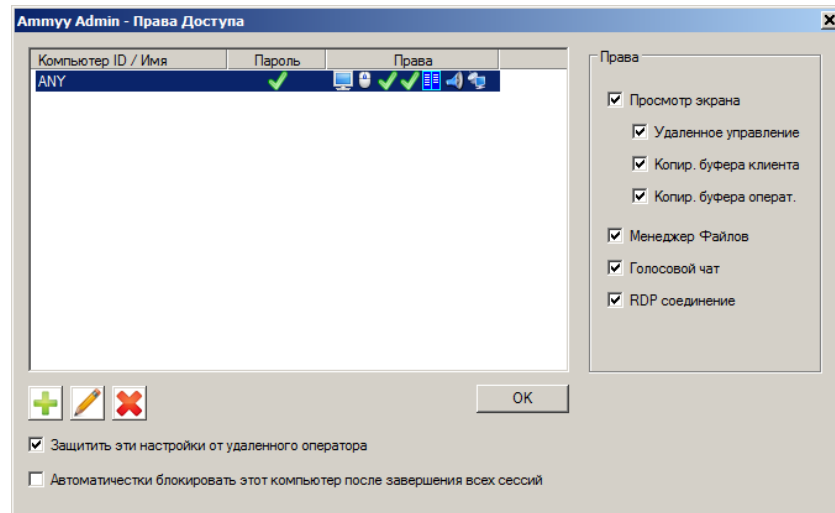


Рисунок 4.18 - Вікно для налаштування обмежень користувача

Особливостями програми Ammyu Admin велика кількість додаткових функцій і особливостей.

– Сумісність. Ammyu Admin має підтримку різних операційних систем лінійки Windows, включаючи як досить застарілі 2000 і XP, так і самі останні версії 8 і 8.1. Додаток також повністю підтримує роботу з серверами в операційних системах Windows Server 2000, 2003 і 2008.

– Висока швидкість. Реалізована в програмі система оптимізації трафіку дозволяє отримати високу швидкість роботи з віддаленим комп'ютером. Завдяки оптимізації програми користувач також отримує високий рівень стабільності роботи при різних швидкостях підключення до Інтернету.

– Голосовий чат. Для більш зручної роботи з віддаленим комп'ютером програма Ammyu Admin забезпечує можливість голосового чату, що значно спрощує роботу при навчанні, а також в технічній допомозі.

– Робота з Firewall. Ammyu Admin відмінно працює з найбільш відомими програмами для мережевого контролю. Користувачеві не потрібно налаштовувати порти, створювати виключення, які створюють вразливість вашої системи, програма підтримує роботу з включеним мережевим екраном.

– Висока функціональність. У додатку Ammyu Admin реалізовано безліч функцій, завдяки яким робота з віддаленими пристроями стає набагато простіше. Одним з нововведень є можливість перезавантаження і зміни користувача за допомогою програми. Ammyu Admin також надає можливість проводити он-лайн презентації, що забезпечує додаткові зручності в роботі.

– Безоплатність. Розробник надає безкоштовну версію для приватного використання. Для комерційного використання, а також для отримання додаткових функцій і можливостей необхідно придбати повноцінну версію програми.

Для більш зручної роботи з великою кількістю віддалених комп'ютерів користувач може створювати книгу контактів для найбільш швидкого підключення до певного комп'ютера. У свою чергу власник комп'ютера, до якого ведеться підключення, для кожного оператора може призначити відповідний рівень доступу, включаючи необхідність введення пароля.

У додатку розробнику вдалося реалізувати нову систему криптооперації. Передача інформації, включаючи переміщення мишки, натискання клавіш, а також інші дії, здійснюється програмою Ammyu Admin виключно в зашифрованому вигляді. Крім того, дана система не може бути відключена користувачем. Шифрування даних відповідає світовим стандартам RSA і AES, що підтверджує високу ступінь захисту даних.

Для підключення використовується спеціальний код, який створюється на основі унікального коду ID самого комп'ютера. При установці з'єднання користувач визначає надаються права користувача. Таким чином, користувач який підключається до комп'ютера не може виконати операції, доступ до яких йому не було надано.

Віддалений доступ до комп'ютера Ammyu Admin також дозволяє передавати різні файли. Вбудований файловий менеджер підтримує передачу файлів розміром до 140 Тб. Більш того в разі втрати зв'язку Ammyu Admin має функцію відновлення передачі файлів, що дуже зручно для передачі великих файлів при нестабільному підключенні.

Ammyu Admin надає одне з найбільш зручних рішень в сфері віддаленого доступу. Високий рівень стабільності, а також швидкості з'єднання, забезпечують надійну роботу з віддаленими комп'ютерами. Більш того, Ammyu Admin має високий рівень безпеки передачі даних. Користувач отримує можливість повноцінної роботи з файлами віддаленого комп'ютера включаючи переміщення, копіювання, видалення, перейменування, а також створення папок. Що досить зручно при дистанційному навчанні та роботі з лабораторним обладнанням [23].

4.2 Методика роботи лабораторії з віддаленим доступом з дисципліни «Електронні прилади»

Отримавши ID та пароль до віртуальної лабораторії, користувач повинен завантажити і встановити на своєму комп'ютері програму Ammyu Admin. Запустивши програму, користувач повинен ввести ID та пароль. Перед користувачем відкривається робоча область керуючого комп'ютера, де він має доступ тільки до одного файлу.

У якому знаходяться файли з:

- методичними вказівками;
- інструкцією роботи з віртуальним приладом;
- віртуальним приладом.

На сам перед користувачу необхідно ознайомитися з методичними вказівками та інструкцією.

Методичні вказівки містять в собі теоретичні відомості що стосуються фотоприладів методи їх обчислення та використання. Методичні вказівки представлені у Додатку Г [24].

Інструкція роботи з віртуальним приладом містить в собі по крокові вказівки роботи з приладом.

- 1) Відкрити файл з віртуальним приладом .
- 2) Оглянути усі елементи віртуального приладу.

- 3) У рядку «ВИБІР СОМ-ПОРТ» вибрати потрібний порт (порт, до якого підключена потрібна плата вказує викладач).
- 4) Почати роботу з віртуальним приладом натиснувши на кнопку «Run».
- 5) Перевести перемикач «ПЕРЕМИКАЧ ЖИВЛЕННЯ» в положення «10 В».
- 6) Перемикач « СВІТЛОДІОД 1» ввімкнути.
- 7) Занотувати показники у рядках «ПОКАЗНИК ТОКУ mA» та «ПОКАЗНИК НАПРУГИ В».
- 8) Вимкнути перемикач « СВІТЛОДІОД 1» .
- 9) Ввімкнути перемикач « СВІТЛОДІОД 2».
- 10) Занотувати показники у рядках «ПОКАЗНИК ТОКУ mA» та «ПОКАЗНИК НАПРУГИ В».
- 11) Вимкнути перемикач « СВІТЛОДІОД 2» .
- 12) Ввімкнути перемикач « СВІТЛОДІОД 3».
- 13) Занотувати показники у рядках «ПОКАЗНИК ТОКУ mA» та «ПОКАЗНИК НАПРУГИ В».
- 14) Вимкнути перемикач « СВІТЛОДІОД 3» .
- 15) Перевести перемикач «ПЕРЕМИКАЧ ЖИВЛЕННЯ» в положення «15 В».
- 16) Перемикач « СВІТЛОДІОД 1» ввімкнути.
- 17) Занотувати показники у рядках «ПОКАЗНИК ТОКУ mA» та «ПОКАЗНИК НАПРУГИ В».
- 18) Вимкнути перемикач « СВІТЛОДІОД 1» .
- 19) Ввімкнути перемикач « СВІТЛОДІОД 2».
- 20) Занотувати показники у рядках «ПОКАЗНИК ТОКУ mA» та «ПОКАЗНИК НАПРУГИ В».
- 21) Вимкнути перемикач « СВІТЛОДІОД 2».
- 22) Ввімкнути перемикач « СВІТЛОДІОД 3».
- 23) Занотувати показники у рядках «ПОКАЗНИК ТОКУ mA» та «ПОКАЗНИК НАПРУГИ В».

- 24) Вимкнути перемикач « СВІТЛОДІОД 3» .
- 25) Вимкнути віртуальний прилад, натиснувши кнопку « STOP».
- 26) Закрити файл з віртуальним приладом.

5 ЕКОНОМІЧНЕ ОБГРУНТУВАННЯ

5.1 Техніко-економічне обґрунтування

Мета проекту - створення програми та стенду для дослідження фотоефекту. Основна мета складання продукту полягає в тому, щоб спростити процес вивчення фотоефекту, так як для проведення експериментів потрібно дороге обладнання. Даний продукт дозволить наочно вивчити фотоефект. Він допоможе вивчити теорію з даної теми, закріпити отримані знання і перевірити на практиці принцип роботи фотоефекту.

Для вивчення принципу фотоефекту потрібна велика кількість часу і коштів. Багато навчальних закладів не можуть дозволити собі провести наочні експерименти, тому вибір мною даної теми дуже актуальне. Програма та стенд з вивчення фотоефекту дозволяє економити час і кошти, так як для проведення експерименту потрібно тільки комп'ютер.

Лабораторія дослідження фотоефекту спростить роботу викладачів за поясненням даної теми.

Для вивчення теми необхідний лише комп'ютер, що помітно скорочує витрати на проведення наочних дослідів і робить процес навчання цікавим. Лабораторія дослідження фотоефекту передбачає деякі витрати на виконання робіт з розробки програмного продукту для ЕОМ, лабораторного стенду, та просування його на ринку. Плануючи витрати на ці роботи, я робив розрахунки на отримання деякого позитивного ефекту, який полягає в отриманні додаткового прибутку.

Якщо загальний результат виявляється позитивним, то проект є корисним.

Для визначення витрат, пов'язаних зі створенням лабораторії дослідження фотоефекту необхідно визначити за фактичними витратами значення:

- а) заробітної плати інженера - програміста (год., денна або місячна), грн .;
- б) частка накладних витрат на підприємстві;

в) час на складання програми (чол. / год., чол. / днів, чол. / міс).

Для визначення трудовитрат на складання програмного продукту необхідно весь трудовий процес розчленувати на складові частини і потім визначити норму робочого часу на продукт.

Основними методами вивчення витрат робочого часу є: хронометраж і фотографія робочого часу. У сучасних умовах широко використовується фотохронометраж.

Таблиця 5.1 Фотохронометраж складання програми та стенду з дослідження фотоефекту.

№ п / п	Елементи витрат часу	Час поточний	Час на етап
	1-ий день		
1	Початок роботи	9:00	10 мін
2	Підготовка робочого місця	9:10	20 мін
3	Складання алгоритму	9:30	30 мін
4	Складання програми	10:00	4 ч
5	Завершення роботи	14:00	
	2-ий день		
1	Початок роботи	9:00	10 мін
2	Підготовка робочого місця	9:10	20 мін
3	Складання алгоритму	9:30	30 мін
4	Складання програми	10:00	4 ч
5	Завершення роботи	14:00	

Продовження таблиці 5.1

№ п / п	Елементи витрат часу	Час поточний	Час на етап
	3,4,5-ий день		
1	Початок роботи	10:00	10 мін
2	Підготовка робочого місця	10:10	20 мін
3	Складання алгоритму	10:30	30 мін
4	Складання програми	11:00	3 ч
5	Завершення роботи	14:00	
	6, 7, 8-ий день		
1	Початок роботи	9:00	10 мін
2	Підготовка робочого місця	9:10	10 мін
3	Підбір інформації	9:20	40 мін
4	Складання програми	10:00	4 ч
5	Завершення роботи	14:00	
	9-ий день		
1	Початок роботи	9:00	10 мін
2	Підготовка робочого місця	9:10	10 мін
3	Розробка теоретичної частини програми	9:20	4 ч 40 мін
4	Завершення роботи	14:00	
	10-ий день		
1	Початок роботи	9:00	10 мін
2	Підготовка робочого місця	9:10	10 мін
3	Розробка теоретичної частини програми	9:20	4 ч 40 мін
4	Завершення роботи	14:00	

Продовження таблиці 5.1.

№ п / п	Елементи витрат часу	Час поточний	Час на етап
	11-ий день		
1	Початок роботи	9:00	10 мін
2	Підготовка робочого місця	9:10	10 мін
3	Розробка практичної частини програми	9:20	4 ч 40 мін
4	Розробка лабораторного стенду		
5	Завершення роботи	14:00	
	12-ий день		
1	Початок роботи	9:00	10 мін
2	Підготовка робочого місця	9:10	10 мін
3	Складання тесту програми	9:20	4 ч 40 мін
4	Складання лабораторного стенду	9:20	
5	Завершення роботи	14:00	
	13, 14-ий день		
1	Початок роботи	9:00	10 мін
2	Підготовка робочого місця	9:10	10 мін
3	Редагування програми	9:20	4 ч 40 мін
4	Редагування лабораторного стенду		
5	Завершення роботи	14:00	
	15-ий день		
1	Початок роботи	9:00	10 мін
2	Підготовка робочого місця	9:10	10 мін
3	Налагодження і редагування програми	9:20	4 ч 40 мін
4	Налагодження і редагування лабораторного стенду		
5	Завершення роботи	14:00	
Разом			72

На підставі даних таблиці 5.1 отримуємо $T_n = 72$ чол. / год.

Витрати на покупку комплектуючих деталей обчислюються виходячи з цін на радіоелектронні елементи в магазині "ТЕХНОГЛОБУС", і представлені в табл. 5.2.

Таблиця 5.2 - Кошторис витрат на покупні деталі

Куповані вироби		Кількість РЕА	Ціна за одиницю, грв. / шт.	Сума на РЕА, грв.
Найменування	Технічна характеристика			
Фотодіод	HPDB1B-48D	1	54	54
Світлодіод	C503C-WAN	3	32	32
Резистор	C2-33 0,125 Вт	7	3,5	25
Плата	Arduino UNO	1	175	175
Релейний модуль	1-Channel 24V Relay Module	1	40	40
Регульований підвищувач напруги	dc-dc mt3608	2	25	50
Разом				396

Таким чином витрати на покупні вироби складуть 396 грн.

Середня заробітна плата інженера – програміста (годинна, денна або місячна):

$$З_{пл_{ср}} = З_{пл} / (20 * 8);$$

де $З_{пл}$ - заробітна плата інженера - програміста (6500грн.);

20 - число робочих днів у місяці;

8 - число робочих годин на день.

$$З_{пл_{ср}} = 6500 / (20 * 8) = 40,62 \text{ грн,}$$

Вартість одного машинного часу розраховується за формулою:

$$C_{\text{мч}} = ((C_{\text{бал.}} + C_{\text{покуп.}}) / (T_{\text{сл}} * D_{\text{р}} * Ч_{\text{д}})) + C_{\text{ел.ен.}}$$

де $C_{\text{бал.}}$ - балансова вартість персонального комп'ютера, (15000грн.);

$C_{\text{покуп.}}$ - вартість на покупні вироби (396грн.);

$T_{\text{сл}}$ - нормативний термін експлуатації ПК (за паспортними даними) (5 років);

$D_{\text{р}}$ - число робочих днів у році (252 дня);

$Ч_{\text{д}}$ - число годин роботи ПК в день (6 год.);

$C_{\text{ел.ен.}}$ - вартість електроенергії.

Вартість електроенергії, яка розраховується за формулою:

$$C_{\text{ел.ен.}} = P_{\text{ел.}} * Ц_{\text{квт.}}$$

де $P_{\text{ел.}}$ - витрата електроенергії на годину (за паспортними даними ПК), (0,3 квт);

$Ц_{\text{квт.}}$ - вартість (ціна) квт. години електроенергії (0,9грн.).

Вартість електроенергії однієї години:

$$C_{\text{ел.ен.}} = 0,3 * 0,9 = 0,27 \text{ грн.}$$

Тоді вартість одного машинного часу:

$$C_{\text{мч}} = ((15000 + 396) / (5 * 252 * 6)) + 0,27 = 2,3 \text{ грн.}$$

Поточні витрати на створення продукту можуть бути визначені наступним чином:

$$A_o = T_n * 3 / \text{пл}_{\text{ср}} * (1 + K_{\text{есн}} + K_{\text{нр}}) + C_{\text{мч}} * T_{\text{мо}};$$

де T_n - час на складання продукту (72 чол. / год.)

3 / $\text{пл}_{\text{ср}}$ - середня заробітна плата інженера - програміста (годинна, денна або місячна) (40,62грн.)

$K_{\text{ЄСН}}$ - коефіцієнт єдиного соціального податку (0,22). Закону України №2464-VI.;

$K_{\text{НР}}$ - Коефіцієнт накладних витрат, який використовується на підприємстві.(0,2);

$C_{\text{мч}}$ - вартість однієї години машинного часу (2,3 грн.);

$T_{\text{мо}}$ - витрати машинного часу на налагодження (5 ч).

$$A_0 = 72 * 40,62 * (1 + 0,22 + 0,2) + 2,3 * 5 = 4164,63 \text{ грн.}$$

На отриману суму нараховується ПДВ за ставкою податку 20%.

Розрахунок поточних витрат з урахуванням ПДВ:

$$A_0 = 4164,63 * 1,2 = 4997,56 \text{ грн.}$$

Ефект від впровадження у виробництво програмного продукту при рівних умовах виробництва буде визначатися як різниця витрат пов'язаних з використанням замінної інформаційної технології і нової.

Ефект розраховується за формулою:

$$E = Z_1 - Z_2 ;$$

де Z_1 - елемент виробничих витрат, пов'язаних з використанням інформаційної технології (або) традиційного способу вирішення завдання;

Z_2 - елемент виробничих витрат, пов'язаних з використанням нової інформаційної технології.

Витрати можуть бути визначені за формулою:

$$Z = Z_{\text{пл.ср}} * T_p (1 + K_{\text{ЄСН}} + K_{\text{НР}}) ;$$

де $Z_{\text{пл.ср}}$ - середня заробітна плата викладача (год., денна, місячна), (55,05 грн.)

Тр - трудомісткість рішення задачі з розрахунку за рік в: чол. / год., чол. / дн., чол. / міс. (До впровадження 4 год., після впровадження 1 год.);

$K_{\text{ЄСН}}$ - коефіцієнт єдиного соціального податку (0,22). Закону України №2464-VI.;

$K_{\text{нр}}$ - Коефіцієнт накладних витрат, який використовується на підприємстві.(0,2).

Середня заробітна плата викладача (годинна, денна, місячна).

$$З_{\text{пл.ср}} = З_{\text{пл}} / (20 * 8) ;$$

де $З_{\text{пл}}$ - заробітна плата викладача (8100грн .) ;

20 - число робочих днів у місяці;

8 - число робочих годин на день.

$$З_{\text{пл.ср}} = 8100 / (20 * 8) = 55,05 \text{ грн,}$$

Елемент виробничих витрат, пов'язаних з використанням інформаційної технології (або) традиційного способу вирішення завдання.

$$З_1 = 55,05 * 4 * 10 * (1 + 0,22 + 0,2) = 3126,84 \text{ грн.}$$

У цій формулі 4 - це кількість годин, які викладач витрачав на пояснення теми. 10 - це кількість груп , які навчаються даної теми протягом одного місяця.

Елемент виробничих витрат, пов'язаних з використанням нової інформаційної технологією.

$$З_2 = 55,05 * 1 * 10 * (1 + 0,22 + 0,2) = 781,71 \text{ грн.}$$

У цій формулі 1 - це кількість годин, які викладач витрачає на пояснення теми, використовуючи даний програмний продукт. 10 - це кількість груп, які навчаються даної теми протягом одного місяця.

Ефект від впровадження у виробництво програмного продукту:

$$E = 3126,84 - 781,71 = 2345,13 \text{ грн.}$$

Термін окупності продукту розраховується за формулою:

$$T_{\text{ок}} = A / E,$$

$$T_{\text{ок}} = 4997,56 / 2345,13 = 2,13$$

де А - витрати, пов'язані зі створенням продукту (4997,56 грн.)

Е - ефект від використання продукту (2345,13 грн.).

Для оцінки ефективності проекту використовуються наступні показники:

- чистий дохід (ЧД);
- чиста поточна вартість (ЧПВ);
- індекс прибутковості (ІП);
- внутрішня норма прибутковості (ВНП).

Чистий дохід підприємства від реалізації інвестиційного проекту являє собою різницю між надходженнями (припливом коштів) і виплатами (відтоком коштів) підприємства в процесі реалізації проекту стосовно кожного інтервалу планування.

Виплати підприємства діляться на капітальні (одноразові) витрати і поточні витрати.

До капітальних витрат відносяться витрати, які спрямовані на створення виробничих потужностей і розробку продукції.

Поточні витрати - це витрати на придбання сировини, матеріалів комплектуючих, оплату праці працівників підприємства, інші види витрат, що

відносяться на собівартість продукції. Поточні витрати здійснюються протягом всього терміну життя проекту.

Надходження - це результат діяльності підприємства в процесі здійснення проекту у вигляді виручки від реалізації виробленої продукції.

Дисконтування - це процес знаходження величини грошової суми на конкретний момент часу по її відомому або передбачуваному значенню в майбутньому, виходячи із заданої відсоткової ставки.

Норма дисконтування (ставка дисконту) розглядається в загальному випадку як норма прибутку на вкладений капітал, тобто як відсоток прибутку, який інвестор або підприємство хоче отримати в результаті реалізації проекту.

Для визначення прибутковості проекту використовується метод розрахунку дисконтованої (чистої) вартості капіталу (або дисконтованого доходу). Чистий дисконтований прибуток - це дохід, отриманий від вкладення капіталу на розробку проекту. При розрахунку йде порівняння ефекту від використання створеного продукту з тим, який дохід можна отримати від вкладення тих же коштів в банк.

При цьому:

1) Заздалегідь задається мінімально допустима ставка відсотка окупності (прибутковий відсоток - норма дисконту), при якій об'єкт інвестування вважається прибутковим. Такий ставкою є ставка банківського відсотка. Вона дорівнює 15% і умовно вважається незмінною для всього строку служби об'єкта інвестування ($\epsilon = 0,15$)

2) Здійснюється дисконтування (приведення до базисного моменту часу) на основі розрахункової ставки відсотка з поточних надходжень, отриманих після придбання об'єкта інвестування (інакше - дисконтований ефект):

n

$$E_d = \sum_{t=1}^n \frac{E_t}{(1 + \epsilon)^t}$$

де n - термін служби об'єкта інвестування (для програмних продуктів зазвичай 3 роки);

t - номер кроку розрахунку ($= 1, 2, 3$),

$E_1 = 2345,13$ грн. - поточні річні надходження в році t .

$$E_d = \frac{E_1}{(1 + \epsilon)^1} + \frac{E_1}{(1 + \epsilon)^2} + \frac{E_1}{(1 + \epsilon)^3}$$

$$E_d = \frac{2345,13}{(1 + 0,15)^1} + \frac{2345,13}{(1 + 0,15)^2} + \frac{2345,13}{(1 + 0,15)^3} = 5364,7 \text{ грн.}$$

Умовно можна вважати, що E_1, E_2, E_3 рівні між собою (вони розраховуються за формулою $E = Z_1 - Z_2$ ці величини можуть коригуватися з урахуванням інфляції).

3) Розраховується дисконтована (чиста) вартість витрат на придбання (створення) об'єкта інвестування з величини виручки від інвестицій:

$$K = E_d - A_0;$$

$$K = 5364,7 - 4997,56 = 367,13 \text{ грн.}$$

де E_d - дисконтований ефект (5364,7 грн.);

A_0 - витрати, пов'язані зі створенням продукту (4997,56 грн.).

Позитивна величина поточної вартості капіталу означає, що проект ефективний (при даній нормі дисконту). Чим більше ця величина, тим ефективніше проект. Негативна величина поточної вартості капіталу означає збитковість інвестицій. Отриманий показник також називається інтегральним ефектом.

Індекс прибутковості відношення наведених ефектів до інвестицій:

$$ІП = E_d / A_0 ;$$

$$ІП = 5364,7 / 4997,56 = 1,07;$$

де Ед - дисконтований ефект(5364,7 грн.);

А0 - витрати, пов'язані зі створенням продукту (4997,56 грн.).

Якщо $ІП > 1$, то проект ефективний, в іншому випадку - неефективний, так як ІД при створенні продукту більше 1, то проект є ефективним.

У зв'язку з тим, що ІП при обчисленні ефективності продукту склав 1,07, можна зробити висновок, що даний програмний продукт є ефективним.

Так як даний продукт буде надаватися для користування навчальним закладам та іншим установам, які придбали право та дозвіл на користування. В даному розділі була проведена оцінка ефективності впровадження проекту і визначена прибутковість проекту. Розрахунок дав позитивні величини результатів, що свідчить про ефективність та прибутковість впровадженого проекту.

6 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

6.1 Аналіз потенційних небезпечних чинників при проведенні науково-дослідної роботи

При проведенні розробки віртуальної лабораторії з віддаленим доступом, проектування проводиться з використанням персонального комп'ютера. На інженерів, які працюють на персональних комп'ютерах (ПК), постійно або періодично діють наступні небезпечні і шкідливі фактори:

- забруднення повітря шкідливими речовинами, пилом і позитивними аероіонами;
- невідповідність нормам параметрів мікроклімату;
- широкий спектр випромінювання від дисплея;
- підвищений рівень електромагнітних випромінювань;
- недостатня освітленість робочої зони;
- підвищена яскравість монітора (недостатня освітленість);
- знижена контрастність;
- тривале перебування в тому ж положенні, і повторення тих же рухів призводить до синдрому тривалих статичних навантажень (СДСН);
- невідповідність ергономічних характеристик обладнання нормованим величинам;
- монотонність праці;
- нервово-емоційні стресові навантаження;
- небезпека виникнення пожежі.

Крім того, відповідно до ГОСТ 12.2.003-74 «Виробниче обладнання». Загальні вимоги безпеки та ГОСТ 12.1.019-79 «Електробезпека» можна виділити небезпечні виробничі фактори, характерні для робочого місця оператора: небезпечний рівень напруги в електричному ланцюзі, замикання якого може відбутися через тіло оператора, а також підвищений рівень статичного електрики при неправильно спроектованій робочій зоні.

6.2 Заходи щодо забезпечення безпеки

Заходи щодо техніки безпеки - це система організаційних заходів і технічних засобів, що запобігає або зменшує вплив на працюючих небезпечних і шкідливих виробничих факторів. Відповідно до класифікації приміщень за ступенем безпеки ураження електричним струмом ГОСТ 12.1.019-79 «Електробезпека», приміщення конструкторського бюро (КБ) відносяться до категорії приміщень без підвищеної небезпеки, тому що відносна вологість повітря не перевищує 75%, температура не більше 27 ° С, відсутня хімічно агресивне середовище. Робочі місця інженерів відповідають ГОСТ 12.2.003-74 «Виробниче обладнання». Загальні вимоги безпеки та ГОСТ 12.1.019-79 «Електробезпека». Складові частини виробничого обладнання (у тому числі кабелі, проводи тощо) виконані з таким розрахунком, щоб виключити можливість їх випадкового пошкодження, яке викликає небезпеку. Конструкцією виробничого обладнання передбачений захист від ураження електричним струмом. Для доступу до струмоведучих частин при ремонті або огляді необхідно розкрутити гвинти і зняти кришку. Металеві частини виробничого обладнання, які можуть в результаті пошкодження ізоляції опинитися під електричною напругою небезпечної величини, заземлені або занулені. Провідники занулення обрані так, щоб при замиканні на корпус або на нульовий провід виникав струм короткого замикання, сила якого перевищує не менше ніж у три рази номінальну силу струму плавкої вставки найближчого запобіжника.

Дисплей комп'ютера здатний накопичувати статичний заряд. Небезпека зарядів статичного електрики виявляється в можливості виникнення електричних зарядів в просторі, вплив на обслуговуючий персонал і порушенні нормального ходу робочого процесу. Вплив статичного електрики на обслуговуючий персонал позначається в розрядах статичних зарядів через людину і вплив електростатичного поля. Безпосередньо струм розряду не є

небезпечним, тому що час його протікання через тіло людини мале (дорівнює декількох мілісекунд). Такі короткочасні імпульси можуть викликати електричні удари, які супроводжуються болючим уколом, що призводять до переляку.

Для зниження рівня статичної електрики передбачено екранування джерела статичної електрики (монітора) захисним екраном і заземлення екрана відповідно до ДСанПіН 3.3.2.007-98 "Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин".

У схемі електричних ланцюгів передбачено пристрій, який централізовано відключає від живильної мережі всі електричні ланцюги. У відділі всі мережеві розетки мають написи зі значенням напруги і застережливий знак.

До основних ергономічних завдань організації робочого місця згідно ДСТУ ГОСТ 12.2.061: 2009 «Загальні вимоги безпеки до робочих місць» відносяться:

- визначення просторових параметрів робочого місця та його елементів, які відповідають антропометричним характеристикам контингенту працюючих;
- оптимальне розміщення елементів робочого місця щодо користувача на основі аналізу його діяльності.

Устаткування і організація робочого місця працюючих з ПК забезпечує відповідність конструкції всіх елементів робочого місця та їх взаємного розташування ергономічним вимогам з урахуванням характеру і особливостей трудової діяльності (ГОСТ 12.2.032-78 «Система стандартів безпеки праці». Робоче місце при виконанні робіт сидячи. Загальні ергономічні вимоги, ГОСТ 22.269-76 Система «людина-машина». Робоче місце оператора. Взаємне розташування елементів робочого місця. Загальні ергономічні вимоги, ГОСТ 21.889-76 Система «людина-машина» Крісло людини-оператора. Загальні ергономічні вимоги).

6.3 Заходи щодо забезпечення виробничої санітарії та гігієни праці

Для нормальної експлуатації машин електронної та обчислювальної техніки і високопродуктивної праці робітників та інженерно-технічного персоналу в процесі проектування пристрою передбачений наступний комплекс заходів:

- а) забезпечення оптимального мікроклімату;
- б) застосування оптимального освітлення;
- в) створення оптимальних умов праці на робочому місці;
- г) зниження психоемоційного та фізичного навантаження.

Загальні санітарно-гігієнічні вимоги до повітря приміщення, в якому розташоване КБ відповідає вимогам ГОСТ 12.1.005-88. У приміщенні застосована система кондиціонування, а також регулярно проводиться провітрювання. У приміщенні регулярно здійснюється вологе прибирання.

Згідно ДБН В.2.5-28-2006 в приміщенні застосований змішаний тип освітлення, що відповідає II-III розряду зорової роботи. Як штучного джерела освітлення застосовані лампи денного світла, які разом зі світильниками вмонтовані в стелю. У світлий час доби в основному застосовується природне освітлення, яке забезпечується наявністю вікон. При необхідності (захист від прямих сонячних променів), вікна закривають жалюзі. Увечері, або в похмуру погоду персонал включає штучне освітлення.

Правильна організація робочого місця - створення на робочому місці необхідних умов для продуктивної роботи і виконання роботи високої якості, при найбільш повному використанні обладнання, малих розтратах фізичної та емоційної енергії працівника, підвищення змістовності та привабливості роботи, збереження здоров'я працюючих.

При розміщенні обладнання витримуються необхідні проміжки між обладнанням, відстані від стін, які забезпечують свободу пересування людей, зручність виконання робіт і безпеку працюючих.

Робочі місця операторів ПК, а так само ділянки підготовки технічних носіїв інформації розташовані в ряди. Розташування місць може бути дворядним, трирядним, чотирирядним, а також прямим і поперечним.

На організацію роботи впливає конструкція і параметри основного і допоміжного обладнання, які відповідають вимогам ергономіки: оптимальному розподілу функцій в системі людина - машина. Конструкція обладнання відповідає антропометричним і психофізіологічним даними організму працюючого. Витримані допустимі параметри санітарно-гігієнічних умов праці, а також безпеки експлуатаційного обладнання.

Конструкція робочого столу забезпечує оптимальне розміщення на робочій поверхні використовуваного обладнання, враховуючи його масогабаритні і конструктивні особливості.

Висота робочої поверхні столу регулюється в межах 680-800 мм, при відсутності такої можливості висота робочої поверхні столу складає 725 мм. Модульними розмірами робочої поверхні столу для ПК, розраховуються конструктивні розміри, слід вважати: ширину 800, 1000, 1200, 1400 мм, глибину 800 і 1000 мм при нерегульованій його висоті, що дорівнює 725 мм.

Робоче місце обладнане підставкою для ніг, має ширину не менше 300 мм, глибину не менше 400 мм, регулювання по висоті в межах до 150 мм і за кутом нахилу опорної поверхні підставки до 200. Підставка має рифлену поверхню.

Конструкція робочого стільця (крісла) підтримує раціональну робочу позу при роботі з ПК, дозволяє змінювати позу з метою зниження статистичного напруження м'язів шийно-плечової області і спини для попередження стомлення. Робочий стілець (крісло) є підйомно - поворотним і регульованим по висоті і кутам нахилу сидіння і спинки, а також відстані від переднього краю сидіння.

Конструкція стільця повинна забезпечувати максимум результатів праці. Досягти цього можна, лише освоївши раціональні методи і прийоми праці на

робочому місці. Тільки вони дозволяють виконати задану роботу якісно, в мінімальні терміни і без зайвої напруги.

Специфіка роботи таких працівників полягає в великих зорових навантаженнях разом з маленькою активністю, монотонністю виконаних операцій, вимушеної робочої позою. Ці фактори негативно відбиваються на самопочутті працюючого.

При організації робочого місця дуже важливим фактором є робоча поза працівника, тобто положення його корпусу, голови, рук і ніг по відношенню до знарядь праці. Якщо працівник працює сидячи, йому необхідно забезпечити правильну і зручну посадку, яка досягається пристроєм опори для спини, рук, ніг, правильною конструкцією сидіння, яке сприяє рівномірному розподілу маси тіла.

Всі матеріальні елементи робочого столу ділять на предмети постійного, тимчасового користування і з урахуванням цього розташовують у видатному порядку на місцях постійного зберігання, це економить трудові рухи і сили працюючого.

Клавіатуру комп'ютера краще розташовувати на відстані 10-15 мм від краю столу, тоді зап'ястя рук будуть спиратися на стіл. Бажано придбати спеціальну підставку під зап'ястя, що, як стверджують медики, допоможе уникнути хвороби кистей.

Для ефективного використання маніпулятора типу «миша» застосовується спеціальний «килимок» - планшет. Килимок - планшет задовольняє основним критеріям: по-перше, добре тримається на поверхні столу, по-друге, матеріал верхньої поверхні планшета забезпечує вільне ковзання миші.

Введення текстової інформації з клавіатури полегшують підставки для документів. Вони можуть або кріпитися до монітора, або встановлюватися безпосередньо на столі.

Інструмент, обладнання та предмети праці знаходяться на відстані 560-750 мм на рівні рук працівника, так їх використання не призводить до зайвих

рухомим нахилом. Важливим елементом раціонального планування робочого місця є облік індивідуальних антропометричних психофізіологічних даних працюючого.

Робочі місця обладнані відповідними меблями та інвентарем, які відповідають найбільш комфортабельним умовам роботи і вимогам фізіології, психології та естетики.

Нижче представлений розрахунок повітрообміну в приміщенні КБ.

У прикладі необхідно визначити необхідний повітрообмін і його кратність для вентиляційної системи КБ, що має довжину $a = 10$ м, ширину $b = 6$ м, висоту $h = 3,5$ м. У повітряне середовище КБ виділяється пил в кількості $W = 1,5$ г / ч (для даного виду пилу ГДК = 4 мг / м³), концентрація пилу в робочій зоні $C_{p.з.} = 2,8$ мг / м³, в припливного повітрі $C_{п.} = 0,3$ мг / м³, концентрація пилу в повітрі, що видаляється з КБ дорівнює концентрація її в робочій зоні ($C_{yx} = C_{p.з.}$), тобто пил рівномірно розподілена в повітрі.

Кількість повітря, що забирається з робочої зони місцевими відсмоктувачами, так само $G_m = 50$ м³ / год.

Рішення даного завдання має наступний алгоритм:

1) Розрахунок обсягу КБ здійснюється за формулою 6.1:

$$V = a \times b \times h, \text{ м}^3$$

$$V = 10 \times 6 \times 3,5 = 210 \text{ м}^3$$

2) Розрахунок необхідного повітрообміну здійснюється за формулою 6.2:

$$G_{np} = G_m + \frac{W - G_m(C_{p.з.} - C_{п.})}{C_{p.з.} - C_{п.}}, \text{ м}^3/\text{ч}$$

$$G_{np} = 50 + \frac{1500 - 50(2,8 - 0,3)}{2,8 - 0,3} = \frac{1375}{2,5} = 550 \text{ м}^3/\text{ч}$$

3) Кратність повітрообміну в КБ розраховується за формулою 6.3:

$$K = \frac{G_{np}}{V}, \text{ 1/ч}$$

$$K = \frac{550}{210} = 2,6 \text{ 1/ч}$$

Тобто за 1 годину повітря в КБ повинен обмінюватися 2,6 рази.

6.4 Заходи з пожежної безпеки

Відповідно до ДБН В.2.5-13-98 «Пожежна автоматика будинків і споруд» найбільш частими причинами пожеж є наступні:

- порушення режимних вимог;
- самозаймання і самозапалювання;
- розряди статичної та атмосферної електрики;
- несправність і неправильна експлуатація обладнання.

Джерелами загоряння можуть бути коротке замикання, перевантаження проводів, перегріті опірні поверхні, несправність апаратури. Окислювачем, як правило, служить кисень. Для електронних пристроїв характерно часта поява джерел відкритого вогню при коротких замиканнях, пробоях і перевантаженнях. Однак потужність і тривалість дії цих джерел запалення порівняно малі, тому горіння, як правило, не розвивається.

Для відводу надмірної теплоти від ПК передбачені системи вентиляції і кондиціонування повітря.

Одним із важливих заходів щодо пожежної безпеки є установка в приміщенні КБ пожежної сигналізації. Вона дає можливість швидко погасити вогнище загоряння. У приміщенні встановлені променеві електричні системи сигналізації ЕРС типу ТОЛ 10/100 на 10-100 променів. В якості сповіщувачів застосовані теплові і димові сповіщувачі типу КД-1.

Приміщення КБ відноситься до категорії «Д» по вибухо- і пожежобезпеки відповідно СНиП II-2-80 "Протипожежні норми проектування будівель і споруд".

Будівля, в якому зазвичай знаходиться КБ, є спорудою з несучими і загороджували конструкціями з природного або штучного каміння, бетону, залізобетону з використанням негорючих матеріалів. З огляду на конструктивні характеристики споруди і категорію провадження у вибухо- і пожежобезпеки, ступінь вогнестійкості повинна бути не нижче III.

У разі виникнення пожежі, евакуація працівників може проводитися через пожежні виходи. Для уникнення пожеж на виробництві:

- розроблені комплексні заходи щодо забезпечення пожежної безпеки;
- відповідно до нормативних актів з пожежної безпеки розроблені положення та інструкції, які діють в межах підприємства, установи та організації, здійснюється постійний контроль за їх виконанням;
- забезпечується дотримання протипожежних вимог, норм, правил, а також виконання вимог, розпоряджень і постанов органів державного пожежного нагляду;
- утримуються в справному стані засоби протипожежного захисту, пожежна техніка, обладнання та інвентар, не допускається їх використання не за призначенням;
- представляються за вимогою державної пожежної охорони відомості та документи про стан пожежної безпеки об'єктів і виробленої ними продукції;
- проводяться заходи щодо впровадження автоматичних засобів виявлення та гасіння пожеж, і використання для цієї мети виробничої автоматики.

Усі працівники при прийнятті на роботу щорічно за місцем роботи проходять інструктажі з питань пожежної безпеки.

Приміщення обладнане наступними елементами пожежогасіння:

- Вогнегасник ОУ- 3;
- Вогнегасник ОП-1 «Момент».

Така кількість вогнегасників відповідає вимогам ISO3941 - 77 «Пожежі. Класифікація», якими передбачене обов'язкова наявність двох вогнегасників на 100 м2 площі для приміщень типу конструкторських бюро.

6.5 Заходи з цивільного захисту

Заходами щодо евакуації називаються розосередження і евакуація населення з категорійних міст у заміську зону. Вони організовуються і

проводяться відповідно до плану ЦО району (об'єкта) та розпоряджень старшого начальника ЦО.

Розосередження - організований вивіз з міст та інших населених пунктів і розміщення в заміській зоні, вільній від роботи зміни робітників і службовців тих об'єктів, які продовжують роботу в умовах НС. Розосереджується робітники і службовці після розселення в заміській зоні позмінно виїжджають в город для роботи на підприємствах, а після роботи повертаються в заміську зону для відпочинку. Евакуація - організований вивіз або вивід з населених пунктів і розміщення в заміській зоні решти населення. На відміну від розосереджених, евакуйовані постійно проживають в заміській зоні до особливого розпорядження.

Заміська зона - територія, розташована за межами зон можливого ураження в містах.

Кожному підприємству і установі міста, з якого планується розосередження і евакуація, в заміській зоні призначається район розміщення населення, який включає один або кілька розташованих поруч населених пунктів.

Видалення районів розосередження від міста має забезпечувати безпеку робітників і службовців, але час на переїзд в город і повернення назад повинен бути мінімальний.

Відповідальність за евакуацію робітників і службовців та їх сімей в призначені райони покладається на начальників ЦО і штабів ЦО підприємств і установ.

Розселення робітників і службовців виконується з дотриманням виробничого принципу, що забезпечує цілісність підприємства, полегшує відправку робочих змін на об'єкт і забезпечення населення харчуванням і медичним обслуговуванням.

Евакуйоване населення розміщують в більш віддалених районах заміської зони, а населення, евакуйоване з зон можливих стихійних лих, - в населених пунктах, що знаходяться поблизу цих зон.

Розосередження і евакуація в багато разів знижують щільність населення і, отже, можливі втрати.

Розосередження і евакуація здійснюються:

- робітників і службовців і членів їх сімей - за виробничим принципом (по лінії об'єктів);
- населення не пов'язаного з виробництвом - за територіальним принципом (тобто за місцем проживання через домоуправління і житлово-експлуатаційні контори).

Щоб евакуація пройшла організовано, її завчасно планують місцеві адміністративні органи і створені при них евакуаційні комісії; попередньо визначаються склад, місця розташування і порядок роботи збірних евакуаційних пунктів (ЗЕП), а в сільській місцевості, куди вивозиться населення, створюються прийомні евакуаційні комісії та приймальні евакуаційні пункти (ПЕП).

ЗЕП призначені для збору, реєстрації та організованого відправлення населення. Коли людей вивозять залізничним і водним транспортом, ЗЕП розміщують поблизу станцій, портів і на підприємствах, які мають залізничні під'їзні шляхи, морські, річкові причали.

На кожному підприємстві і установі повинні бути евакуаційні списки, які разом з паспортами є основними документами для обліку, розміщення і забезпечення всім необхідним в районах розселення.

Евакуація може здійснюватися усіма видами транспорту і пішки. Для швидкої евакуації застосовується комбінований спосіб її проведення, тобто перший порядок об'єднується з максимальним використанням усіх видів транспорту.

Коли надійшло розпорядження про початок розосередження і евакуації, начальник ГО повідомляє про це начальникам виробничих підрозділів, останні оповіщають робітників і службовців, а ті - членів своїх сімей.

Для підтримки громадського порядку на об'єктах ЗЕП і ПЕП, на станціях (пристанях, пунктах) посадки і висадки, в місцях розселення в заміській зоні

виставляються пости охорони громадського порядку, організовується патрулювання. До виконання цих заходів залучаються формування охорони громадського порядку (команди, групи), що створюються з відомчої воєнізованої і сторожової охорони та добровільних народних дружин.

ВИСНОВОК

У процесі виконання дипломного проекту був розроблений проект віртуальної лабораторії, яку можна використовувати для отримання навичок роботи з фотоприладами з дисципліни «Електронні прилади».

У цій роботі була розроблена структурна схема віртуальної лабораторії, яка складається з трьох модулів:

- експериментальна установка;
- керуючий пристрій;
- інтерфейс користувача.

В якості експериментальної установки було використана установка для зняття вольт амперних характеристик та світлових характеристик фото приладів. Для виконання лабораторної роботи «Дослідження фотопровідності напівпровідників та фотоелектричних властивостей р-п переходу». Дана установка була перетворена для цифрового управління. В роботі представлена електрична принципова схема перетвореної установки.

Крім того в даній роботі представлені алгоритми програм для керуючого ПК і мікроконтролера. Також запропонована для використання програма для інтерфейсу користувача. Вона дозволяє відтворювати вже на керуючому ПК віртуальний прилад за зразком розробленого, що може бути використано в дисципліні «Технологія віртуальних приладів та систем аналізу».

Програмне забезпечення для керуючої плати Arduino UNO та програми Ammuu Admin знаходиться у безкоштовному користуванні на офіційних сайтах. А програмне забезпечення для LabVIEW є у користуванні на кафедрі ІТЕЗ. Отже, усе програмне забезпечення для віртуальної лабораторії не потребує матеріальних витрат на придбання.

Таким чином, віртуальні лабораторії дозволяють студентам отримати практичний навик роботи з реальними пристроями, перебуваючи за межами навчального закладу. Це не тільки покращує якість освіти, завдяки тому, що з'являється більше часу приділяти на вивчення теорії в освітній установі, а й

дозволяє долучити людей з обмеженими здібностями до віддаленого процесу навчання. Крім того, користувачі отримують можливість працювати з реальними пристроями, не піддаючи їх будь-яким пошкодженням, та спостерігати за сигналами і процесами, що протікають в ході роботи в режимі реального часу.

З вищезазначеного можна зробити висновок, що віртуальна лабораторія може використовуватися на кафедрі ІТЕЗ, з мінімальними витратами часу і коштів.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Про освіту [Електронний ресурс]: Закон України [прийнятий Постановою Верховної Ради України № 2145-VIII від 5 вересня 2017 року]. - Режим доступу: <http://zakon.rada.gov.ua/laws/show/2145-19>
2. Бобрівник, К.Є. Проектування віртуальної навчальної лабораторії для студентів технічно-технологічних спеціальностей / К.Є. Бобрівник, М.В.Гладка, М.О. Кіктєв // Енергетика і автоматика. - 2014. - № 3. - С. 18-23. - Режим доступу: http://nbuv.gov.ua/UJRN/eia_2014_3_5
3. Разработка проекта виртуальной лаборатории для дистанционного образования [Текст]: матеріали IV Регіональної молодіжної конференцій «Мій вибір – НАУКА!», 2017 р.: тези доповідей / [В.В. Ширяева (відпов.)]: - Режим доступу: <http://elibrary.asu.ru/xmlui/bitstream/handle/asu/3619/vkr.pdf>
4. Винокуров, А.Ю. Использование технологий виртуализации в учебном процессе - Режим доступу: <http://www.ict.edu.ru/vconf/files/8914.pdf>
5. Electronics Workbench.[Електронний ресурс]: Теоретические основы электротехники и электроники - Режим доступу: <http://bourabai.kz/toe/1/1-1.htm>
6. Система моделирования. [Електронний ресурс]: ISIS Proteus - Режим доступу:<http://easyelectronics.ru/sistema-modelirovaniya-isis-proteus-bystryj-start.html>.
7. ЕлектроХоббі. [Електронний ресурс]: Proteus – программа для автоматического проектирования - Режим доступу: <https://electrohobby.ru/proteus.html>.
8. Сайт Паяльник [Електронний ресурс]: MPLAB - Режим доступу: <http://cxem.net/software/mplab.php>.
9. СамГТУ кафедра Электромеханики и промышленной автоматки // [Електронний ресурс]: Обзор систем моделирования и инженерных расчетов, применяемых в нефтегазовой отрасли. - Режим доступу: <http://sstu.syzran.ru/epa/docs/ITiOvNGO/4.2.pdf> .

10. CadSoft EAGLE Importer. [Электронный ресурс]: Altium Designer - Режим доступа: [https://www.altium.com/documentation/18.0/display/ADES/\(\(EAGLE+Import\)\)_AD](https://www.altium.com/documentation/18.0/display/ADES/((EAGLE+Import))_AD).

11. Intec. [Электронный ресурс]: Altium Designer - Режим доступа: http://cxem.net/software/altium_designer.php.

12. Сайт Паяльник. [Электронный ресурс]:- Режим доступа: <http://cxem.net/software/micro-cap.php>.

13. MATLAB [Электронный ресурс]:- Режим доступа: <http://matlab.ru/products/matlab> .

14. Горшков, М.Д. Виртуальные симуляторы: обзор, устройство и классификация. //УВК «Ментор Медикус» ГМУ им. И.М. Сеченова. - с. 17 - 26. Режим доступа: http://www.medsim.ru/file/20171/VIRTUAL_REALITY_SIMULATOR_SREVIEW_CONSTRUCTION_CLASSIFICATION.pdf.

15. Zillion. [Электронный ресурс]:Виртуальные лаборатории для дистанционного - Режим доступа: <http://zillion.net/ru/blog/247/virtual-nyie-laboratorii-dlia-dstantsionnogho-obuchieniia-vuzov-i-shkol> .

16. Virtulab. [Электронный ресурс]: Виртуальная лаборатория- Режим доступа: <http://www.virtulab.net/> .

17. VALUE @ Amrita. [Электронный ресурс]: Amrita Vishwa Vidyapeetham Режим доступа: <http://vlab.amrita.edu/index.php> .

18. Трухин, А.В. Об использовании виртуальных лабораторий в образовании [Электронный ресурс]: Открытое и дистанционное образование .— 2002. — № 4 (8). — С. 27-30. Режим доступа: [https://ido.tsu.ru/files/pub2002/4\(8\)309Truhin_A._\(TUSUR\).pdf](https://ido.tsu.ru/files/pub2002/4(8)309Truhin_A._(TUSUR).pdf)

19. Remote-LAB GymKT. [Электронный ресурс]: Температура зависимость сопротивления металлов и полупроводников - Режим доступа: <http://remotelab.fyzika.net/index.php?lng=en> .

20. RELLE [Электронный ресурс]: - Ambiente de Aprendizagem com Experimentos - Режим доступа: <http://relle.ufsc.br>.

21. Шликов, В.В.. Мікропроцесорна техніка: Практикум [Электронный

ресурс]: навч. посіб. для студ.спеціальності 163 «Біомедична інженерія» та 152 «Метрологія та інформаційно-вимірювальна техніка»/; КПІ ім. Ігоря Сікорського. – Київ : КПІ ім. Ігоря Сікорського, 2018. – 144 с.

22. Arduino IDE [Електронний ресурс]:- Програмне забезпечення / Режим доступу: <https://www.arduino.cc/en/main/software>.

23. Ammyu Admin [Електронний ресурс]:Удаленный доступ и удаленное управление компьютером- Режим доступу: <http://www.ammyu.com/ru/> .

24. Методичні вказівки Дослідження фоторезистора, фотодіода та фототранзистора. [Текст]: Інститут радіотехніки, зв'язку та приладобудування
Режим доступу: <http://ogorodnyk.vk.vntu.edu.ua/.../2f5b3666d6c5d5b0979c107e..>
та <http://ogorodnyk.vk.vntu.edu.ua/.../94a51892d289afe867d322b4f....>

ДОДАТОК А

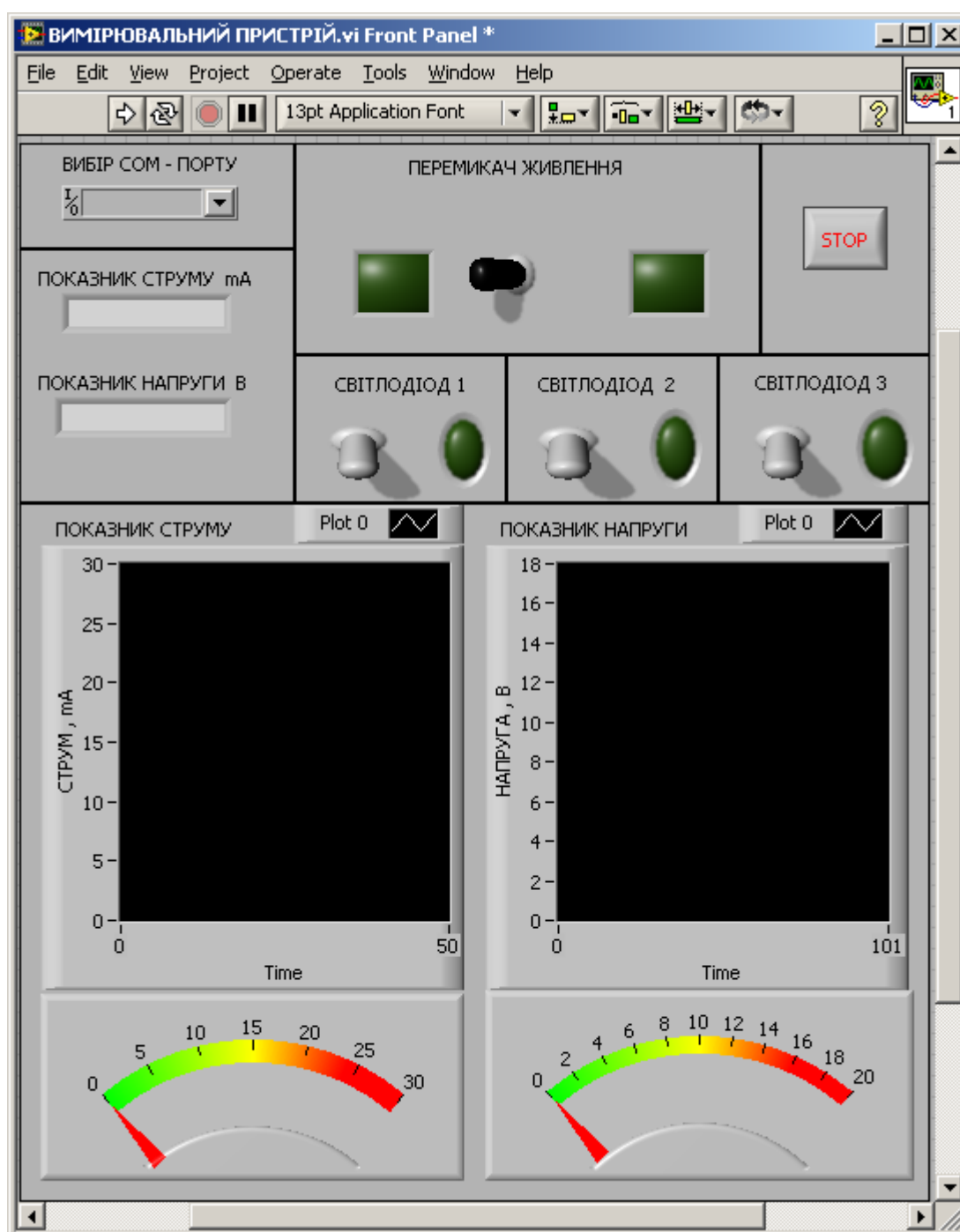


Рисунок А.1 - Фронтальна або лицьова панель пристрою

ДОДАТОК Б

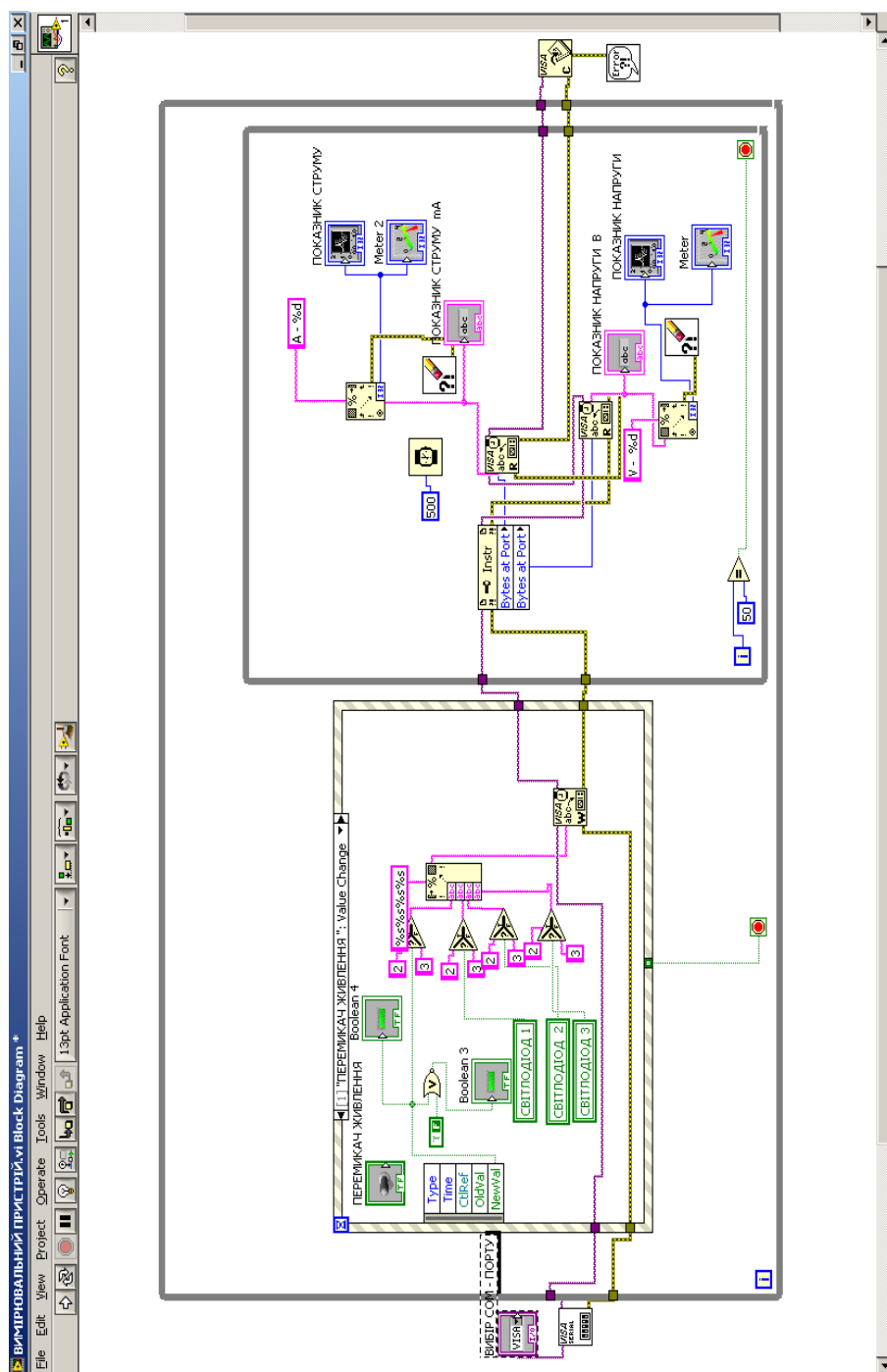


Рисунок Б.1 – Блок-схема віртуального пристрою

ДОДАТОК В

Скетч – код до програмування плати Arduino UNO

```
#define LED_PCB 12 // номер порту ардуіно для контролювання живленням
#define LED_PCS 11 // номер порту ардуіно для управління світлодіодом 1
#define LED_PCA 10 // номер порту ардуіно для управління світлодіодом 2
#define LED_PCL 9 // номер порту ардуіно для управління світлодіодом 3
int val ;// вихідна змінна для зчитування контролювання перемикачами
int analogInput = A0; // номер порту ардуіно для виводу значень вольтметра
int ampIn = A1; // номер порту ардуіно для виводу значень амперметра
float vin =0;// вхідна напруга
float ain =0;// вхідний тік

void setup() {
    pinMode(LED_PCB, OUTPUT);//зазначення виводу роботи порту
    pinMode(LED_PCS, OUTPUT);//зазначення виводу роботи порту
    pinMode(LED_PCA, OUTPUT);//зазначення виводу роботи порту
    pinMode(LED_PCL, OUTPUT);//зазначення виводу роботи порту
    Serial.begin(9600);// з'єднання з комп'ютером на швидкості 9600 бот
}

void loop() {
    if (Serial.available()){
val=Serial.parseInt();//зчитування з COM -порту значень, які подають перемикачі
if (val == 2333){
```

```

digitalWrite(LED_PCB, HIGH); //ввімкнення живлення на 10 В
digitalWrite(LED_PCS, LOW); //світлодіод вимкнений 1
digitalWrite(LED_PCA, LOW); //світлодіод вимкнений 2
digitalWrite(LED_PCL, LOW); //світлодіод вимкнений 3
}

else if (val == 2323){

    digitalWrite(LED_PCB, HIGH); //ввімкнення живлення на 10 В
    digitalWrite(LED_PCS, LOW); //світлодіод вимкнений 1
    digitalWrite(LED_PCA, HIGH); //ввімкнення світлодіода 2
    digitalWrite(LED_PCL, LOW); //світлодіод вимкнений 3
}

else if (val == 2332){

    digitalWrite(LED_PCB, HIGH); //ввімкнення живлення на 10 В
    digitalWrite(LED_PCS, LOW); //світлодіод вимкнений 1
    digitalWrite(LED_PCA, LOW ); //світлодіод вимкнений 2
    digitalWrite(LED_PCL, HIGH); //ввімкнення світлодіода 3
}

else if (val == 2233){

    digitalWrite(LED_PCB, HIGH); //ввімкнення живлення на 10 В
    digitalWrite(LED_PCS, HIGH); //ввімкнення світлодіода 1
    digitalWrite(LED_PCA, LOW); //світлодіод вимкнений 2
    digitalWrite(LED_PCL, LOW); //світлодіод вимкнений 3
}

```

```

}

else if (val == 3233){

    digitalWrite(LED_PCB, LOW); //включення живлення 15 В
    digitalWrite(LED_PCS, HIGH); //ввімкнення світлодіода 1
    digitalWrite(LED_PCA, LOW); //світлодіод вимкнений 2
    digitalWrite(LED_PCL, LOW); //світлодіод вимкнений 3
}

else if (val == 3323){

    digitalWrite(LED_PCB, LOW); //включення живлення 15 В
    digitalWrite(LED_PCS, LOW); //світлодіод вимкнений 1
    digitalWrite(LED_PCA, HIGH); //ввімкнення світлодіода 2
    digitalWrite(LED_PCL, LOW); //світлодіод вимкнений 3
}

else if (val == 3332){

    digitalWrite(LED_PCB, LOW); //включення живлення 15 В
    digitalWrite(LED_PCS, LOW); //світлодіод вимкнений 1
    digitalWrite(LED_PCA, LOW); //світлодіод вимкнений 2
    digitalWrite(LED_PCL, HIGH); //ввімкнення світлодіода 3
}

else if (val == 3333){

    digitalWrite(LED_PCB, LOW); //включення живлення 15 В
    digitalWrite(LED_PCS, LOW); //світлодіод вимкнений 1
    digitalWrite(LED_PCA, LOW); //світлодіод вимкнений 2

```

```
digitalWrite(LED_PCL, LOW); //світлодіод вимкнений 3
```

```
vin = float(analogRead(analogInput))/18.70; // зчитуємо, порівнюємо з опорним та  
ділимо на 18,70
```

```
ain = float(analogRead(ampIn))*1.45; // зчитуємо, порівнюємо з опорним та  
помножуємо на 1,45
```

```
Serial.print("V - "); //виводимо опис
```

```
Serial.println ( vin ); // виводимо значення вольтметра
```

```
Serial.print("A - "); //виводимо опис
```

```
Serial.println ( ain ); // виводимо значення амперметра
```

```
delay(500); //час через яке буде здійснене зчитування показань
```

```
}
```

```
}
```

ДОДАТОК Г

Методичні вказівки до виконання лабораторної роботи «Дослідження фотопровідності напівпровідників та фотоелектричних властивостей р-п переходу»

Мета роботи: вивчення основних фізичних закономірностей, визначаючих властивостей та параметри фото резисторів, фотодіодів та фототранзисторів, дослідження вольт-амперних і світлових характеристик цих приладів.

Теоретичні відомості:

Г.1 Фоторезистор

Г.1.1 Визначення понять квантовий вихід, чутливість та довгохвильова границя

Фоторезистор – це фотоелектричний напівпровідниковий приймач випромінювання, принцип дії якого базується на фотоелектричному ефекті. Це напівпровідникові фотоелементи здатні змінювати свій активний опір під дією падаючого на них світла.

Фоторезистор представляє собою пластину напівпровідника, на протилежних кінцях якого утворені омичні контакти (рисунок Г.1). При падінні світла на поверхню фоторезистора у останньому генеруються носії заряду в результаті збудження чи між зонних переходів (власне збудження), чи переходів з участю енергетичних рівнів у забороненій зоні (домішкове збудження), що призводить до збільшення провідності.

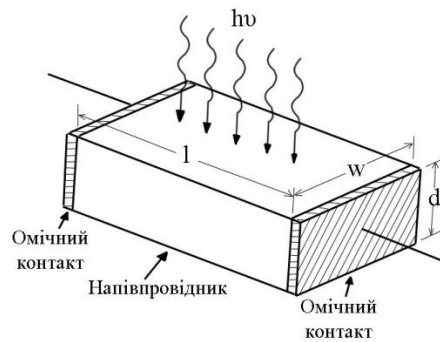


Рисунок Г.1 – Структура фоторезистора

У об'ємі напівпровідника відбувається створення фотоелектронами нових носіїв заряду шляхом ударної іонізації, тому квантовий вихід такого фотоприймача може бути значно менше одиниці

Квантовий вихід - число, яке вказує на те, скільки бажаних подій відбувається при поглинанні речовиною одного фотона. Це відношення кількості частинок, які зазнали перетворень під дією світла до кількості фотонів, що були поглинуті.

Основними характеристиками фоторезистора є інтегральна та спектральна чутливості. Інтегральна чутливість фоторезисторів визначається як відношення різниці струмів при освітленні та темнового струму до світлового потоку, що падає на резистор при номінальному значенні напруги:

$$K_{\phi} = \frac{I_{cs} - I_m}{\Phi}.$$

Спектральна чутливість фоторезистора визначається величиною фотоструму при освітленні його одиницею світлового потоку визначеної довжини хвилі.

Зважаючи на те, що залежність між струмом та напругою лінійна, вводять параметр питомої чутливості фоторезистора. Вона рівна відношенню фотоструму до добутку світлового потоку та прикладеної до фоторезистора напруги:

$$K_0 = \frac{I_\phi}{\Phi U}.$$

Довгохвильова границя – мінімальна частота або максимальна довжина хвилі світла, при якій ще можливий зовнішній фотоефект, тобто початкова кінетична енергія фотоелектронів більше нуля.

Г.1.2 Види фоторезисторів

Сірчисто-кадмієві (CdS) фоторезистори мають чутливість, що досягає максимуму в червоній області видимого спектру (довжина хвилі $\lambda=640\text{нм}$), захоплюючи ближню інфрачервону область до $\lambda=800\text{нм}$. Це досить велика чутливість, але вона повністю програє фоторезисторам з сірчаного свинцю. Фоторезистори з сірчаного кадмію здатні керувати відносно великими струмами (порядку декількох мА), при цьому їх передатна люксамперна характеристика близька до лінійної. Високий темновий опір, що досягає десятків ГОм, забезпечує максимально можливий динамічний діапазон по освітленості і найменші нелінійні спотворення сигналу. Проте швидкодія таких фоторезисторів низька: постійна часу реакції на зміну освітленості досягає 140мс при +25 С.

Фоторезистори з селеніду кадмію мають максимально можливу чутливість, що в 5-10 разів перевищує чутливість сірчисто-кадмієвих приладів. Максимум чутливості лежить в червоній області видимого спектру або в ближній інфрачервоній області довгохвильова границя якого досягає 1100 нм. При відносно вузькому динамічному діапазоні та гіршої лінійності селенід кадмію забезпечує кращу швидкодію – постійна часу його не перевищує 20 мс.

Фоторезистори з сірчаного свинцю мають довгохвильову границю чутливості при 3 мкм. При цьому вони охоплюють як зону видимого світла, так і зону інфрачервоного випромінювання, маючи при цьому досить велику швидкодію. Проте, фоторезистори виготовлені з сірчаного свинцю мають деякі недоліки. По-перше, це швидка окислюваність його на повітрі, по-друге,

присутність максимуму спектральної характеристики лише у зоні інфрачервоного випромінювання і слабка його чутливість у зоні короткохвильового видимого випромінювання, по-третє, досить великі габарити самого фото резистора.

Г. 1.3 Використання фоторезисторів

Завдяки досить високій чутливості, простоти конструкції, малих розмірів напівпровідникові фоторезистори знаходять широке застосування у різних електротехнічних схемах та установках. В останні роки фоторезистори стали широко використовуватися замість вакуумних та газонаповнених фотоелементів. Фоторезистори мають більш високу потужність розсіювання у порівнянні з фотодіодами та фототранзисторами, що дозволяє створювати прості та надійні реле без прискорювачів.

Г.2 Фотодіод

Г.2.1 Принцип дії

Фотодіод – це фотогальванічний приймач випромінювання без внутрішнього підсилення, фоточутливий елемент якого містить структуру напівпровідникового діода.

На рисунку Г.2.а, показані енергетичні зони в областях р-типу та n-типу напівпровідника за відсутності їх контакту. Під час дотикання областей (рис.Г.2.б) відбувається взаємна дифузія електронів та дірок – електрони дифундують з n- області в р- область, дірки – навпаки. У результаті р- область заряджена негативно, n- область – позитивно, і в районі переходу виникає електричне поле різниці потенціалів, що утворює потенціальний бар'єр U_0 .

При попаданні на р-n- перехід фотонів, енергії яких достатньо для внутрішнього фотоефекта, відбувається поглинання світла з утворенням

електронно-дірковій парі. Під дією електричного поля $U_{\text{б}}$ електрон переміщується в n-область, а дірка - у р-область. На контактах, що підведені до n- та р-областей, виникає вентильне фото-е.р.с. ΔU .

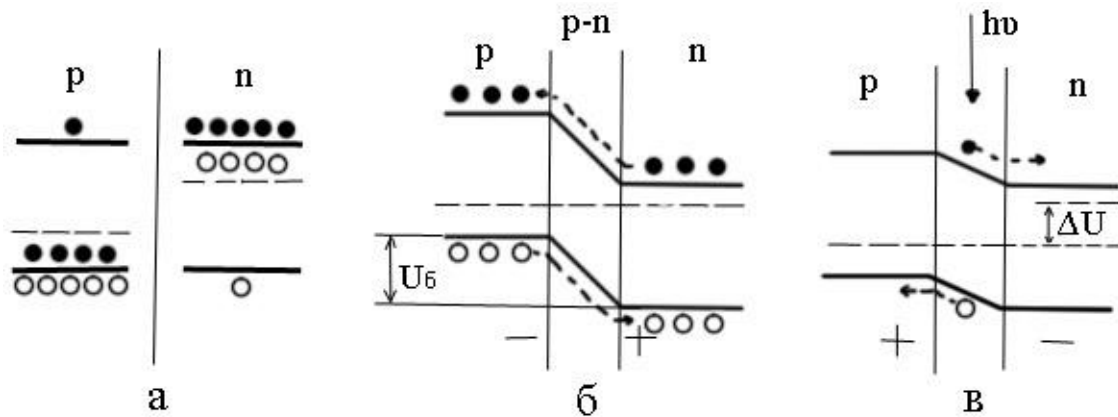


Рисунок 7.2 – Енергетичні зони р- та n- областей напівпровідника без контакту (а), у темряві (б) та при освітленні (в)

7.2.2 Режим роботи фотодіода

Відгук фотоприймача з р-n переходом можна реєструвати двома способами: у фотодіодному режимі, коли фотоприймач безпосередньо підключений до вимірювального пристрою, та в фотовентильному, коли фотоприймач безпосередньо підключений до вимірювального пристрою та вимірюється його власна фото-е.р.с, чи фотострум.

У фотодіодному режимі до р-n- переходу прикладена замикаюча напруга (додатній полюс до n- області, від'ємний – до р- області). У темряві через фотодіод тече слабкий темновий струм. При освітленні величина потенціального бар'єра $U_{\text{б}}$ зменшується на ΔU , струм збільшується. Величина струму в широких межах пропорційна інтенсивності падаючого світла і практично не залежить від напруги зміщення, що вигідно відрізняє фотодіоди від фоторезисторів. За своїми характеристиками фотодіоди подібні до вакуумних фотоелементів з зовнішнім фотоелементом.

При відсутності зовнішньої напруги фотодіоди працюють у фотовентильному режимі. Фотодіодний режим у порівнянні з фотовентильним характеризується кращою швидкістю, стабільністю, більшим динамічним діапазоном лінійного відгука, покращеною чутливістю у довгохвильовій області. Недолік фотодіодного режиму – темновий струм у відсутність засвітки, що призводить до додаткових шумів, які залежать від температури.

Г.2.3 Характеристики фотодіода

1) Вольтамперна характеристика $I_{\text{н}} = f(U_{\text{н}})$ при визначає залежність струму фотодіода від напруги на ньому при постійній величині світлового потоку. Характерною особливістю робочої області вольт амперної характеристик є практично повна незалежність струму фотодіода від прикладеної напруги. Такий режим настає при зворотних напругах на діоді порядку 1 В. Оскільки темновий струм малий, то відношення струму при освітленні до темнового струму велике, що досить важливо при індикації освітлення. Якщо зворотна напруга перевищить деяке допустиме значення, то у р-п- переході виникне ефект лавиноподібного розмноження носіїв заряду, який може вивести фотодіод зі строю (рис. Г.3).

2) Світлова характеристика зображує залежність струму фотодіода від величини світлового потоку при постійній напрузі на фотодіоді: $I_{\text{н}} = f(U)$ при $U = \text{const}$. У широкому діапазоні змін світлового потоку світлова характеристика фотодіода виявляється лінійною (рис. Г.4).

3) Спектральна характеристика показує залежність спектральної чутливості від довжини хвилі падаючого на фотодіод світла. Спектральні характеристики германієвих та кремнієвих фотодіодів показані на рисунку Г.5.

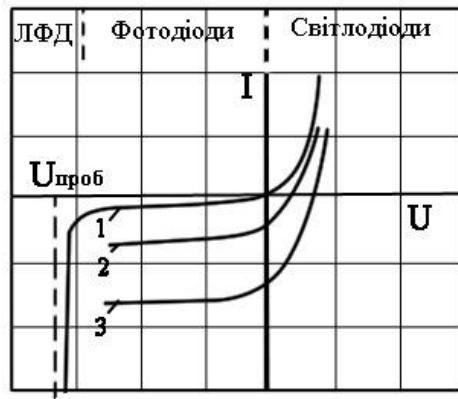


Рисунок Г.3 – Вольтамперна характеристика: 1 – за відсутності освітлення, 2 – при слабкому освітленні, 3 – при більш сильному освітленні.

Показані області роботи фотодіода, світлодіода та ЛФД. Вентильний режим відповідає $U=0$.

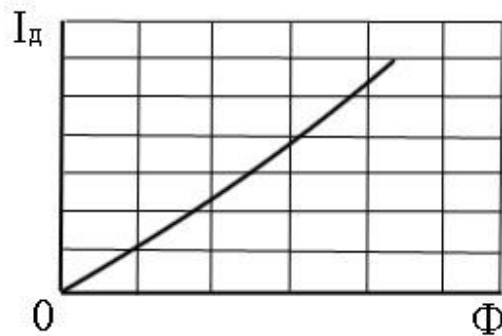


Рисунок Г.4 – Світлова характеристика фотодіода

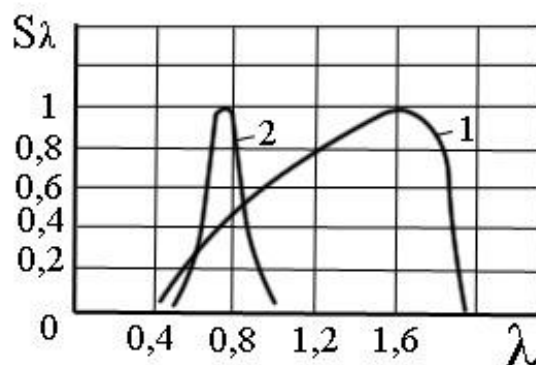


Рисунок Г.5 – Спектральна характеристика фотодіода (1 – германієвий фотодіод, 2 – кремнієвий фотодіод)

Г.2.4 Чутливості фотодіода

Інтегральна чутливість фотодіода – відношення фотоструму діода до інтенсивності падаючого немонахроматичного випромінювання заданого спектрального складу.

Питома чутливість – це відношення фотоструму до добутку величини падаючого на фотодіод світлового потоку на прикладену до нього напругу.

Г.2.5 Використання фотодіодів

Використовуються у фотометрії, для контролю джерел світла, вимірювань інтенсивності освітлення, прозорості середовища, реєстрації та рахунку ядерних частинок, автоматичного регулювання та контролю температури та інших параметрів, зміна яких супроводжується зміною оптичних властивостей матеріалу чи середовища. Широке застосування фотодіоди знаходять також у пристроях вводу та виводу сучасних ЕОМ, у оптоелектронних схемах.

Г.3 Фототранзистор

Г.3.1 Двополюсна схема включення фото транзистора

Фототранзистор являє собою фотогальванічний приймач випромінювання, фоточутливий елемент якого містить структуру транзистора, яка забезпечує внутрішнє підсилення.

Структура фототранзистора показана на рисунку Г.6. У корпусі приладу передбачене прозоре вікно, через яке світловий потік попадає на одну з областей транзистора. Напругу живлення підводять так, щоб колекторний перехід був закритий, а емітерний – відкритий. База може бути відключеною.

При освітленні бази у ній виникають електронно-діркові пари. Так, як і у фотодіоді, пари, що досягли у результаті дифузії колекторного переходу, розділяються полем переходу, дірки (у р-n-p) транзисторі рухаються у колектор, при цьому струм збільшується, електрони залишаються в базі, понижуючи її потенціал. При цьому на емітерному переході утворюється додаткова пряма напруга, що викликає додаткову інжекцію дірок з емітера у базу і відповідне збільшення струму колектора.

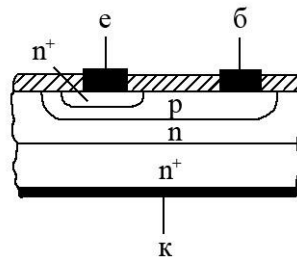


Рисунок Г.6 – Структура фото транзистора

Еквівалентна схема включення біполярного фототранзистора представлена на рисунку Г.7. Слід зазначити, що біполярний фототранзистор відрізняється від звичайного фототранзистора лише тим, що перехід база-колектор, що виконує роль фоточутливого елементу (на рисунку показаний як діод з паралельно включеною ємністю) має більшу площину.

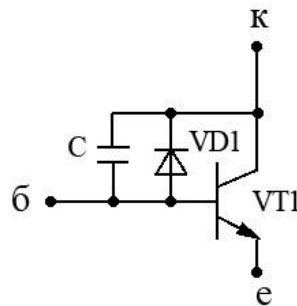


Рисунок Г.7 – Еквівалентна схема фото транзистора

Двополюсна схема підключення фото транзистора показана на рисунку Г.8. При такому включенні вивід бази фототранзистора залишається вільним, тобто струм бази $I_b=0$. При освітленні бази, на ній з'являються вільні електрони та дірки. Для бази фото транзистора типу р-п-р дірки є неосновними носіями заряду, тому вони втягуються полем колекторного переходу у колектор, збільшуючи струм у колі. Електрони, основні носії заряду, що залишилися на базі, створюють просторовий заряд, що понижує висоту потенціального бар'єра емітерного переходу. При цьому полегшується перехід дірок з емітера у базу, а потім у колектор, що призводить до ще більшого зростання колекторного струму, який проходить через опір навантаження. Таким чином, навіть при малому світловому потоці, падаючому на базу, струм колектора виявляється достатньо великим, що свідчить про високу чутливість фототранзистора.

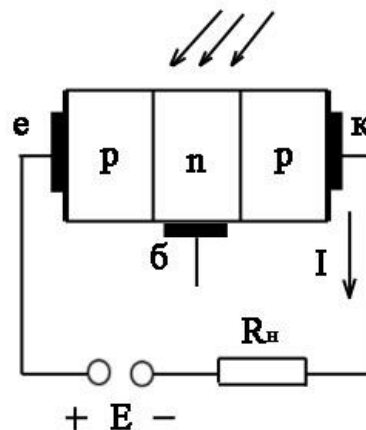


Рисунок Г.8 – Двополюсна схема включення фото транзистора

Г.3.2 Довгохвильова границя, квантова ефективність, чутливість

Довгохвильова границя - «Червона» межа фотоефекту - мінімальна частота або максимальна довжина хвилі світла, при якій ще можливий зовнішній фотоефект, тобто початкова кінетична енергія фотоелектронів більше нуля.

Квантова ефективність - це відношення числа носіїв, зібраних сонячним елементом, до числа фотонів даної енергії, падаючих на сонячний елемент.

Іншими словами, квантова ефективність - це ефективність перетворення оптичної потужності в електричну, виражена у %.

Чутливість - це відношення фотоструму до потоку випромінювання, що викликав цей струм. Чутливість фотоелемента визначає його якість.

Г.3.3 Використання фототранзисторів

Спершу фототранзистори використовувались лише у наведеній вище двополюсній схемі включення. Фототранзистор, оснащений трьома виводами, представляє додаткові можливості його використання, засновані на тому, що, крім світлового сигналу, на його вхід можна подати сигнал електричний.

Фототранзистори використовуються у якості чутливих елементів у різноманітних автоматичних пристроях, фототелеграфії, кіноапаратурі, у приладах вводу-виводу інформації у обчислювальній техніці, для реєстрації ультрафіолетового та інфрачервоного випромінювання та у оптоелектроніці.