

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

МЕТОДИЧНІ ВКАЗІВКИ

до виконання лабораторних робіт з дисципліни
«Обчислювальна техніка та програмування» частина 1
для студентів спеціальності 141 – Електроенергетика,
електротехніка та електромеханіка всіх форм навчання

Методичні вказівки до виконання лабораторних робіт з дисципліни «Обчислювальна техніка та програмування» частина 1 для студентів спеціальності 141 – Електроенергетика, електротехніка та електромеханіка всіх форм навчання. /Укл.: Л.С. Скрупська, С.І. Шило – Запоріжжя: НУЗП, 2025.– 31с.

Укладачі: Л.С. Скрупська, ст. викл.
С.І. Шило, ст. викл.

Рецензент: В.В.Василевський, доцент, к. т. н.

Відповідальний за випуск: Л.С. Скрупська, ст. викл.

Затверджено
на засіданні кафедри
«Електричні та
електронні апарати»
Протокол № 8
від « 21 » 02 2025 р.

Затверджено НМК ЕТФ
Протокол № 7
від « 24 » 02 2025 р.

ЗМІСТ

Вступ.....	4
Лабораторна робота № 1. Архітектура та принципи функціонування обчислювальних систем	5
Лабораторна робота № 2. Основи логіки та двійкових обчислень.....	8
Лабораторна робота № 3. Архітектура процесорів	10
Лабораторна робота № 4. Оперативна пам'ять (ОЗП) - основний компонент комп'ютерної системи	13
Лабораторна робота № 5. Системи команд	16
Лабораторна робота № 6 Організація вводу-виводу	19
Лабораторна робота № 7.1. Основи роботи в текстовому редакторі Microsoft Word	22
Лабораторна робота № 7.2. Основи роботи в текстовому редакторі Microsoft Word	26
Список використаних джерел посилання	30

ВСТУП

Лабораторні роботи з дисципліни "Обчислювальна техніка та програмування" є важливою складовою навчального процесу для студентів спеціальності 141 – "Електроенергетика, електротехніка та електромеханіка". Вони спрямовані на засвоєння основних принципів функціонування обчислювальних систем, їх компонентів та взаємодії з іншими пристроями.

Ці лабораторні роботи охоплюють кілька ключових тем, що включають: архітектуру обчислювальних систем, основи логіки та двійкових обчислень, структуру процесорів, оперативну пам'ять, організацію вводу-виводу даних та роботу в текстовому редакторі Microsoft Word. Всі ці аспекти є основою для розуміння того, як комп'ютерні системи працюють на рівні апаратного та програмного забезпечення, і як вони використовуються для вирішення задач у сфері електроенергетики та електротехніки.

Крім того, лабораторна робота передбачає розвиток практичних навичок роботи з основними компонентами комп'ютерних систем, зокрема, з процесорами, пам'яттю, системами команд і засобами введення/виведення даних, а також освоєння роботи з текстовими та обчислювальними програмами. Знання цих основ є критично важливим для майбутніх інженерів, адже вони дозволяють ефективно використовувати комп'ютери для розв'язування спеціалізованих завдань у галузі електроенергетики та електротехніки.

Метою виконання лабораторних робіт є формування навичок практичного застосування теоретичних знань, здобутих на лекціях, а також розвитку вмінь самостійно вирішувати інженерні задачі, використовуючи сучасні обчислювальні засоби та програмне забезпечення.

Завдання лабораторних робіт допоможуть студентам оволодіти базовими методами і підходами до обробки даних у комп'ютерних системах, а також освоїти роботу з основними програмами для створення документів, обробки даних і проведення розрахунків.

Ці методичні вказівки містять покрокові інструкції для виконання лабораторних робіт, а також допомогу для виконання розрахунків і завдань з програмування та роботи з інформацією.

ЛАБОРАТОРНА РОБОТА № 1

Архітектура та принципи функціонування обчислювальних систем

Мета роботи - ознайомити студентів з основними компонентами обчислювальної системи, принципами її роботи, структурою процесора, організацією пам'яті та шинами даних.

1. Обладнання та програмне забезпечення

- Персональний комп'ютер або комп'ютерна система з доступом до апаратних компонентів (процесор, пам'ять, шини).
- Підключення до комп'ютера для виконання лабораторних завдань програмного забезпечення, яке може включати емулятори апаратних компонентів (якщо необхідно) або конкретні інтерфейси для моніторингу шини та пам'яті.

2. Завдання лабораторної роботи

У ході лабораторної роботи студент повинен:

- Вивчити основні компоненти обчислювальної системи та їх взаємодію.
- Ознайомитись зі структурою процесора, принципами його роботи та основними його елементами.
- Зрозуміти принципи роботи пам'яті, типи пам'яті та їх роль в роботі комп'ютерної системи.
- Вивчити функціонування шин даних, адресних шин та шин керування, а також їх взаємодію в процесі обробки інформації.

3. Теоретичні відомості

3.1 Основні компоненти комп'ютера

Комп'ютер складається з кількох основних частин:

1. Процесор (CPU) — центральний елемент, який виконує програму, обробляє дані і керує роботою інших компонентів системи.
2. Оперативна пам'ять (RAM) — використовується для тимчасового зберігання даних, що активно використовуються під час роботи програми.

3. Постійна пам'ять (ROM) — використовується для зберігання важливих даних, таких як прошивка або базова система введення/виведення (BIOS).

4. Пристрої вводу/виводу (I/O) — включають периферійні пристрої, такі як клавіатура, мишка, принтер, монітор та інші.

5. Шини даних (Bus) — канали для передачі даних між компонентами комп'ютера.

3.2 Структура процесора

Процесор складається з таких основних частин:

- Арифметико-логічний пристрій (ALU) — виконує арифметичні і логічні операції.

- Блок керування (Control Unit) — керує виконанням програм, даючи команди іншим частинам комп'ютера.

- Регістрами — тимчасово зберігають дані, що використовуються в поточній операції.

- Кеш-пам'ять — використовується для зберігання часто використовуваних даних, щоб скоротити час доступу до них.

3.3 Пам'ять комп'ютера

Пам'ять комп'ютера можна поділити на кілька типів:

- Оперативна пам'ять (RAM) — швидка, але тимчасова пам'ять, яка зберігає дані тільки під час роботи комп'ютера.

- Постійна пам'ять (ROM) — неволатильна пам'ять, яка зберігає дані після вимкнення живлення.

- Кеш-пам'ять — швидка пам'ять, що зберігає часто використовувані дані між процесором і основною пам'яттю.

3.4 Шини даних

Шини — це системи для передачі даних між компонентами комп'ютера. Існують кілька видів шин:

- Шина даних (Data Bus) — передає самі дані.

- Адресна шина (Address Bus) — передає адреси, за якими зберігаються дані.

3.5. Шина керування (Control Bus) — передає сигнали, які управляють операціями в системі.

4. Хід виконання роботи

4.1 Ознайомлення зі структурою процесора

1. Використовуючи програмне забезпечення для моделювання комп'ютерної архітектури або реальний комп'ютер, здійснити наступні кроки:

- Вивчити організацію процесора (визначити наявність ALU, CU, регістрів та кеш-пам'яті).

- Перевірити, як процесор виконує операції (запустити прості програми та спостерігати за виконанням команд).

2. Розглянути принципи роботи процесора: як він отримує інструкції, розшифровує їх і виконує.

4.2 Вивчення роботи пам'яті

Провести тестування різних типів пам'яті:

- Використати програму для моніторингу використання оперативної пам'яті (RAM).

- Дослідити роботу кеш-пам'яті і спостерігати, як вона прискорює доступ до даних.

- Ознайомитись з різницями між RAM і ROM, а також як дані зберігаються в кожному типі пам'яті.

4.3 Аналіз роботи шин

Використовуючи спеціалізоване програмне забезпечення або апаратні інтерфейси, здійснити моніторинг роботи шин:

- Перевірити, як передаються дані через шину даних.

- Проаналізувати використання адресної шини для доступу до пам'яті.

- Дослідити шину керування та її роль в синхронізації компонентів комп'ютерної системи.

4.4 Практичне застосування компонентів комп'ютера

На базі отриманих знань, студентам пропонується створити просту програму для виконання арифметичних операцій (наприклад, додавання чи множення чисел), що продемонструє роботу процесора, пам'яті та шин даних.

Контрольні питання

1. Яка роль процесора в обчислювальній системі?

2. Як побудована організація пам'яті комп'ютера і в чому різниця між різними типами пам'яті?

3. Які типи шин даних використовуються в обчислювальній техніці та як вони взаємодіють?

4. Як кеш-пам'ять допомагає прискорити роботу комп'ютера?

5. Які основні функції блоку керування процесором?
6. Які процеси відбуваються в процесорі під час виконання програми?

ЛАБОРАТОРНА РОБОТА № 2

Основи логіки та двійкових обчислень

Мета роботи:

Метою цієї лабораторної роботи є ознайомлення студентів з основами логіки та двійкових обчислень, розумінням їхнього застосування в обчислювальній техніці, а також розвиток навичок роботи з логічними операціями та двійковими числами. Студенти також отримають практичні навички в побудові та аналізі простих логічних схем, що лежать в основі цифрових пристроїв.

1. Обладнання та програмне забезпечення

1. Комп'ютери з доступом до інтерфейсу для моделювання логічних схем (наприклад, використання програми для візуального моделювання цифрових схем, такої як Logisim або Digital Works).
2. Калькулятори або програмне забезпечення для виконання математичних операцій з двійковими числами.
3. Теоретичні матеріали та навчальні ресурси з логіки та двійкових обчислень (підручники та наукові статті).
4. Різноманітні ресурси в Інтернеті для самостійного вивчення теоретичних основ (вебсайти, відеолекції тощо).

2. Завдання лабораторної роботи

1. Ознайомитись із основними логічними операціями (AND, OR, NOT, XOR) та побудувати їх таблиці істинності.
2. Перевести десяткові числа в двійкову систему числення та виконати прості арифметичні операції з двійковими числами (додавання, віднімання).
3. Побудувати прості логічні схеми з використанням елементів AND, OR, NOT та XOR, та провести їх аналіз.
4. Виконати розрахунки на базі двійкових чисел для задачі, яка стосується конкретних інженерних застосувань, таких як контролери, інтерфейси і цифрові системи.

5. Пояснити застосування двійкових чисел та логічних операцій в реальних пристроях, таких як мікропроцесори та цифрові схеми.

3. Теоретичні відомості

3.1. Логічні операції:

- AND (І): Логічна операція «І» повертає істинне значення тільки тоді, коли обидва операнди істинні.

- OR (АБО): Логічна операція «АБО» повертає істинне значення, якщо хоча б один з операндів істинний.

- NOT (НЕ): Логічна операція «НЕ» змінює значення на протилежне.

- XOR (І виключне): Операція «І виключне» повертає істинне значення, коли операнди різні.

3.2 Двійкові числа:

- Переведення з десяткової в двійкову систему: Процес переведення числа з

десяткової системи числення в двійкову здійснюється за допомогою ділення числа на 2 з записом залишків.

- Арифметичні операції з двійковими числами: додавання, віднімання, множення та ділення.

3.3. Логічні схеми:

- Логічні елементи (AND, OR, NOT) використовуються для побудови складних логічних схем, які є основою роботи цифрових пристроїв.

- Логічні схеми дозволяють реалізовувати алгоритми обробки сигналів, прийняття рішень, а також працюють в схемах пам'яті та обчислювальних блоках.

4. Хід виконання роботи

Завдання 1: Створення таблиць істинності для логічних операцій

- Для кожної з операцій (AND, OR, NOT, XOR) створіть таблиці істинності, враховуючи можливі значення операндів 0 та 1.

Завдання 2: Переведення чисел з десяткової системи в двійкову

- Виконайте переведення декількох десяткових чисел (наприклад, 15, 22, 38) у двійкову систему числення.

Завдання 3: Арифметичні операції з двійковими числами

- Виконайте операції додавання та віднімання з двійковими числами.

Наприклад:

$$1011_2 + 1101_2$$

$$11001_2 - 1001_2$$

Завдання 4: Побудова логічних схем

- Створіть схему, що реалізує логічну функцію, яка обробляє комбінацію декількох входів за допомогою елементів AND, OR, NOT.

Завдання 5: Аналіз логічних схем

- Проведіть аналіз побудованих схем. Поясніть, яку задачу вони вирішують, і як це може бути використано в реальних електронних пристроях.

Контрольні питання

1. Що таке логічні операції? Наведіть приклади їх застосування.
2. Які основні відмінності між системами числення (десятьковою та двійковою)?
3. Як виконується переведення числа з десятикової в двійкову систему?
4. Як проводиться арифметична операція додавання двійкових чисел? Покажіть приклад.
5. Які функції виконують елементи AND, OR, NOT в цифрових схемах?
6. Як працює логічний елемент XOR?

ЛАБОРАТОРНА РОБОТА № 3**Архітектура процесорів****Мета роботи:**

- Ознайомлення з основними принципами архітектури сучасних процесорів.

- Вивчення основних характеристик і принципів роботи процесорів.

- Дослідження впливу архітектури процесора на продуктивність електронних систем.

1. Обладнання та програмне забезпечення

Для виконання лабораторної роботи необхідне наступне обладнання та програмне забезпечення:

- Комп'ютер з доступом до Інтернету для доступу до навчальних матеріалів.

- Програмне забезпечення для моделювання та аналізу процесорів, наприклад, спеціалізовані пакети для вивчення архітектури комп'ютерних систем (для візуалізації структури процесора і його роботи).

- Інтерактивні моделі та симулятори для демонстрації архітектури процесорів (наприклад, онлайн-симулятори, які дають можливість переглядати схеми внутрішніх підсистем процесорів).

2. Завдання лабораторної роботи

2.1. Ознайомитись з архітектурою процесора, визначити основні компоненти процесора (арифметико-логічний пристрій, реєстри, кеш-пам'ять, шина даних та інші).

2.2. Вивчити класифікацію процесорів: одноядерні, багатоядерні, RISC і CISC.

2.3. Описати принципи роботи процесора, зокрема цикли вибірки, декодування та виконання команд.

2.4. Виконати практичні завдання для дослідження характеристик та функцій процесора, таких як його частота такту, кількість ядер, кеш-пам'ять.

2.5. Розглянути взаємодію процесора з іншими компонентами системи (пам'ять, периферійні пристрої).

3. Теоретичні відомості

3.1. Архітектура процесора

Процесор є ключовим елементом обчислювальної системи. Основними компонентами процесора є:

- Арифметико-логічний пристрій (АЛП): виконання арифметичних і логічних операцій.

- Регістрами: невеликі швидкодіючі пам'яті, що використовуються для тимчасового зберігання даних.
- Шина даних: забезпечує обмін даними між процесором, пам'яттю і периферійними пристроями.
- Кеш-пам'ять: для прискореного доступу до часто використовуваних даних.

3.2. Класифікація процесорів

Процесори класифікуються за різними ознаками:

- RISC (Reduced Instruction Set Computing) — процесори з обмеженим набором простих команд.
- CISC (Complex Instruction Set Computing) — процесори з великим набором команд, здатних виконувати складні операції.
- Одноядерні і багатоядерні процесори — багатоядерні процесори можуть виконувати кілька команд паралельно.

3.3. Принцип роботи процесора

Процесор працює за принципом циклу: вибірка інструкції → декодування → виконання → запис результату. Це дозволяє йому по черзі виконувати операції на оброблюваних даних.

4. Хід виконання роботи

4.1. Запуск CPU-Z та аналіз характеристик процесора

- Відкрити програму CPU-Z та переглянути вкладки "CPU", "Caches", "Mainboard".
- Звернути увагу на модель процесора, архітектуру, технологічний процес.
- Визначити кількість ядер і потоків, базову та динамічну тактову частоту.
- Ознайомитися з інформацією про кеш-пам'ять L1, L2 та L3.

4.2. Дослідження продуктивності процесора

- Запустити програму AIDA64 та відкрити вкладку "CPU Benchmark".
- Виконати тести "CPU Queen", "FPU Julia", "Memory Read/Write".
- Порівняти отримані результати з еталонними показниками для аналогічних процесорів.
- Проаналізувати продуктивність процесора у розрахункових задачах та при роботі з пам'яттю.

4.3. Аналіз температурного режиму

- У CPU-Z переглянути вкладку "Sensors" або скористатися AIDA64 (вкладка "Computer → Sensor").

- Виміряти температуру процесора у стані спокою та під навантаженням.

- Виконати короткочасний стрес-тест та зафіксувати зміну температур.

4.4. Зробити висновки щодо архітектури процесора

- Порівняти отримані характеристики з даними офіційної документації.

- Визначити, чи відповідає продуктивність заявленим параметрам.

- Оцінити ефективність системи охолодження та можливості розгону процесора.

4.5. Документація результатів.

Оформіть звіт, в якому:

- Описано основні характеристики процесора.

- Приведено результати досліджень продуктивності процесорів.

- Визначено, які типи процесорів найбільш ефективні для певних завдань.

Контрольні питання

1. Які основні елементи архітектури процесора?
2. Чим відрізняється багатоядерний процесор від одноядерного?
3. Як впливає кеш-пам'ять на продуктивність процесора?
4. Які функції виконує контролер пам'яті в процесорі?
5. Які програми використовуються для аналізу архітектури процесора?

ЛАБОРАТОРНА РОБОТА № 4

Оперативна пам'ять (ОЗП) - основний компонент комп'ютерної системи

Мета роботи:

- Ознайомлення з призначенням, функціями та видами оперативної пам'яті.
- Вивчення основних характеристик ОЗП та принципів її роботи.
- Аналіз взаємодії оперативної пам'яті з іншими компонентами комп'ютерної системи.
- Навчитися визначати оптимальну конфігурацію ОЗП для різних завдань.

1. Обладнання та програмне забезпечення

1. Персональний комп'ютер (ПК) або ноутбук.
2. Операційна система Windows або Linux.
3. Програми для діагностики пам'яті (CPU-Z, AIDA64, MemTest86).

2. Завдання лабораторної роботи

- 2.1. Визначити обсяг та характеристики встановленої оперативної пам'яті.
- 2.2. Проаналізувати тип і швидкодію ОЗП за допомогою програмного забезпечення.
- 2.3. Дослідити принципи роботи оперативної пам'яті та її взаємодію з процесором та накопичувачами.
- 2.4. Запропонувати оптимальну конфігурацію ОЗП для різних типів задач.

3. Теоретичні відомості

3.1. Призначення та функції ОЗП

Оперативна пам'ять (ОЗП) – це основний вид пам'яті комп'ютера, який використовується для тимчасового зберігання даних і програм під час їх виконання. Вона забезпечує швидкий доступ до інформації, необхідної процесору для обробки.

3.2. Види оперативної пам'яті

- DRAM (Dynamic RAM) – найбільш поширений тип пам'яті, що використовується у сучасних ПК.
- SRAM (Static RAM) – швидша, але дорожча пам'ять, що використовується в кеш-пам'яті процесорів.
- DDR (Double Data Rate) RAM – сучасний стандарт оперативної пам'яті, що включає покоління DDR3, DDR4, DDR5.

3.3. Основні характеристики ОЗП

- Об'єм пам'яті – визначає, скільки даних може бути одночасно завантажено в пам'ять.
- Частота (MHz) – швидкість передачі даних між пам'яттю та процесором.
- Латентність (CL – CAS Latency) – час затримки між запитом і отриманням даних.
- Пропускна здатність – обсяг даних, що може бути передано за одиницю часу.

3.4. Принципи роботи ОЗП

ОЗП працює як буфер між процесором і накопичувачем (SSD/HDD), дозволяючи зберігати та швидко обробляти поточні завдання.

3.5. Взаємодія ОЗП з іншими компонентами системи

- Процесор зчитує та записує дані в ОЗП для виконання програм.
- Жорсткі диски та SSD обмінюються даними з ОЗП через контролери пам'яті.
- Відеокарта може використовувати частину оперативної пам'яті (вбудовані графічні адаптери).

3.6. Вибір оптимальної конфігурації ОЗП для різних задач

- Для офісної роботи: 8 ГБ DDR4/DDR5.
- Для ігор та мультимедіа: 16-32 ГБ DDR4/DDR5 з високою частотою.
- Для професійних завдань (віртуалізація, 3D-моделювання): 32+ ГБ DDR5 з мінімальною затримкою.

4. Хід виконання роботи

Крок 1: Визначення характеристик встановленої ОЗП

- Відкрийте програму CPU-Z або AIDA64.
- Перейдіть у вкладку "Memory" (ОЗП) і запишіть основні параметри: тип, обсяг, частоту, таймінги.
- Виконайте аналіз і порівняйте отримані характеристики з рекомендованими для вашої конфігурації.

Крок 2: Аналіз продуктивності ОЗП

- Запустіть програму AIDA64 або Windows Task Manager для моніторингу використання пам'яті.

- Виконайте тест пропускну́ї здатності пам'яті (Memory Benchmark).

- Порівняйте результати з середніми значеннями для вашого типу пам'яті.

Крок 3: Тестування на надійність

- Запустіть MemTest86 або вбудований у Windows "Діагностичний інструмент пам'яті".

- Дочекайтеся завершення тесту та перегляньте результати на наявність помилок.

Крок 4: Вибір оптимальної конфігурації ОЗП

- Використовуючи отримані результати, оцініть, чи відповідає об'єм і швидкість пам'яті вашим потребам.

- Якщо необхідно, оберіть додаткові модулі ОЗП для покращення продуктивності.

- Перевірте сумісність нових модулів із материнською платою за специфікацією виробника.

Контрольні питання

1. Яке основне призначення оперативної пам'яті?
2. Які основні види оперативної пам'яті існують?
3. Що таке пропускну́а здатність ОЗП і як вона впливає на продуктивність?
4. Як можна перевірити працездатність модулів ОЗП?
5. Які фактори слід враховувати при виборі оперативної пам'яті для різних задач?

ЛАБОРАТОРНА РОБОТА № 5

Системи команд

Мета роботи:

- Ознайомлення зі структурою та форматами команд у комп'ютерних системах.
- Вивчення етапів виконання команд процесором.
- Аналіз класифікації систем команд.
- Дослідження організації програмного управління виконанням програм.

- Розгляд використання систем команд у електронних апаратах.

1. Обладнання та програмне забезпечення

1. Персональний комп'ютер або ноутбук.
2. Операційна система Windows або Linux.
3. Програми-емулятори для аналізу виконання команд (EMU8086, QEMU, Logisim).
4. Інструменти моніторингу роботи процесора (CPU-Z, Intel VTune, AIDA64).

2. Завдання лабораторної роботи

- 2.1. Вивчити структуру і формати команд у мікропроцесорних системах.
- 2.2. Дослідити етапи виконання команди в процесорі.
- 2.3. Ознайомитися з класифікацією систем команд.
- 2.4. Виконати аналіз програмного управління виконанням програм.
- 2.5. Розглянути практичне використання команд у електронних апаратах.

3. Теоретичні відомості

3.1. Структура і формати команд

Команда – це машинний код, що складається з:

- Коду операції (Opcode) – визначає дію, яку потрібно виконати.

- Операндів – вказують на дані або адреси.
- Режиму адресації – спосіб доступу до даних.

Формати команд:

- Однобайтові – містять тільки код операції.
- Двобайтові – код операції + операнд.
- Багатобайтові – складніші інструкції (адреси, дані).

3.2. Етапи виконання команд

- Вибірка команди – процесор отримує команду з пам'яті.
- Декодування команди – процесор аналізує її структуру.
- Виконання команди – обчислення або передача даних.
- Запис результату – збереження отриманого значення.

3.3. Класифікація систем команд

- CISC (Complex Instruction Set Computing) – складні команди, велика кількість операндів.
- RISC (Reduced Instruction Set Computing) – мінімалістичні команди, висока швидкодія.
- DSP (Digital Signal Processing) – спеціалізовані команди для обробки сигналів.

3.4. Організація програмного управління виконанням програм

- Послідовне виконання команд.
- Використання умовних і безумовних переходів.
- Робота з циклами та підпрограмами.

3.5. Використання систем команд у електронних апаратах

Системи команд застосовуються в:

- Мікроконтролерах (Arduino, STM32).
- Програмованих логічних контролерах (PLC).
- Вбудованих системах у побутовій техніці та промислового обладнанні.

4. Хід виконання роботи

Крок 1: Визначення структури команд у сучасних процесорах

- Встановіть та відкрийте програму CPU-Z.
- Перейдіть у вкладку "Instructions" та перегляньте підтримувані системи команд (SSE, AVX).
- Запишіть отримані дані.

Крок 2: Аналіз виконання команди у емуляторі

- Встановіть емулятор EMU8086 або QEMU.
- Введіть просту програму на мові асемблера, наприклад:

```
MOV AX, 05H
ADD AX, 03H
```

- Запустіть виконання та простежте, як змінюється стан регістрів.
- Зробіть висновок про виконання команди.

Крок 3: Дослідження програмного управління виконанням

- Введіть у емулятор наступний код для циклічного додавання чисел:

```
MOV CX, 05H
LOOP_START:
ADD AX, 01H
LOOP LOOP_START
```

- Проаналізуйте, як працює команда LOOP.
- Запишіть висновки про управління виконанням.

Крок 4: Визначення оптимальної системи команд для задачі

- Виберіть між CISC та RISC залежно від задачі:
- Для швидких розрахунків – RISC.
- Для універсальності – CISC.
- Порівняйте виконання команд у різних процесорах за допомогою тесту AIDA64.
- Запишіть висновки щодо ефективності використання систем команд.

Контрольні питання

1. Які основні компоненти команди?
2. Які етапи проходить команда під час виконання?
3. Які основні відмінності між CISC і RISC?
4. Як відбувається управління виконанням програм у мікропроцесорі?
5. Як системи команд використовуються в електронних апаратах?

ЛАБОРАТОРНА РОБОТА № 6

Організація вводу-виводу

Мета роботи: Засвоїти основні принципи організації вводу-виводу в комп'ютерних системах, ознайомитися з різними режимами роботи процесора з зовнішніми пристроями, а також вивчити методи програмного обміну даними, обробки переривань та організації прямого доступу до пам'яті (DMA) за допомогою програмного інтерфейсу Electronics Workbench.

1. Обладнання та програмне забезпечення:

- Комп'ютер з встановленою операційною системою Windows або Linux.
- Програмне забезпечення Electronics Workbench для моделювання та тестування електронних схем.

- Схеми для моделювання організації вводу-виводу та управління перериваннями.
- Зовнішні пристрої для симуляції вводу-виводу (клавіатура, мишка, світлодіоди, кнопки тощо).
- Додаткове програмне забезпечення для симуляції: текстові редактори, таблиці для запису результатів.

2. Завдання лабораторної роботи:

- 2.1. Ознайомитись з принципами роботи зовнішніх пристроїв в комп'ютерних системах.
- 2.2. Моделювати різні режими роботи процесора з зовнішніми пристроями у Electronics Workbench.
- 2.3. Реалізувати програмний обмін даними між процесором і зовнішнім пристроєм.
- 2.4. Ознайомитись з обробкою переривань в організації вводу-виводу.
- 2.5. Моделювати та проаналізувати механізм прямого доступу до пам'яті (DMA).

3. Теоретичні відомості:

- 3.1. Режими роботи процесора з зовнішніми пристроями:
 - Програмний режим (постійна перевірка стану пристрою).
 - Режим обробки переривань (коли пристрій ініціює обробку даних).
 - Режим прямого доступу до пам'яті (DMA), коли процесор не втручається в обробку даних.
- 3.2. Програмний обмін даними в організації вводу-виводу:
 - Під програмним обміном розуміється передача даних між процесором та зовнішнім пристроєм під контролем програмного коду.
 - Для цього використовуються операції читання та запису, в яких процесор періодично перевіряє стан зовнішнього пристрою.
- 3.3 Обробка переривань:
 - Переривання — це механізм, що дозволяє зовнішнім пристроям повідомляти процесору про необхідність обробки певної події (наприклад, натискання кнопки).
 - Процесор реагує на переривання, призупиняючи виконання поточної програми і переходячи до обробки події.

3.4. Організація прямого доступу до пам'яті (DMA):

- В режимі DMA зовнішній пристрій може безпосередньо передавати дані до або з пам'яті без участі процесора.

- Це дозволяє знизити навантаження на процесор і прискорити обробку даних.

4. Хід виконання роботи:

Крок 1. Ознайомлення з інтерфейсом Electronics Workbench:

- Запустіть програму Electronics Workbench.

- Ознайомтесь з базовими функціями симулятора: створення схем, підключення елементів, налаштування параметрів компонентів.

Крок 2. Моделювання роботи зовнішнього пристрою:

- Створіть схему, що включає процесор і зовнішній пристрій (наприклад, світлодіод або клавіатуру).

- Налаштуйте режим програмного обміну даними, підключивши порти вводу/виводу до відповідних елементів.

Крок 3. Реалізація обробки переривань:

- Додайте блок, який імітує переривання від зовнішнього пристрою.

- Підключіть його до процесора та налаштуйте механізм обробки переривань.

- Проаналізуйте, як процесор припиняє виконання поточної програми та переходить до обробки переривання.

Крок 4. Моделювання DMA:

- Створіть схему з використанням DMA для передачі даних між пам'яттю і зовнішнім пристроєм.

- Перевірте, як процесор взаємодіє з DMA контролером і чи зменшується навантаження на процесор.

Крок 5. Аналіз результатів:

- Після виконання всіх етапів роботи, проаналізуйте симуляційні результати.

- Зробіть висновки щодо ефективності різних методів організації вводу-виводу.

Контрольні питання:

1. Які основні режими роботи процесора з зовнішніми пристроями?

2. У чому полягає принцип обробки переривань і як це реалізується на апаратному рівні?

3. Що таке програмний обмін даними і які його основні переваги та недоліки?

4. Які переваги використання DMA в порівнянні з програмним обміном даними?

5. Як організувати моделювання вводу-виводу в Electronics Workbench?

ЛАБОРАТОРНА РОБОТА № 7.1

Основи роботи в текстовому редакторі Microsoft Word

Мета роботи: Метою лабораторної роботи є освоєння основних функцій текстового редактора Microsoft Word, навчання створення, редагування та збереження документів, робота з таблицями та списками. Студенти повинні ознайомитись з інтерфейсом Microsoft Word та вивчити основні принципи його використання для створення документів.

1. Обладнання та програмне забезпечення

Обладнання:

- Комп'ютер з доступом до Microsoft Word (версія 2016, 2019 або Office 365).

Програмне забезпечення:

- Microsoft Word.

2. Завдання лабораторної роботи

2.1. Ознайомитись з інтерфейсом Microsoft Word.

2.2. Створити новий документ і зберегти його.

2.3. Внести зміни в документ: редагувати текст, змінювати шрифт і форматування.

2.4. Створити таблицю, заповнити її даними, змінити форматування.

2.5. Використати нумерацію та марковані списки для організації

інформації.

2.6. Зберегти та закрити документ.

3. Теоретичні відомості

3.1. Створення та збереження документу:

- Microsoft Word дозволяє створювати текстові документи з різними форматуваннями.

- Документ можна зберігати у різних форматах, таких як .docx, .pdf, .rtf.

- Щоб створити новий документ, потрібно вибрати "Файл" → "Новий" або натискати клавіші Ctrl + N.

- Збереження документу виконується через "Файл" → "Зберегти як" або Ctrl + S.

3.2. Редагування документів:

- Редагування тексту включає зміну шрифту, розміру, кольору, стилю (жирний, курсив, підкреслений).

- Для вставки, видалення, копіювання та вирізання тексту використовуються стандартні команди або комбінації клавіш (Ctrl + C, Ctrl + X, Ctrl + V).

3.3. Робота з таблицями:

- Таблиці використовуються для структурування даних у вигляді рядків та стовпців.

- Для створення таблиці потрібно вибрати "Вставка" → "Таблиця" і вказати розміри.

- Таблицю можна формувати, додавати або видаляти рядки та стовпці, змінювати розміри клітинок.

3.4. Використання списків:

- Списки допомагають організувати текстову інформацію в логічній послідовності.

- Існують марковані (з точками або іншими символами) та нумеровані списки.

- Для створення списку потрібно вибрати "Вставка" → "Список" або використовувати відповідні кнопки на панелі інструментів.

4. Хід виконання роботи

Крок 1: Створення нового документу

- Запустіть програму Microsoft Word.

- Для створення нового документу натисніть Ctrl + N або виберіть у меню "Файл" → "Новий".
- Встановити наступні параметри форматування:
 - параметри сторінки: розмір A4, орієнтація книжкова, праве поле 2 см, всі інші поля 1 см;
 - параметри абзацу:
 - вирівнювання по ширині;
 - перший рядок - 1,5 см;
 - міжрядковий інтервал - 1,5 строки;
 - відступ - 0;
 - інтервал до та після - 0 пт;
 - встановити наступні параметри символів:
 - шрифт Times New Roman;
 - розмір шрифту - 14 пунктів;
 - нарис курсив..

- Збережіть документ, вибравши "Файл" → "Зберегти як" і вкажіть ім'я та місце збереження (наприклад, на робочому столі).

Крок 2: Редагування документу

- Напишіть кілька рядків тексту (наприклад, абзац опису вашої лабораторної роботи).
- Виділіть частину тексту та змініть його шрифт, розмір, стиль (наприклад, зробіть текст жирним або курсивом).
- Використовуйте панель інструментів для зміни кольору тексту, встановлення відступів і вирівнювання.
- Знайти та замінити в doc. файлі всі літери «а» на літеру «щ».
- Розподілити текст з файлу в дві колонки з вирівнюванням по ширині. Зберегти документ у файлі doc.
- Використайте команду Ctrl + Z, щоб скасувати останню операцію (якщо потрібно).

Крок 3: Робота з таблицями

- Для створення таблиці натисніть на вкладку "Вставка" → "Таблиця".
- Виберіть розміри таблиці, наприклад, 5 рядки і 5 стовпці, та заповнити її за власним бажанням

Наявність товару

Назвах товару	10.09.2024		15.09.2024	
	Кількість, шт.	Ціна, грн.	Кількість, шт.	Ціна, грн.
Підсумок				

- Підрахувати загальну кількість товару та його середню ціну за кожний день засобами програми Microsoft Word.

- Додати в таблицю колонку «№ п/п» - номери по порядку, рядок з нумерацією колонок та відформувувати таблицю: зміст комірок розмістити посередині комірки, підібрати розміри комірок відповідно їх змісту, та промалювати межі комірок (зовнішні лінії – товсті, внутрішні лінії - тонкі).

- Використовуйте команду "Додати рядок" або "Додати стовпець", щоб змінити розміри таблиці.

- Зберегти створену таблицю у файлі docx.

Крок 4: Створення та форматування списків

- Для створення маркованого списку виділіть кілька рядків тексту і натисніть кнопку на панелі інструментів для маркованого списку (або натисніть Ctrl + Shift + L).

- У новому документі створити нумерований список студентів Вашої групи з 6 чоловік.

- Скопіювати список у нижче першого та зробіть список маркірованим, позначку маркеру виберіть за власним бажанням.

- Для створення нумерованого списку використовуйте кнопку для нумерації на панелі інструментів.

- Переміщуйте елементи списку, використовуючи функцію переміщення пунктів.

- Спробуйте використати вкладені списки для додаткової організації інформації.

- Зберегти створені списки у файлі doc.

Крок 5: Збереження та закриття документа

- Після завершення редагування збережіть документ, вибравши "Файл" → "Зберегти".

- Якщо хочете зберегти документ в іншому форматі, виберіть "Зберегти як" і вкажіть потрібний формат, наприклад, PDF.

- Закрийте документ, натиснувши "Файл" → "Закрити" або Ctrl + W.

Контрольні питання

1. Як можна змінити шрифт і колір тексту в Microsoft Word?
2. Як створити таблицю та змінити її розміри?
3. Які існують основні функції для форматування списків?
4. Як зберегти документ у різних форматах?
5. Як вставити та відформатувати таблицю в документі?

ЛАБОРАТОРНА РОБОТА № 7.2

Основи роботи в текстовому редакторі Microsoft Word

Мета роботи:

Метою лабораторної роботи є освоєння основних функцій текстового редактора Microsoft Word, навчання використанню формул, створенню шаблонів та змісту документа. Студенти повинні навчитися ефективно працювати з текстовими документами, використовуючи розширені можливості програми.

1. Обладнання та програмне забезпечення

Обладнання:

- Комп'ютер з доступом до Microsoft Word (версія 2016, 2019 або Office 365).

Програмне забезпечення:

- Microsoft Word.

2. Завдання лабораторної роботи

2.1. Ознайомитися з інтерфейсом Microsoft Word та його функціями.

- 2.2. Створити документ і додати формули.
- 2.3. Навчитися створювати та застосовувати шаблони для документів.
- 2.4. Створити зміст документа на основі заголовків.
- 2.5. Зберегти і закрити документ.

3. Теоретичні відомості

3.1. Використання формул у документах:

- У Microsoft Word є вбудований редактор для введення математичних формул. Для цього використовується вкладка "Вставка" → "Формула".
- Формули можуть бути введені як за допомогою спеціального інтерфейсу або за допомогою стандартних математичних виразів.
- Word також підтримує написання складних математичних виразів, таких як дроби, ступені, інтеграли, сумування тощо.

3.2. Створення шаблонів:

- Шаблон документа – це заздалегідь підготовлений макет документа, в якому заздалегідь налаштовані стилі, шрифти, відступи, поля, колонтитули та інші елементи форматування.
- Шаблони дозволяють значно скоротити час на створення стандартних документів (наприклад, звітів, договорів).
- Для створення шаблону потрібно зберегти документ як шаблон, вибравши формат .dotx або .dotm.

3.3 Створення змісту документа:

- Зміст (титульна сторінка) документа дозволяє автоматично створювати перелік розділів та підрозділів, що допомагає при роботі з великими текстами.
- Зміст створюється на основі стилів заголовків, які застосовуються до розділів документа.
- Щоб додати зміст, потрібно вибрати вкладку "Вставка" → "Зміст" або "Розмітка сторінки" → "Зміст".

4. Хід виконання роботи

Крок 1: Використання формул у документі

- Відкрийте Microsoft Word і створіть новий документ.
- Перейдіть на вкладку Вставка.

- Виберіть кнопку **Формула**. З'явиться вікно редактора формул.

- У редакторі введіть просту математичну формулу, наприклад, рівняння квадратичної функції: $y=ax^2+bx+cy$

- Розмістити її по середині рядку. В крайній правій позиції цього ж рядка поставити порядковий номер формули в круглих дужках: (1).

- Скопіювати формулу, вставити нижче введеної, збільшити розмір символів та присвоїти номер (2).

- Змініть вигляд формули за допомогою інструментів редактора: додайте числові індекси, змініть шрифт або колір.

- Для створення більш складних формул скористайтесь іншими доступними функціями, наприклад, введіть дроби або корені.

- Зберегти у файлі docx.

Крок 2: Створення шаблону документа

- Створіть новий документ і відформатуйте його згідно з вимогами (налаштуйте поля, шрифт, абзаци, нумерацію сторінок).

- Створити шаблон довідки про те, що особа є студентом ЗНТУ з визначенням спеціальності, групи та курсу навчання, зберегти його за ім'ям Shablon_lab.

Довідка

Надана _____

(призвищ, ім.'я та по батькові)

у тому, що він(вона) навчається у ЗНТУ на ____ курсі на спеціальності «_____» у групі _____

Довідка надана.

Декан факультету _____

(підпис)

П.І.Б

(розшифрування підпису)

- Для створення шаблону, зайдіть у вкладку **Файл** → **Зберегти як**.

- Виберіть тип файлу **Шаблон Word (.dotx)**.

- Виберіть місце для збереження шаблону, наприклад, на робочому столі.

- Назвіть шаблон і натисніть **Зберегти**.

- Тепер цей шаблон можна використовувати для створення нових документів за тим самим форматом.

Крок 3: Створення змісту документа

- Для створення змісту спочатку потрібно відформатувати заголовки. Виберіть текст, який має бути заголовком, та застосуйте стиль заголовка, наприклад, Заголовок 1, Заголовок 2 з панелі стилів на вкладці Головна.

- Після цього перейдіть на вкладку Вставка і натисніть Зміст.

- Виберіть один із запропонованих форматів змісту. Зміст автоматично відобразить заголовки, відформатовані в попередньому кроці.

- За потреби відредагуйте зміст, додавши нові заголовки або змінивши їхні назви.

- Для оновлення змісту натискайте Оновити таблицю змісту після внесення змін у заголовки.

Зберегти документ у файлі doc

- Ввести у колонтитули назву поточного файлу ліворуч та номер поточної сторінки праворуч, номер сторінки почати з 5.

- Зберегти зміни, які внесені у поточний документ.

- Створити зміст цього документу у кінці документу

- Вставити створений документ у стандартну рамку (файли Проста рамка та Складна рамка у папці Commons) при цьому перший лист документу розмістити у файлі Складна рамка, а всі інші у файлі Проста рамка. Поєднати створені листи у одному документі.

Крок 4: Збереження та закриття документа

- Після виконання всіх кроків збережіть документ, вибравши Файл → Зберегти.

- Збережіть документ у форматі .docx або .pdf, в залежності від вимог.

- Закрийте документ, натиснувши Файл → Закрити або Ctrl + W.

Контрольні питання

1. Як додати математичну формулу в документ Microsoft Word?

2. Яка різниця між шаблоном і звичайним документом у Microsoft Word?

3. Як створити зміст на основі заголовків?

4. Які основні способи редагування формул у Word?

5. Як оновити зміст документа після внесення змін у заголовки?

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ ПОСИЛАННЯ

1. Синявський В. В., Єгупов М. Г. Системи та технології штучного інтелекту. – Київ: КПІ ім. І. Сікорського, 2018. – 320 с.
2. Stallings W. Computer Organization and Architecture: Designing for Performance. – Pearson, 2021. – 880 p.
3. Гончаров Н. В., Чумаченко І. С.. Основи обчислювальної техніки. – Харків: ХНУРЕ, 2019. – 280 с.
4. Hayes J. P. Computer Architecture and Organization. – McGraw-Hill Education, 2018. – 848 p.
5. Щербаков Г. В., Лихачов Ю. М. Основи обчислювальної техніки. – Київ: Вища школа, 2018. – 300 с.
6. Пушкар В. С., Резніченко В. В. Логіка та цифрові системи. – Київ: КНУТШ, 2019. – 260 с.
7. Столяров І. І. Основи двійкових обчислень. – Львів: ЛДУ, 2017. – 220 с.
8. Скуратівський В. Комп'ютерна пам'ять та її організація. – Київ: Вища школа, 2018. – 280 с.
9. Офіційна документація виробників процесорів та материнських плат (Intel, AMD, ASUS, MSI).
10. Угринович О. В. Основи обчислювальної техніки. – Київ: КНЕУ, 2020. – 250 с.
11. Станкевич В. І. Архітектура комп'ютерних систем. – Харків: ХНУРЕ, 2018. – 320 с.
12. Tanenbaum A. S. Computer Organization and Design. – Morgan Kaufmann, 2018. – 540 p.
13. Hennessy J. L., Patterson D. A. Computer Architecture: A Quantitative Approach. – Morgan Kaufmann, 2019. – 960 p.
14. Опанасенко А. С. Моделювання та оптимізація систем пам'яті. – Київ: Вища школа, 2017. – 200 с.
15. Tanenbaum A. S., Bos H. Modern Operating Systems. – Pearson, 2015. – 912 p.

16. Upton E., Halfhill T. R. Raspberry Pi 4 User Guide. – Wiley, 2020. – 300 p.
17. Gagne G., Roussel P., Molnar D. Computer Organization and Design: The Hardware/Software Interface. – Morgan Kaufmann, 2018. – 672 p.
18. "Пам'ять комп'ютера: типи, характеристики, принципи роботи" [Електронний ресурс]. URL: <https://www.computerhardware.com/memory-types>.
19. Ласкун В. І., Степанов В. В. Microsoft Word для початківців. – Київ: Освіта, 2019. – 250 с.
20. Захарова В. Г. Практикум з використання Microsoft Word. – Харків: Освітня думка, 2020. – 180 с.
21. Шаров О. А. Microsoft Office для студентів і фахівців. – Львів: Літера, 2021. – 200 с.
22. Захарова В. Г. Microsoft Word. Повний посібник для користувачів. – Харків: Вища школа, 2020. – 350 с.
23. Шаров О. А. Практикум з використання Microsoft Word для студентів технічних спеціальностей. – Львів: Літера, 2021. – 200 с.
24. Онлайн-ресурси: tomshardware.com, anandtech.com, overclockers.ua, arstechnica.com.