

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ІНЖЕНЕРНИЙ НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ
ІМ. Ю.М. ПОТЕБНІ
ЗАПОРІЗЬКОГО НАЦІОНАЛЬНОГО УНІВЕРСИТЕТУ

З.А. Ніконова
С.Л. Шмалій

МЕТОДИЧНІ РЕКОМЕНДАЦІЇ

**ДО ВИКОНАННЯ ПРАКТИЧНИХ РОБІТ З ДИСЦИПЛІНИ
«МІКРОЕЛЕКТРОННІ ІНФОРМАЦІЙНІ СИСТЕМИ»**

**Для студентів освітнього рівня «Магістр» за спеціальністю 176
«Мікро- та наносистемна техніка»**

Обговорено та ухвалено на засіданні
кафедри електроніки, інформаційних систем та
програмного забезпечення
Протокол №1 від 28.08.23

Запоріжжя
2023

ЗМІСТ

ВСТУП

ПЕРЕЛІК ПРАКТИЧНИХ ЗАВДАНЬ

1. Практична робота №1 Моделювання мікроелектронної інформаційної системи.
2. Практична робота №2 Способи кодування інформації.
3. Практична робота №3 Зв'язування і впровадження об'єктів у MS Excel.
4. Практична робота №4 Побудова блок – схеми мікроелектронної інформаційної системи керування електричними мережами.
5. Практична робота №5 Рішення логічних задач засобами алгебри логіки.
6. Практична робота №6 Створення базових таблиць в *ACCESS*.
7. Практична робота №7 Розрахунок параметрів та розробка технічного завдання для індивідуального проєкту.

Рекомендована література

ВСТУП

Мета викладання дисципліни «Мікроелектронні інформаційні системи» це підготовка висококваліфікованих фахівців, які отримають спеціальні теоретичні та практичні знання, вміння і навички для вирішення завдань з розроблення, розгортання та експлуатації сучасних мікро- та наноелектронних інформаційних систем.

Завданням є формування стійких уявлень про мікроелектронні інформаційні системи; конструкторсько – технологічні засади їх створення із заданим функціональним призначенням і переліком технічних параметрів окремих вузлів, склад модулів та підсистем; визначення характеристик та шляхи їх покращення; вдосконалення сучасних та розробку нових видів інформаційних систем.

У результаті вивчення навчальної дисципліни студент **зможе:**

- Збирати необхідну інформацію, використовуючи науково-технічну літературу, бази даних та інші джерела, аналізувати і оцінювати її.
- Забезпечувати професійний розвиток членів колективу з урахуванням світового досвіду і вимог до персоналу в сфері розробки та експлуатації мікро- та наноелектронних систем.
- Досліджувати процеси у мікро- та наноелектронних системах, приладах й компонентах з використанням сучасних експериментальних методів та обладнання, здійснювати статистичну обробку та аналіз результатів експериментів.
- Практикувати інформаційний та науковий пошук, використовувати бази даних і знань, критично осмислювати та інтерпретувати результати, робити висновки та формувати напрями дослідження з урахуванням вітчизняного й закордонного досвіду.

ПРАКТИЧНА РОБОТА №1

Тема: Моделювання мікроелектронної інформаційної системи

Мета: Ознайомлення з хмарними технологіями обробки даних при моделюванні мікроелектронних інформаційних систем на базі мікроконтролерів

1.1 Публічні хмарні сервери для IoT пристроїв

1.1.1 MQTT брокери

Протокол MQTT - це мабуть найперше, на що натикаєшся в пошукових системах, коли шукаєш спосіб віддаленого управління різними "розумними пристроями". Для даного протоколу існує багато готових бібліотек для Arduino IDE. Мобільних програм - достатньо, щоб вибрати і для смартфона, і для контрольної панелі, зробленої з стародавнього планшета. Тож це, напевно, найпопулярніший варіант для реалізації віддаленого керування. Хмарних серверів також досить багато, називаються вони MQTT брокерами. Короткі відомості наведені в таблиці 1.1, детальніше кожен брокер буде описано нижче.

Таблиця 1.1 — Перелік MQTT брокерів.

Сервіс	Версія протоколу та порти	Тип авт*	Web	API	Префікс топиків	Найбільш бюджетний тарифний план (за місяць)		Особливості
						Ціна	Обмеження	
cloudmqtt.com	MQTT 3.1.1 TCP, TLS, WS+TLS	L+P	✓	✗		5 \$	25 підключень; 20 Kbit/s	Немає безкоштовних тарифів
mqtt.by	TCP	L+P	✗	✗	/user/xx/	FREE	без обмежень	Потрібен певний префікс для топиків
flespi.io	MQTT 5 TCP, TLS	ТОК	✓	✓		FREE	100 підключень; 1 млн повідомлень за хвилину; 100MB retained повідомлень	Складний інтерфейс та керування доступом. Платні послуги на запит.
shiftr.io	MQTT 3.1.1 TCP, TLS, WS+TLS	L+P	✓	✓		FREE	100 підключень; 5000 повідомлень за секунду; 6 годин роботи на добу	Цікавий інтерфейс моніторингу роботи сервера
HiveMQ Public Broker	MQTT 3.1.1 TCP, WS	✗	✗	✗		FREE	без обмежень	Публічний сервер, без авторизації
HiveMQ Cloud Cluster	MQTT 3.1.1 TLS only	L+P	✗	✗		FREE	100 підключень; 10 GB щомісяця; термін зберігання до 3 днів	Невеликий час зберігання retained повідомлень, при перевищенні ліміту basic погодинна ставка

mosquitto.org	MQTT 3.1.1 TCP, TLS, WS, WS+TLS	×	×	×		FREE	без обмежень	Публічний сервер без авторизації. Ідеальний для тестування TLS
IBM Cloud (Bluemix)	TCP, TLS	?	✓	✓	топики в строго опр. форматі	FREE	500 підключень; 200 MB на метрику	IoT платформи. Дуже складний варіант. Топіки у строго визначеному форматі
mqtt.dioty.co	MQTT 3.1.1 TCP, TLS, WS+TLS	L+P	×	×	/e@mail/	FREE	без огр. по клієнтам; 50000 повідомлень на місяць (~70 на годину)	Потрібний певний префікс для топиків. Обмеження приблизно 70 повідомлень на годину
Solace.Cloud (PubSub+)	TCP, TLS	?	✓	✓		FREE	50 підключень; 1 GB даних на місяць; 4 GB збережених даних	IoT платформи. Дуже складний варіант.
myqthub.com	MQTT 3.1.1 TCP, TLS	L+P	✓	✓		FREE 50 підключень; 10000 повідомлень на день; 10 MB retained	50 підключень; 10000 повідомлень на день; 10 MB retained	Складна система обмежень

Примітка: у стовпці "Тип авт." вказано тип авторизації:

- **×** – без авторизації (публічний брокер)
- **L+P** – авторизація за логіном та паролем
- **TOK** – авторизація за токеном, що генерується сервісом
- **SRT** – авторизація за сертифікатом користувача

Сервіс cloudmqtt.com досить зручний та простий, все інтуїтивно зрозуміло. У 2018 році сервіс допускав на безкоштовному тарифному плані підключення до 10 пристроїв, але на цей час CloutMQTT.com став повністю платним. Найдешевший платний тарифний план Humble Hedgehog коштує на момент написання статті 5 \$ на місяць. У даному пакеті всього – 25 підключень.

Плюси: багато прикладів на різних сайтах.

Мінуси: немає безкоштовних тарифів, все лише за гроші.

Простий у використанні брокер mqtt.by. Про обмеження не сказано нічого. Підтримує авторизацію, але є можливість зробити деякі топіки публічними, тобто відкрити для решти користувачів сервісу. Але немає підтримки безпечного з'єднання (TLS), це може бути критично для деяких програм. Кожен публікований топик повинен починатися з префіксу "/user/ім'я_користувача", що дещо ускладнює налаштування мобільних mqtt клієнтів, але не суттєво. З плюсів – на сервері є загальнодоступні топіки з датою та часом, що дуже зручно для контрольної панелі на основі планшета, яка працює 24 години на добу – разом і годинники ще одні.

Плюси: без будь-яких обмежень, є публічні топіки, можна опублікувати свої топіки.

Мінуси: потрібний певний префікс для топіків, немає SSL/TLS шифрування підключень.

Сервіс `mqtt.flespi.io` допускає аж 100 конкуруючих підключень одночасно і величезні обсяги повідомлень, що передаються і зберігаються. Однак сервіс не такий зручний, як CloudMQTT. Контрольна панель не дуже зручна (хоча й удосконалюється), налаштування прав доступу на перший погляд ускладнено. Авторизація і пристроїв, і смартфонів виконується як у всіх, а по генерованим токенам з обмеженим терміном дії. Строго кажучи, це далеко не тільки MQTT брокер, так що, можливо, іноді на різних смартфонах перестав оновлюватися інформація в деяких топах.

Плюси: до 100 підключень.

Мінуси: не дуже зручний, складний у налаштуванні спочатку, доступ по токенам з обмеженим терміном дії.

Сервіс `shiftr.io` - досить цікавий брокер. Сервіс підтримує MQTT протокол версії 3.1.1, підключення можливе через TCP, SSL/TLS та WebSocket з'єднання. Крім того, можлива взаємодія з брокером по HTTP-протоколу, наприклад, це може бути корисно для отримання даних на будь-який сайт або додаток. Сервіс надає вже готовий Desktop-додаток для роботи з брокером. Головною ж особливістю брокера є досить цікавий інтерфейс управління, на якому в динамічному режимі відображаються у вигляді схеми підключені пристрої, генеровані ними топіки, підписки на них, і дані.

На безкоштовному тарифі можливе підключення до 100 пристроїв одночасно та до 5000 повідомлень за секунду. Крім того, можна створити кілька екземплярів брокера та працювати з ними паралельно. Для особистого застосування цілком прийнятні умови. Але, на превеликий жаль, є ще одне обмеження: безкоштовні екземпляри не можуть працювати більше 6 годин на добу, після чого вони переводяться в сплячий режим.

Проблеми із підключенням до `shiftr.io`: на тестовому пристрої проявляється як періодичні обриви зв'язку клієнтів (ESP32 та смартфона) від брокера, панель управління у цей час видає "Instance Unreachable".

З інших особливостей варто відзначити відсутність префікса перед топіками (тобто немає необхідності ставити ім'я користувача або слеш (/) перед ім'ям топіка: замість `"/device/lamp0"` слід писати `"device/lamp0"`, тобто цей брокер цілком підходить для роботи з IoT Manager. Довжина топіка обмежена 128 символами, а розмір повідомлення, що публікується, не може бути більше 64 кілобайт. Кількість підписок на кожне активне підключення не повинна перевищувати 100. Є можливість перевести створений Вами приватний екземпляр у публічний доступ, тобто надати доступ до нього всім бажаючим (наприклад, для демо-версії Вашого проекту).

Плюси: до 100 підключень; немає необхідності в префіксі топіків (підходить для IoT Manager).

Мінуси: сервер працює лише 6 годин на добу; досить часті обриви зв'язку з брокером (Instance Unreachable).

Повністю безкоштовний публічний брокер HiveMQ Public Broker без жодних обмежень. Принаймні, на головній сторінці брокера про них не написано. Але є - цей брокер не має жодної авторизації, тобто взагалі ніякої. Тобто всі повідомлення, що пересилаються між пристроєм і користувачем, може прочитати будь-хто. У цьому не було б великої трагедії, якщо це тільки якісь показники температури і вологості, секрету вони не представляють. Але коли справа стосується віддаленого управління "розумним будинком", то такий брокер вже в принципі не підходить.

Плюси: безкоштовний, без обмежень щодо підключень.

Мінуси: підходить тільки для тестування та метеостанцій без віддаленого керування.

Комерційний брокер HiveMQ Cloud Cluster з авторизацією. HiveMQ Cloud підтримує всю специфікацію MQTT, включаючи QoS 1 та 2, збережені повідомлення, загальні передплати, властивості користувачів та негативні підтвердження. На безкоштовному тарифному плані Basic допускається 100 конкурентних підключень, 10 гігабайт трафіку щомісяця, максимальний розмір повідомлення 5 мегабайт, при цьому термін зберігання повідомлень не більше трьох днів. На платних тарифах (при перевищенні ліміту) ціни погодинні та можуть сягати майже 8\$.

Ім'я сервера та порт підключення можна дізнатися на вкладці Overview після реєстрації у сервісі. Сервер дозволяє налаштувати облікові записи для кожного пристрою окремо, а також має вбудовані інструменти для початку роботи з брокером. З опису не зовсім ясно, чи підтримуються TLS-з'єднання чи ні.

Плюси: 100 конкурентних підключень на базовому тарифі.

Мінуси: невеликий час зберігання retained повідомлень, при перевищенні ліміту basic погодинна ставка.

Ще один публічний брокер без авторизації - mosquitto.org. Можна не повторюватися, все те саме, що й у HiveMQ Public Broker. Зверніть увагу на те, що всі ці сервери підтримують шифрування TLS аж ніяк не означає, що дані користувача захищені. Захищений канал передачі між пристроєм і сервером, але дані знаходяться у відкритому доступі.

Особливістю даного сервера і те, що у ньому реалізовані порти з різними варіантами підключення по SSL:

- 1883: MQTT, незашифрований
- 8883: MQTT, зашифрований
- 8884: Зашифрований MQTT потрібен сертифікат клієнта
- 8887: Зашифрований MQTT, термін дії сертифіката сервера навмисно закінчився

- 8080: MQTT через WebSockets, незашифрований

- 8081: MQTT через WebSockets, зашифрований

Це дозволяє тестувати процес підключення до MQTT у різних ситуаціях.

Плюси: відмінно підходить для налагодження підключення; безкоштовний, без обмежень щодо підключень.

Мінуси: підходить тільки для тестування та метеостанцій без віддаленого керування.

IBM Cloud (Bluemix) -це не mqtt брокер, це ціла платформа "Internet of Things Platform". Але в її складі є і mqtt брокер. На безкоштовному тарифному плані "Lite" допускається аж до 500 зареєстрованих пристроїв та по 200 МБ інформації на кожну метрику. Документація є, але її дуже багато. По-друге, платформа дозволяє публікувати топики строго у певному форматі: "iot-2/evt/event_id/fmt/format_string". Так само з командами: "iot-2/cmd/command_id/fmt/format_string". І все – тобто тільки номер повідомлення і саме повідомлення. А ось форматів повідомлень, що передаються, вже більше – "json", "xml", "txt", та "csv". Можливо, комусь такий підхід здається дуже правильним і зручним.

Плюси: дуже великі ліміти щодо підключень.

Мінуси: дуже складний для новачка, обмеження по топіках.

Вже повністю безкоштовний брокер - mqtt.dioty.co. На сторінці параметрів підключення написано наступне: "Цей брокер MQTT в даний час працює в бета-версії. На цьому етапі бета-тестування ми не стягуватимемо плату, якщо ви перевищите ліміт у 50000 повідомлень на місяць. Однак ми залишаємо за собою право тимчасово призупинити дію вашого облікового запису, якщо ви впливаєте на обслуговування інших користувачів. У цьому випадку ми завжди будемо зв'язуватися з вами електронною поштою, щоб спробувати зрозуміти ваші конкретні потреби і те, як ми можемо їх задовольнити.". 50000 повідомлень на місяць – це приблизно 70 повідомлень на годину, що не так вже й багато, якщо врахувати, що один пристрій може публікувати за сеанс відправки даних з датчиків до 50 повідомлень. Брокер підтримує авторизацію та TLS-підключення. Реєстрація прив'язана до облікового запису Google. Є невелике обмеження – усі топики користувача повинні починатися з адреси електронної пошти, через яку ви зареєструвалися у сервісі, наприклад: "/user12@i.ua/.../.../...". Але ця проблема більш-менш істотна тільки при налаштуванні MQTT клієнтів на смартфоні (більше тексту набирати), та й то не завжди.

Плюси: повністю безкоштовний.

Мінуси: потрібен певний префікс для топіків, обмеження приблизно 70 повідомлень на годину.

Solace.Cloud (PubSub+) - це не так MQTT брокер, як ціла IoT платформа. На безкоштовному тарифному плані допускається до 50 підключень, до 1 GB даних, що передаються в місяць, до 4 GB збережених даних (retained повідомлень). Але навіть у free версії дуже багато налаштувань – у ньому складно розібратися.

Плюси: до 50 підключень.

Мінуси: досить складний в освоєнні.

На безкоштовному тарифному плані myqthub.com припускає: до 100 користувачів, але до 50 підключень; 10MB даних, що зберігаються; до 50

підписок на підключення, максимум 250 підписок; 300 повідомлень за хвилину, але не більше 800 на годину і не більше 10000 на день. Якщо виходити з регулярних (кожних п'ять хвилин) публікацій, виходить не більше 34 повідомлень за раз. Є й ще обмеження – докладніше дивіться на головній сторінці сервісу. Обмеження, втім, виглядають цілком реальними домашнього використання.

Плюси: до 50 підключень.

Мінуси: обмеження щодо кількості повідомлень.

Таким чином, можна зробити висновок, що MQTT простий та зручний спосіб запису даних та подальшої обробки на стаціонарній техніці, але в розглянутих сервісах не має функцій обробки графічної інформації. Для перегляду графіків є інші послуги, налагодити роботи з якими також не становить особливих труднощів. Деякі з них представлені нижче.

1.2 Сервіс beebotte

На головній сторінці цього сервісу написано наступне: “підключення всього в режимі реального часу з використанням багатого API, що підтримує REST, WebSockets та MQTT”. У безкоштовному режимі кількість каналів та ресурсів не обмежена, але обмежена кількість повідомлень: 0.05 Million per day 1.5 Million per month. Для повідомлень, що зберігаються, ліміти ще менше: 5000 per day 150000 per month. Крім того, повідомлення зберігаються лише максимум три місяці, після чого видаляються. Про мобільний клієнт для смартфона поки що нічого не відомо.

Контрольні питання

1. Які сервіси є безкоштовними?
2. Які дані можуть записуватись на сервери?
3. Як можна використовувати мобільні пристрої в якості клієнтів?

ПРАКТИЧНА РОБОТА №2

Тема: Способи кодування інформації

Мета: Вивчення способів обробки інформації на базі мікроконтролерів Arduino (приклад- мікроконтролери ESP8266)

2.1 Опис чипу ESP8266EX

Чіп ESP8266EX компанії Espressif - високо-інтегроване Wi-Fi SoC рішення, що задовольняє запити індустрії Інтернету речей у низькому енергоспоживанні, компактності та надійності з режимами 802.11 b/g/n/d/e/i/k/r.

Маючи повноцінний Wi-Fi та мережевий стек, чіп ESP8266EX може як виконувати програми самостійно, так і працювати під керуванням зовнішнього мікроконтролера. Працюючи самостійно, ESP8266EX виконує програму, завантажуючи її із зовнішньої флеш-пам'яті. Вбудований високошвидкісний кеш підвищує продуктивність системи та дозволяє ефективно використовувати оперативну пам'ять. Працюючи під керуванням зовнішнього мікроконтролера, ESP8266EX може виступати у ролі Wi-Fi адаптера, передаючи дані через SPI, SDIO, I2C чи UART інтерфейси.

ESP8266EX містить антенний перемикач, узгоджуючий трансформатор, підсилювач потужності, малошумливий підсилювач, фільтри, модулі керування живленням. Компактна конструкція та високий ступінь інтеграції дозволяють мінімізувати розмір друкованої плати та кількість зовнішніх компонентів.

ESP8266EX містить розширену версію 32-бітного процесора Lx106 фірми Tensilica серії Diamond та вбудовану оперативну пам'ять (SRAM). Він може бути з'єднаний із зовнішніми датчиками та іншими пристроями через інтерфейси введення/виведення загального призначення (GPIO). Пакет програм для розробки програм (SDK) містить зразки програм для різноманітних застосувань.

Smart Connectivity Platform (ESCP) компанії Espressif Systems забезпечує складні функції, включаючи швидкий перехід між режимом сну та режимом пробудження для цілей енергоефективності, адаптивне підстроювання радіо тракту для роботи з низьким енергоспоживанням, удосконалену обробку сигналу, придушення паразитних сигналів та механізм радіопередачі для стільникового зв'язку, Bluetooth, DDR, LVDS, пом'якшення LCD перешкод.

Існує безліч модифікацій плат, які називаються зазвичай від ESP-01 до ESP-12. Сьогодні вже з'явилися ще інші назви плат від сторонніх розробників. Відмінності в платах полягає в основному в портах вводу-виводу, кількості флеш-пам'яті, виду конекторів і т.п. Процесор - той самий, отже з погляду програмування немає значення яку плату програмувати.

Специфікація ESP8266:

- Напруга живлення: 3,3 В
 - Енергоспоживання: 10 мкА ... 170 мА
 - Флеш-пам'ять: до 16 мб максимум (зазвичай 512 кб)
 - Процесор: Tensilica L106, 32 біта
 - Швидкість процесора: 80...160 МГц
 - ОЗП: 32 кб + 80 кб
 - Порти введення-виведення загального призначення: 17
(мультиплексовані з іншими функціями)
 - Максимальна кількість підключень TCP: 5
- На рис. 2.1 Представлена функціональна схема ESP8266EX.

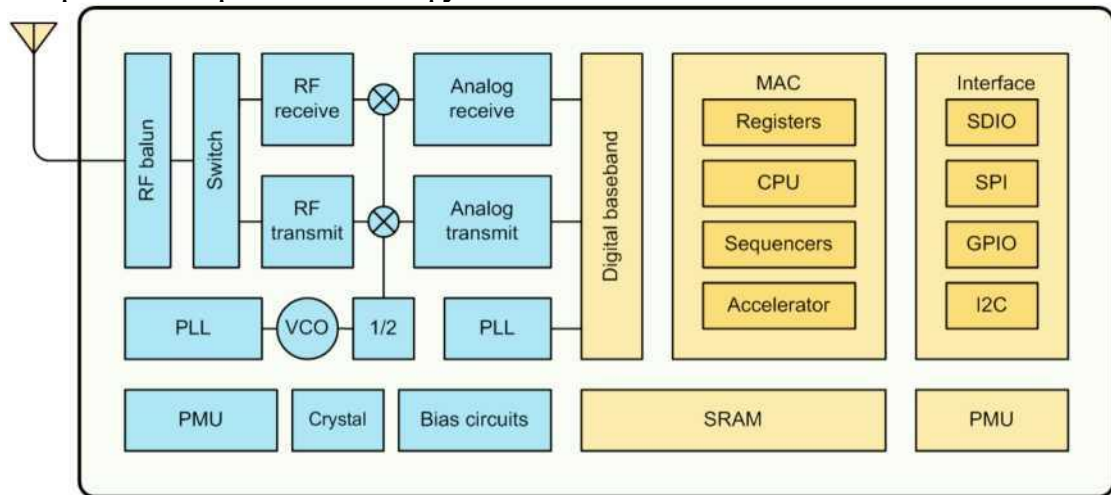


Рисунок 2.1 - Функціональна схема ESP8266EX

ESP8266EX має 17 виводів GPIO, які можуть бути призначені для різних функцій шляхом програмування відповідних регістрів.

Кожен GPIO є двоспрямованим та неінвертованим, має внутрішні підтяжки на напругу живлення або на землю, і може бути переведений у стан з високим імпедансом. Коли GPIO налаштовано як вхід, вхідні значення можуть бути прочитані з регістрів. Вхід також може бути використаний як джерело переривань за рівнем або перепадом напруги.

Ці виводи можуть бути поєднані з іншими функціями, такими як I2C, I2S, UART, PWM, IR тощо.

Для роботи в режимі енергозбереження виводи GPIO можуть підтримувати стан. Наприклад, коли чіп вимкнено, всі виходи можуть бути вимкнені.

Додаткова функція утримання за потреби може бути вбудована у IO. Коли IO не управляється внутрішньою або зовнішньою схемою, функція утримання може бути використана для збереження останнього використаного стану. У зв'язку з цим, зовнішня схема управляюча контактом, повинна подавати струм більший, ніж струм зворотного зв'язку. Необхідна сила струму невелика - близько 5 мкА.

Таким чином, модуль є закінченим функціональним пристроєм, який здатний використовувати технології хмарних обчислень, (наприклад, thingspeak.com).

Порядок виконання практичної роботи

1. Засвоїти питання: призначення портів мікроконтролера, які інтерфейси він підтримує для підключення периферійних пристроїв.
2. Визначити види датчиків, які можна підключати до мікроконтролера.
3. Огрунтувати обмеження по напрузі при використанні інтерфейсів.

Контрольні питання

1. До яких портів можна підключати периферійні пристрої?
2. На яких частотах працює процесор?
3. Які існують модифікації плат?
4. Які форми програмування мікроконтролера?
5. Яка напруга живлення мікроконтролера?

ПРАКТИЧНА РОБОТА №3

Тема: Зв'язування і впровадження об'єктів у MS Excel

Мета: Побудова діаграми Парето у Microsoft Excel

У Excel 2016 розробники додали кілька нових діаграм. Одна з них – гістограма Парето, яка відбиває графічне зображення відомого принципу Парето чи закону 20/80. Насправді цей спосіб досить багато використовуюється, оскільки його простота, і водночас ефективність, приносять багато переваг в аналізі даних.

Діаграма (крива) Парето містить як стовпці, відсортовані за спаданням, так і лінію, що означає загальний сукупний відсоток. За допомогою діаграми закону Парето в бізнес-аналізі наочно показують, наприклад, які товари або клієнти приносять найбільший прибуток. З її допомогою можна легко побачити найпоширеніші проблеми.

Порядок виконання практичної роботи залежить від версії Excel

3.1 Діаграма Парето в Excel 2013 та ранніх версіях

Не в усіх користувачів встановлено новітню версію Excel. Розглянемо, як побудувати діаграму Парето в Excel 2013 і попередніх версіях на конкретному прикладі. Є таблиця, що містить прибуток за товарами.

	A	B	C	D
1	Товар	Прибуток	Накопичена частка прибутку	Поріг
2	Персик	455 710		80%
3	Ківі	88 173		80%
4	Яблуко	82 485		80%
5	Картопля	355 526		80%
6	Капуста	870 960		80%
7	Полуниця	144 152		80%
8	Морква	713 811		80%
9	Абрикос	206 326		80%
10	Баклажан	205 949		80%

Для побудови діаграми потрібно:

1. Відсортувати дані за спаданням у стовпі «Прибуток», щоб значення, що мають найбільший внесок, були на початку списку. Можна ознайомитись з матеріалом, {як виконати сортування даних в Excel}.

2. Розрахувати стовпець з накопиченими частками за такою формулою:

$$=SUM(\$B\$2:B2)/SUM(\$B\$2:\$B\$10)$$

	A	B	C	D
1	Товар	Прибуток	Накопичена частка прибутку	Поріг
2	Капуста	870 960	28%	80%
3	Морква	713 811	51%	80%
4	Персик	455 710	65%	80%
5	Картопля	355 526	77%	80%
6	Абрикос	206 326	83%	80%
7	Баклажан	205 949	90%	80%
8	Полуниця	144 152	95%	80%
9	Ківі	88 173	97%	80%
10	Яблуко	82 485	100%	80%

3. Виділити всі дані та на вкладці «Вставлення», у розділі діаграми, будемо звичайну гістограму. Легенду можна видалити або розмістити у зручному місці.

4. Перенести ряди із відсотками на допоміжну вісь.

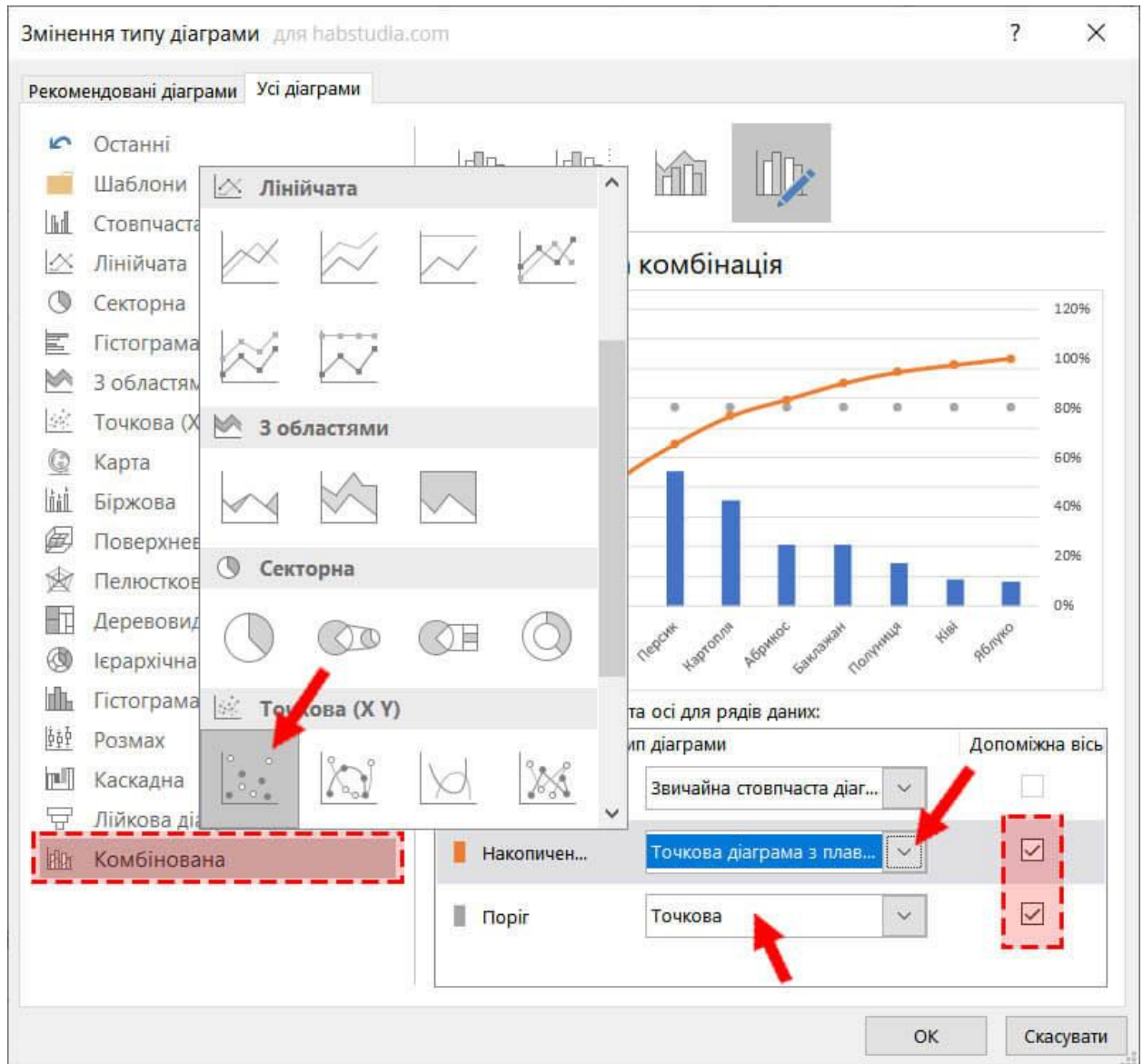
Виділити ці ряди на отриманій діаграмі за допомогою миші складно, тому що їх погано видно на тлі великих стовпців прибутку. Тому, щоб виділити прибуток, потрібно виділити діаграму, і на вкладці «Формат» у розділі «Поточний виділений фрагмент» вибрати потрібні ряди.

The screenshot shows the Excel interface with the 'Формат' (Format) ribbon active. The 'Область діаграми' (Chart Area) dropdown menu is open, displaying a list of options for selecting chart elements. A red arrow points to the 'Область діаграми' option. Below the menu, a portion of the data table from the previous image is visible, with columns C and D highlighted in red. The table data is as follows:

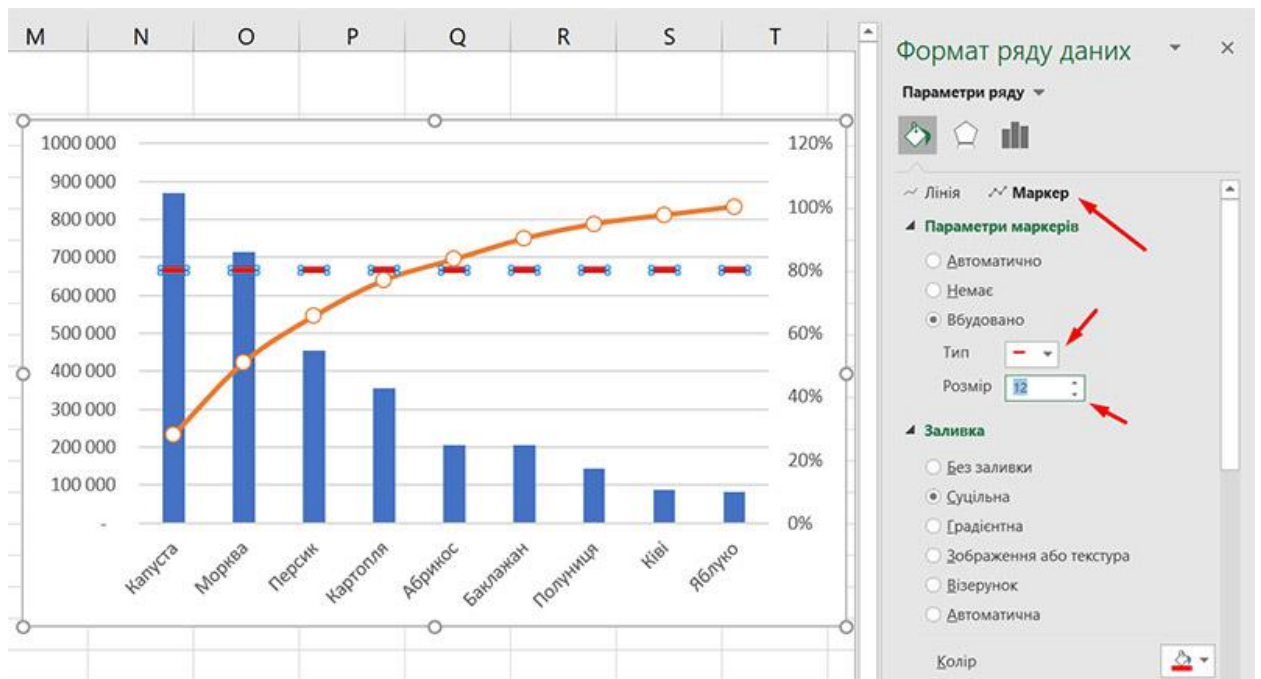
	C	D	E	
	опичена частка прибутку	Поріг		
2	Капуста	870 960	28%	80%
3	Морква	713 811	51%	80%
4	Персик	455 710	65%	80%
5	Картопля	355 526	77%	80%
6	Абрикос	206 326	83%	80%
7	Баклажан	205 949	90%	80%
8	Полуниця	144 152	95%	80%
9	Ківі	88 173	97%	80%
10	Яблуко	82 485	100%	80%

5. Клацнути по виділеному ряду правою кнопкою миші та вибрати команду «Формат ряду даних...» і у вікні вибрати опцію «По допоміжній осі».

6. Змінити тип діаграми зі стовпців на лінію. Для цього натисніть на цьому ряду правою кнопкою миші і виберіть команду «Змінити тип діаграми для рядів...». Як новий тип діаграми для ряду «Поріг» вибрати Точкова, а для «Накопичений прибуток» – Точкова діаграма з плавними лініями.



7. Відформатувати ряди «Прибуток» та «Накопичений прибуток» так, щоб вони стали схожими на лінію відсічення, а не на дані. Для цього можна змінити маркери, зробити лінію червоною пунктирною тощо. Всі ці операції виконуються, якщо клацнути правою кнопкою миші по ряду і вибравши команду «Формат ряду даних...».



Основні особливості діаграми:

* Кожен синій стовпець гістограми є прибуток за товаром в абсолютних величинах.

* Помаранчевий графік є накопичений відсоток прибутку (тобто частку прибутку зростальним підсумком).

* На умовному кордоні 80% вказана порогова горизонтальна лінія. Всі товари лівіше точки перетину цієї лінії з графіком накопиченого прибутку – приносять 80% грошей, всі товари правіше – 20%.

3.2 Побудова діаграми Парето в Excel 2016 та вище

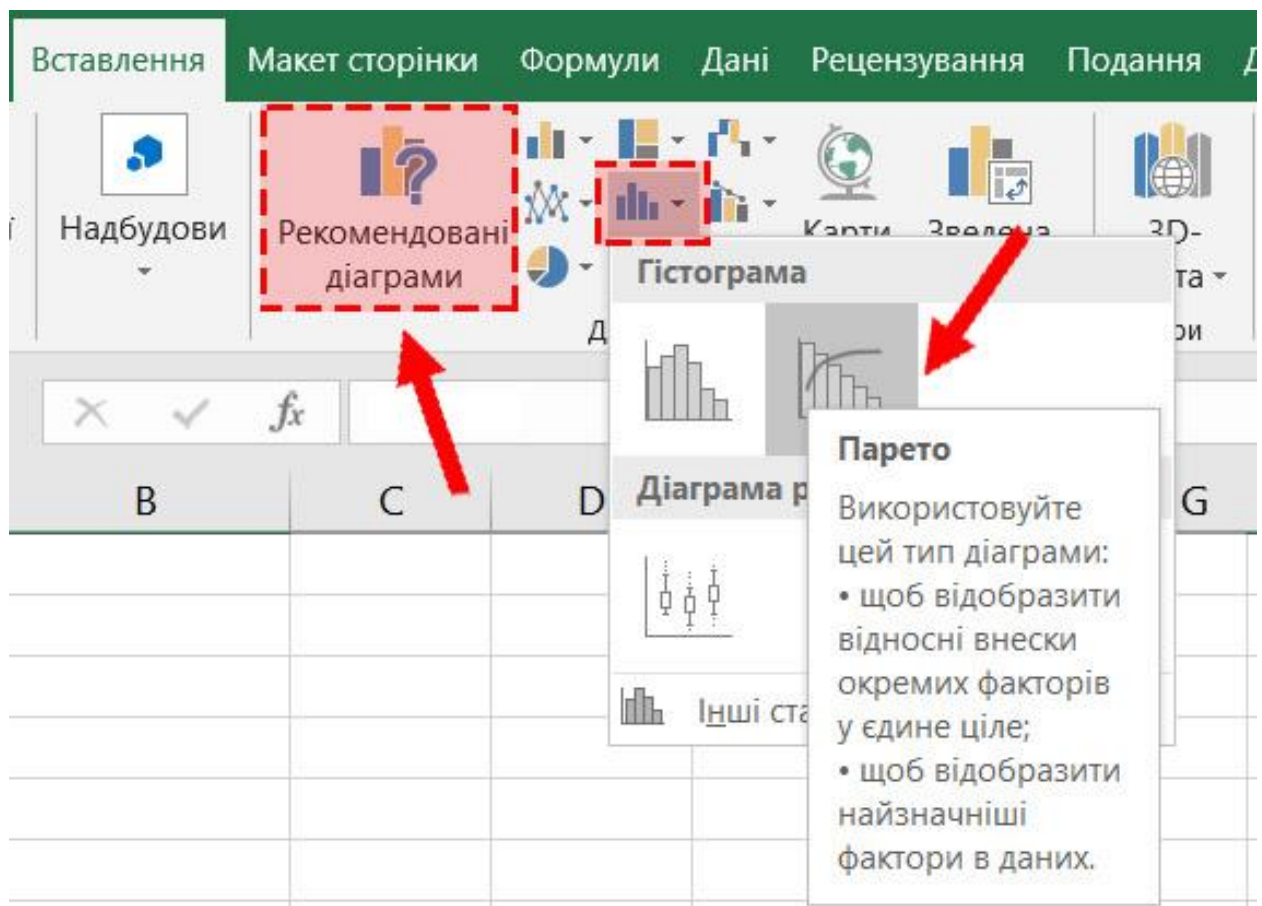
Загалом побудови діаграми Парето вказаним способом не викликає труднощів. Однак при оновленні даних, як мінімум сортування даних доведеться провести заново. Можливо, доведеться налаштувати нові діапазони та формули, якщо кількість даних змінилася.

У Excel 2016 більше не потрібно сортувати дані, розраховувати окремий стовпець із накопиченими частками та колупатися в налаштуваннях. Достатньо мати вихідні дані, щоб у пару натискань мишкою побудувати гістограму Парето.

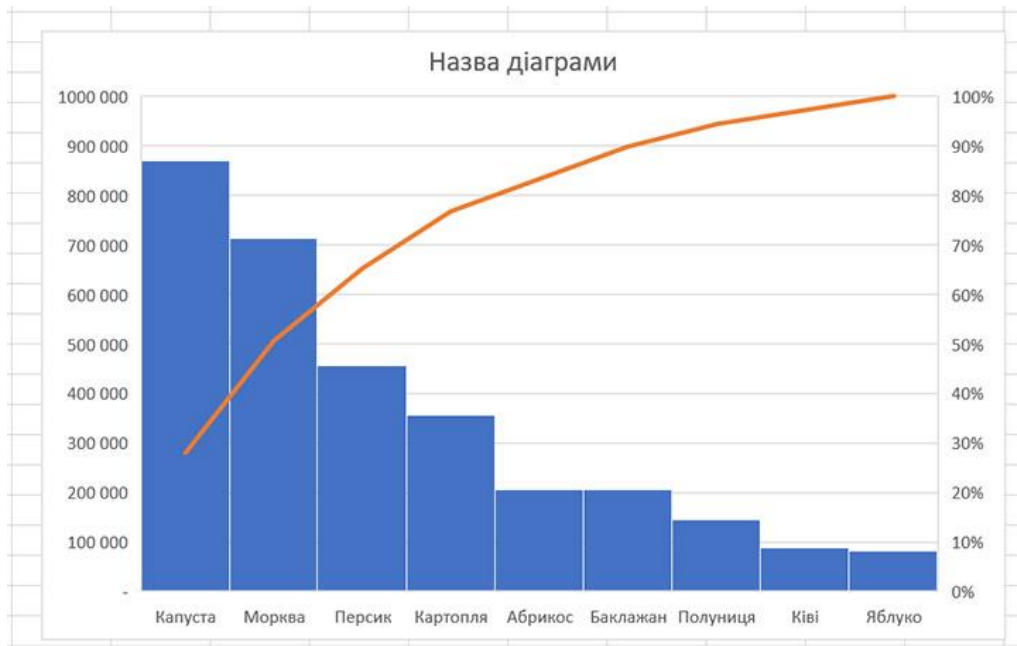
Допустимо, аналізується дохідність груп товарів, розташованих за абеткою.

	A	B	C
1	Товар	Прибуток	
2	Абрикос	206 326	
3	Баклажан	205 949	
4	Капуста	870 960	
5	Картопля	355 526	
6	Ківі	88 173	
7	Морква	713 811	
8	Персик	455 710	
9	Полуниця	144 152	
10	Яблуко	82 485	

Зобразимо на основі цих даних гістограму Парето в Excel 2016. Спочатку активуємо будь-яку клітинку у вихідних даних. Потім на стрічці «Вставлення» через рекомендовані діаграми або безпосередньо виберемо гістограму Парето.



За замовчуванням отримаємо наступний вид діаграми Парето.



Як бачимо процес створення набагато легший і швидший. Нам не довелося робити сортування товарів у таблиці, на діаграмі вони автоматично розташувалися у порядку максимального прибутку. Залишилося підібрати дизайн діаграми через меню «Конструктор» у розділі «Стилі діаграм» або налаштувати окремі її елементи через контекстне меню.



Зручність цієї діаграми у тому, що вона залежить від порядку розташування вихідних даних. Найважливіші елементи завжди будуть на першому місці. Тож якщо дані оновити, то все працюватиме, як треба. Дані для діаграм найкраще зберігати як Таблиця. Це дозволить діаграмі автоматично підлаштовуватись під розмір діапазону.

Контрольні запитання: назвати переваги та недоліки діаграми Парето.
Охарактеризувати запропоновані версії Excel.

ПРАКТИЧНА РОБОТА №4

Тема: Побудова блок – схеми мікроелектронної інформаційної системи керування електричними мережами

Мета: Способи використання особливостей архітектури та функціонування контролерів Arduino (на базі ATmega328) при створенні мікроелектронних інформаційних систем управління електричними мережами

4.1 Структурна схема пристрою

З урахуванням прийнятих попередніх рішень структурна схема пристрою для вимірювання струму та напруги в мережі живлення повинна включати наступні компоненти:

1) Мікроконтролер ARDUINO NANO або будь-який інший подібний мікроконтролер (наприклад, ARDUINO UNO, MICRO, MINI) використовується як основний керуючий елемент пристрою. Він відповідає за збір даних, обробку і керування пристроєм.

2) Датчик струму ACS712 30A GY-712 використовується для вимірювання струму в мережі живлення. Він здатний вимірювати струм до 30A і генерує аналоговий сигнал, який подається на мікроконтролер для подальшої обробки.

3) Ethernet модуль ENC28J60 дозволяє забезпечити з'єднання з мережею Інтернет і передавати дані з пристрою на віддалений сервер або інші пристрої.

4) Модуль живлення TSP-05 220В - 5В 3Вт або будь-яке інше джерело живлення 5В для Arduino використовується для живлення мікроконтролера і інших електронних компонентів пристрою.

5) Трансформатор 220/12В невеликої потужності використовується для зниження напруги з мережі 220В до потрібного рівня напруги для живлення ізолюваних компонентів пристрою.

6) Чотири резистори 0.25W використовуються як датчики напруги.

7) Електролітичний конденсатор на 10 мкФ використовується для стабілізації напруги і забезпечення плавного живлення електронних компонентів.

Загалом, ця структурна схема, яка показана на рис. 4.1 дозволяє вимірювати струм та напругу в мережі живлення за допомогою датчика струму ACS712, обробляти та керувати цими даними за допомогою мікроконтролера ARDUINO та передавати результати через Ethernet модуль ENC28J60. Використовуються також інші компоненти, такі як джерело живлення, трансформатор, резистори та електролітичний конденсатор, для забезпечення правильної роботи пристрою.

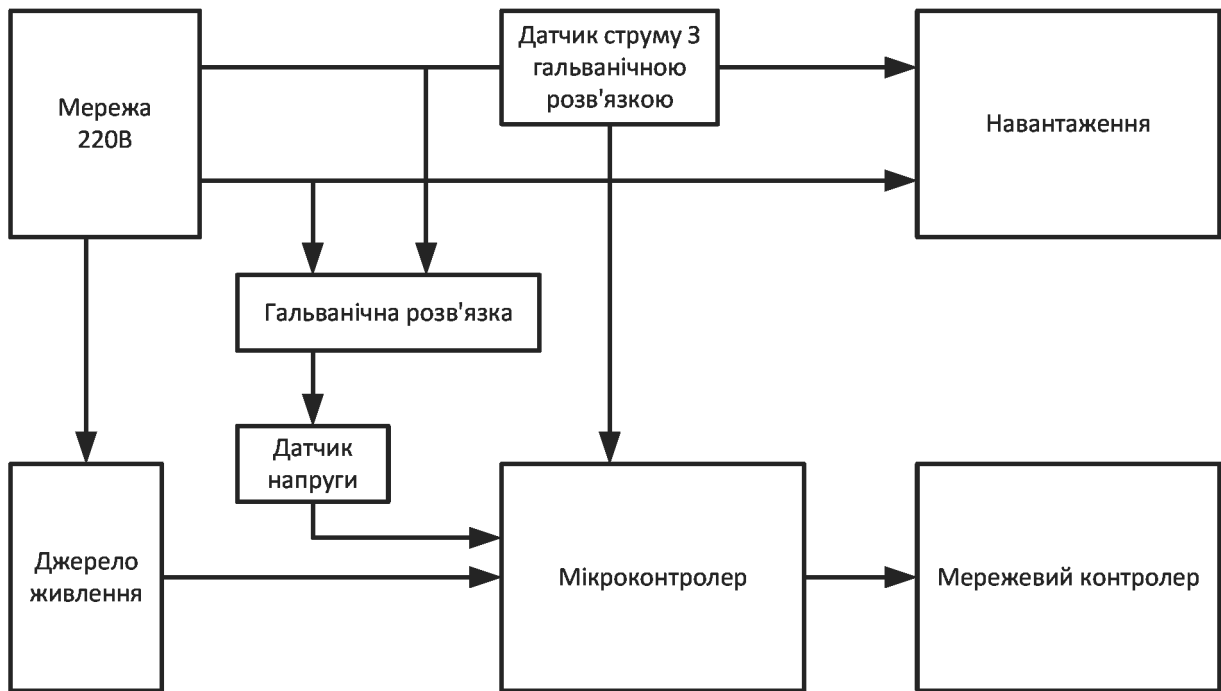


Рисунок 4.1 - Структурна схема пристрою

4.2 Аналіз комплектуючих

Для вимірювання струму в електричній мережі використовується датчик ACS712 30A GY-712, який побудований на однойменній мікросхемі ACS712ELCTR-30A-T фірми Allegro, схема підключення показано на рис 4.2.

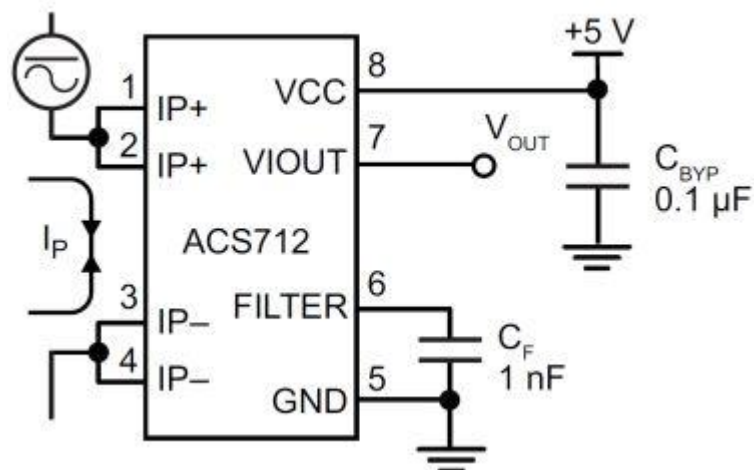


Рисунок 4.2 — Схема підключення датчику струму

Ця мікросхема забезпечує точне вимірювання постійного та змінного електричного струму за рахунок використання ефекту Холла і забезпечує гальванічну розв'язку високовольтної та вимірювальної частини кола. Нижче показано важливу схему включення мікросхеми.

Вимірюваний струм, I_p , протікає від контактів 1, 2 до контактів 3, 4. На виході мікросхеми (контакт 7) при $I_p=0$, напруга дорівнює половині напруги живлення мікросхеми $V_{out} = 5/2 = 2.5\text{В}$. При $I_p=+30\text{А}$ (струм протікає у напрямку від контактів 1, 2 до контактів 3, 4) на виході мікросхеми (контакт 7)

буде напруга $V_{out} = 2.5 + 30 \cdot 0.066 = 4.48\text{В}$. При $I_p = -30\text{А}$ (струм протікає у напрямку від контактів 3,4 до контактів 1,4) на виході мікросхеми (контакт 7) буде напруга $V_{out} = 2.5 - 30 \cdot 0.066 = 0.52\text{В}$.

Таким чином, при протіканні змінного струму в контрольованій електричній мережі, на аналоговому вході А0 мікроконтролера на базі ARDUINO NANO буде напруга, яка змінюється за синусоїдою з частотою, що дорівнює частоті в електричній мережі. Амплітуда напруги на вході А0 буде пропорційна електричному струму (лінія А0 на графіку внизу).

Для вимірювання напруги в електричній мережі використовується звичайний трансформатор 220/12В, який підключений через чотири резистори та конденсатор до аналогового входу А1 мікроконтролера на базі ARDUINO NANO згідно схеми на рис. 4.3. Ці чотири резистори та конденсатор забезпечують зменшення амплітуди коливань напруги зі зміщенням до напруги, рівному половині напруги живлення ($5/2=2.5\text{В}$) на вході мікроконтролера. У результаті на вході А1 при напрузі в електричній мережі рівній нулю буде напруга 2.5В. У точці максимуму амплітуди напруги 220В, напруга на А1 складатиме приблизно 3.6В. У точці мінімуму амплітуди напруги, напруга на А1 буде приблизно 1.44В. Цю напругу можна підігнати до потрібного діапазону шляхом підбору пари резисторів 100кОм та 5.6кОм.

У результаті на вході А1 має бути синусоїда, як показано на графіку рис 4.4.

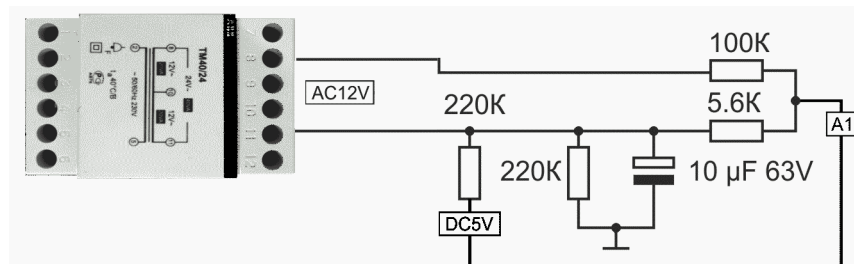


Рисунок 4.3 — Схема вимірювання змінної напруги

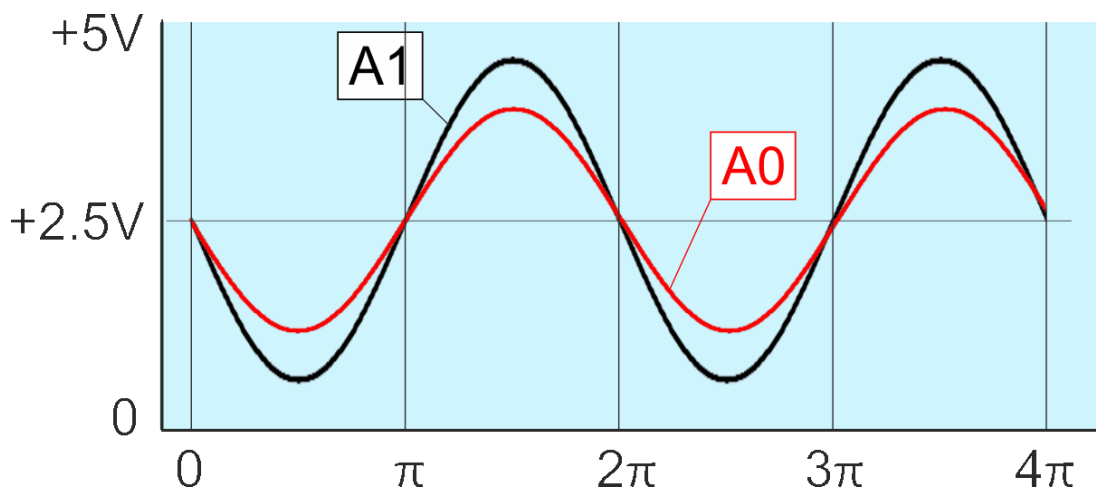


Рисунок 4.4 — Сигнал на вході мікроконтролера

У колах змінного синусоїдального струму, внаслідок постійної зміни значення напруги та струму, потужність не можна обчислити шляхом

простого множення напруги на струм. Тому виділяють відразу три види електричної потужності: активну, реактивну та повну.

Активна потужність у колах синусоїдального струму

Одиниця виміру - ват (позначення: Вт; міжнародне позначення: W).

$$P=U*I*\cos \varphi$$

де P - активна потужність, Вт;

U - середньоквадратична напруга, В;

I - середньоквадратичний струм, А;

φ - кут зсуву фаз напруги та струму, град.

Активна потужність визначає частину електричної енергії, яка використовується безпосередньо на виконання корисної роботи.

Реактивна потужність у колах синусоїдального струму

Одиниця виміру - вольт-ампер реактивний (позначення: вар; міжнародне позначення: var)

$$Q = U*I*\sin \varphi$$

де Q - реактивна потужність, вар;

U - середньоквадратична напруга, В;

I - середньоквадратичний струм, А;

φ - кут зсуву фаз напруги та струму, град.

Реактивна потужність визначає ту частину електричної енергії, яка марно витрачається в електричних мережах.

Повна потужність у колах синусоїдального струму

Одиниця повної електричної потужності - вольт-ампер (позначення: ВА; міжнародне позначення: VA)

$$S= \sqrt{P^2 + Q^2}$$

де S - повна потужність, ВА;

P - активна потужність, Вт;

Q - реактивна потужність, вар;

Повна потужність відповідає всій енергії, яка витрачається в електричних мережах.

4.2 Алгоритм обчислення та код проекту

Через аналогові входи A0 і A1 постійно зчитуємо поточні значення напруги та струму, підносимо їх у квадрат і підсумовуємо. З періодичністю в одну секунду обчислюємо середньоквадратичне значення для напруги та струму. Добуток середньоквадратичного значення напруги і струму дає величину повної потужності.

При розробці пристрою також буде потрібна бібліотека UIPEthernet. В IDE Arduino її можна встановити через меню "Скетч"->"Підключити бібліотеку" - "Керувати бібліотеками" - "Менеджер бібліотек" - UIPEthernet. Також можна завантажити цю бібліотеку з GitHub за адресою <https://github.com/UIPEthernet/UIPEthernet>.

Після завершення проектування лістинг отриманої програми наведено нижче.

```
#include <UIPEthernet.h>
#define DEMO true //встановить false для реального виміру
const char page1[] PROGMEM =
"<!DOCTYPE html>\n"
"<html>\n"
"<meta charset='UTF-8'>\n"
"<title>WEB MONITOR</title>\n"
"<link href='http://liccontrol.com/web/gauge.ico' rel='icon' type='image/x-
icon'/>\n"
"<link rel='stylesheet' href='http://liccontrol.com/web/style.css'
type='text/css'>\n"
"<script src='http://code.jquery.com/jquery-3.1.1.min.js'></script>\n"
"<script src='http://code.highcharts.com/highcharts.js'></script>\n"
"</head>\n"
;
const char page2[] PROGMEM =
"<body>\n"
"<table id='MainTab' border='0' cellspacing='0' cellpadding='0'>\n"
"<tbody><tr ><td><div id='trend'></div></td></tr></tbody>\n"
"</table>\n"
"<script src='http://liccontrol.com/web/trend.js'></script>\n"
"</body></html>\n"
;
const char page3[] PROGMEM =
"HTTP/1.1 200 OK\n"
"Content-Type: text/html\n"
"Pragma: no-cache\n"
"Connection: close\r\n\r\n"
;

int IPin = A1; //Аналоговий вхід для вимірювання струму
int VPin = A0; //Аналоговий вхід для вимірювання напруги
//поточне та сумарне значення струму
double I=0.0, I1 = 0.0, I2 = 0.0;
//поточне та сумарне значення напруги
double V=0.0, V1 = 0.0, V2 = 0.0;
float S = 0.0; //повна потужність
float P = 0.0; //активна потужність
float Q = 0.0; //Реактивна потужність
float P1 = 0.0; //миттєва потужність
float kI = 0.07398; //Коефіцієнт вимірювання струму
float kV = 0.97656; //Коефіцієнт вимірювання напруги
//Унікальна mac-адреса. Якщо буде повторення в мережі,
```

```

//Ethernet - модуль працювати не буде!
byte mac[] = {0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED};
//Встановить відповідну для вас адресу
IPAddress ip(192, 168, 1, 10);
EthernetServer server(80);
String result="";
unsigned long time1 = 0, count=0, count1=0;
unsigned long t1 = 0;
boolean Restart = true;

//Перезапуск мікросхеми ENC28J60
void resetENC28J60() {
digitalWrite(3, LOW);
delay(200);
digitalWrite(3, HIGH);
delay(100);
}

//Процедура вимірювання струму та напруги
void MeasureData() {
unsigned long dt = (unsigned long)(millis() - t1);
//миттєве значення струму, А
I1 = kI*(analogRead(IPin) - 512);
if (DEMO) I1 = sin(1.5*(millis()-time1)/60000.0)*10.0+20.0;
I2 += pow(I1,2);
//миттєве значення напруги, В
V1 = kV*(analogRead(VPin) - 512);
if (DEMO) V1 = sin((millis()-time1)/60000.0)*50.0+200.0;
V2 += pow(V1,2);
P1 += I1 * V1;
if ((dt>1000)and(count>0)) {
count1++;
I = sqrt(I2/count); //Середньоквадратичне значення струму, А
V = sqrt(V2/count); //Середньоквадратичне значення напруги,
S = I * V; //повна потужність, ВА
P = P1/count; //активна потужність, Вт
if (DEMO) P=P*(0.8 + 0.2*sin(0.5*(millis()-time1)/60000.0));
float _Q = pow(S,2)-pow(P,2);
if (_Q>0.0) Q = sqrt(_Q); //реактивна потужність, вар
else Q = 0.0;
P1 = 0.0;
V2 = 0.0;
I2 = 0.0;
count = 0;
t1 = millis();
}
}

```



```

}
count++;
}

//Процедура надсилання до мережі даних, які зберігаються в PROGMEM
void printProgStr (EthernetClient client, const char * str){
char c;
if (!str) return;
while ((c = pgm_read_byte(str++))) client.print (c);
}

//Формуємо JSON рядок даних для відправки її через мережу до браузера
String DataToBrowser() {
int t = int((millis()-time1)/1000);
String s = String(t)+", "+String(V,1)+"", "+
String(I,1)+"", "+String(S/1000.0,2)+"", "+
String(P/1000.0,2);
return s;
}

//Початкові установки та ініціалізація
void setup() {
Serial.begin(9600);
resetENC28J60();
Ethernet.begin(mac, ip);
server.begin();
Serial.print("START SERVER AT:");
Serial.println(Ethernet.localIP());
}

//Головний цикл виконання програми
void loop() {
MeasureData();
EthernetClient client = server.available();
if (client) {
result="";
while (client.connected()) {
if (client.available()) {
char c = client.read();
result+=c;
if (c == '\n') {
int y=result.indexOf("dat");
if ((y>0) and (!Restart)) {
printProgStr(client, page3);
String s = DataToBrowser();

```

```

client.println(s);
} else {
printProgStr(client, page1);
printProgStr(client, page2);
Restart = false;
time1 = millis();
}
break;
}
}
}
delay(1);
client.stop();
}
}

```

Підключіть модуль ENC28J60 Ethernet до мережі. Після завантаження програми в контролер, у пошуковому рядку браузера наберіть <http://192.168.1.10/> та натисніть Enter.

Комп'ютер повинен бути обов'язково підключений до Інтернету. В цьому випадку з'явиться сторінка з демонстраційними графіками. Після цього можна підключити датчик струму - ACS712 30A GY-712, трансформатор та підібрати коефіцієнти kI,kV щоб отримати правильні дані.

Потрібно встановити `#define DEMO false` коли треба вимірювати реальні дані. Це параметр потрібен для відладки.

Контрольні питання

1. Як використовуються мікроконтролери для мікроелектронних інформаційних систем управління електричними мережами?
2. Які комплектуючі потрібні для реалізації типового проекту?
3. Яка платформа використовувалась для вирішення поставлених питань?

ПРАКТИЧНА РОБОТА №5

Тема: Рішення логічних задач засобами алгебри логіки

Мета: Ознайомитись з функціями по роботі з матрицям та їх застосуванням при рішенні логічних задач

5.1 Теоретичні відомості

Загальні відомості про матриці

Значна частина математичних моделей різних об'єктів і процесів записується в досить простій і компактній матричній формі. Зокрема, при рішенні лінійних рівнянь ми маємо справу з матрицями і арифметичними діями з ними.

Матрицею розмірністю $m \times n$ називається прямокутна таблиця чисел, що містить m рядків і n стовпців. Матриці позначаються прописними (заголовними) буквами латинського алфавіту. Числа, що становлять матрицю, називаються елементами матриці і позначаються рядковими буквами з подвійною індексацією: a_{ij} , де i - номер рядка, а j - номер стовпця. Наприклад, матриця A розміром $m \times n$ може бути представлена у виді:

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{pmatrix} = (a_{ij})$$

де $i=1, \dots, m; j=1, \dots, n$.

Дві матриці A і B одного розміру називаються рівними, якщо вони співпадають поелементно, тобто $a_{ij}=b_{ij}$ для будь-яких $i=1, 2, \dots, m; j=1, 2, \dots, n$.

Матриця, що складається з одного рядка, називається матрицею (вектором) - рядком:

$$A = (a_{11}, a_{12}, \dots, a_{1n}),$$

а з одного стовпця - матрицею (вектором) - стовпцем:

$$B = \begin{pmatrix} b_{11} \\ b_{21} \\ \dots \\ b_{m1} \end{pmatrix}.$$

Якщо число рядків матриці дорівнює числу стовпців і рівне n , то таку матрицю називають квадратною n -го порядку. Наприклад, квадратна матриця 2-го порядку :

$$A = \begin{pmatrix} 1 & 3 \\ 4 & 5 \end{pmatrix}.$$

Якщо у елементу матриці a_{ij} номер стовпця дорівнює номеру рядка ($i=j$), то такий елемент називається діагональним. Діагональні елементи утворюють головну діагональ матриці.

Квадратна матриця з рівними нулю усіх недіагональних елементів називається діагональною.

Квадратна матриця називається одиничною, якщо вона діагональна, і усі діагональні елементи дорівнюють одиниці. Одинична матриця має вигляд:

$$E = \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 1 \end{pmatrix}.$$

Операції з матрицями

Як і над числами, над матрицями можна проводити ряд операцій, причому у випадку з матрицями деякі з операцій є специфічними.

Транспонування

Транспонованою називається матриця (A^T) , у якій стовпці початкової матриці (A) замінюються рядками з відповідними номерами.

Для позначення транспонованої матриці іноді використовують символ $'$ (A') . Транспонуванням називається операція переходу від початкової матриці (A) до транспонованої (A^T) .

$$A = \begin{pmatrix} 3 & -7 & 11 \\ 18 & 19 & 39 \\ -5 & 91 & 87 \end{pmatrix}; \quad A^T = \begin{pmatrix} 3 & 18 & -5 \\ -7 & 19 & 91 \\ 11 & 39 & 87 \end{pmatrix}.$$

З визначення транспонованої матриці виходить, що якщо початкова матриця A має розмір $m \times n$, то транспонована матриця A^T має розмір $n \times m$.

Для здійснення транспонування в **MS Excel** використовується функція “**ТРАНСП**”, яка дозволяє поміняти орієнтацію масиву на робочому листі з вертикальною на горизонтальну і навпаки.

Функція має вигляд “**ТРАНСП (масив)**”. Тут **(масив)** - це масив, що транспонується, або діапазон осередків на робочому листі. Транспонування масиву полягає в тому, що перший рядок масиву стає першим стовпцем нового масиву, другий рядок масиву стає другим стовпцем нового масиву і т. д.

Обчислення визначника матриці

Важливою характеристикою квадратних матриць є їх визначник. Визначник матриці - це число, що обчислюється на основі значень елементів масиву. Визначник матриці A позначається як $|A|$ або Δ .

Визначником матриці першого порядку A , або визначником першого порядку, називається елемент a_{11} .

$$\Delta_1 = |A| = a_{11}$$

Визначником матриці другого порядку A , або визначником другого порядку, називається число, яке обчислюється за формулою:

$$\Delta_2 = |A| = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} = a_{11} \cdot a_{22} - a_{12} \cdot a_{21}.$$

У **MS Excel** для обчислення визначника квадратної матриці використовується функція “**МОПРЕД**”. Функція має вигляд

“**МОПРЕД(масив)**”. Тут (*масив*) - це числовий масив, в якому зберігається матриця з рівною кількістю рядків і стовпців. При цьому масив може бути заданий як інтервал осередків, наприклад, A1:C3; чи як масив констант, наприклад, {1;2;3;4;5;6;7;8;9}.

Знаходження зворотної матриці

Для кожного числа $a \neq 0$ існує зворотне число a^{-1} , і для квадратних матриць вводиться аналогічне поняття. Зворотні матриці зазвичай використовуються для вирішення систем рівнянь з декількома невідомими.

Матриця A^{-1} називається зворотною по відношенню до квадратної матриці A , якщо при множенні цієї матриці на дану як ліворуч, так і праворуч виходить одинична матриця:

$$A \times A^{-1} = A^{-1} \times A = E.$$

Як випливає з визначення, зворотна матриця є квадратною того ж порядку, що і початкова матриця.

Існують спеціальні досить складні алгоритми для ручного обчислення зворотних матриць. У **MS Excel** для знаходження зворотної матриці використовується функція “**МОБР**”, яка обчислює зворотну матрицю для матриці, що зберігається в таблиці у вигляді масиву.

Складання і віднімання матриць

Складати (віднімати) можна матриці одного розміру. Сумою матриць $A = (a_{ij})$ і $B = (b_{ij})$ розміру $m \times n$ називається *матриця* $C = A + B$, елементи якої $c_{ij} = a_{ij} + b_{ij}$ для $i = 1, 2, \dots, m; j = 1, 2, \dots, n$ (тобто матриця складається поелементно). Наприклад, якщо:

$$A = \begin{pmatrix} 1 & 2 & 7 \\ 9 & -1 & 13 \end{pmatrix}; \quad B = \begin{pmatrix} 0 & -4 & -3 \\ 5 & 19 & 31 \end{pmatrix},$$

$$\text{то } C = \begin{pmatrix} 1+0 & 2-4 & 7-3 \\ 9+5 & -1+19 & 13+31 \end{pmatrix} = \begin{pmatrix} 1 & -2 & 4 \\ 14 & 18 & 44 \end{pmatrix}.$$

Аналогічно визначають різницю двох матриць $C = A - B$.

У **MS Excel** для виконання операцій підсумовування і віднімання матриць можуть бути використані формули, що вводяться у відповідні осередки.

Множення матриці на число

Добутком матриці A на число k називається матриця $B = kA$, елементи якої $b_{ij} = ka_{ij}$ для $i = 1, 2, \dots, m; j = 1, 2, \dots, n$. Інакше кажучи, при множенні матриці на постійну величину кожен елемент цієї матриці множиться на цю величину:

$$k \cdot A_{ij} = (k \cdot a_{ij}).$$

Множення матриць

Добуток матриць визначений, якщо число стовпців першої матриці дорівнює числу рядків другою.

Нехай $A = (a_{ij}) m \times n$, $B = (b_{ij}) n \times p$, тоді розмірність добутку $A \times B$

дорівнює $m \times p$. При цьому матриця C називається добутком матриць A і B , якщо кожен її елемент c_{ij} дорівнює сумі добутків елементів i -го рядка матриці A на відповідні елементи j -го стовпця матриці B :

$$c_{ij} = a_{i1}b_{1j} + a_{i2}b_{2j} + \dots + a_{ip}b_{pj} = \sum_{k=1}^p a_{ik}b_{kj}, \quad i=1,2,\dots,m; \quad j=1,2,\dots,n.$$

Таким чином, перемножування матриць здійснюється за наступним правилом:

$$C = A \cdot B = \begin{pmatrix} 1\text{стр} \cdot 1\text{стб} & 1\text{стр} \cdot 2\text{стб} & \dots & 1\text{стр} \cdot p\text{стб} \\ 2\text{стр} \cdot 1\text{стб} & 2\text{стр} \cdot 2\text{стб} & \dots & 2\text{стр} \cdot p\text{стб} \\ \dots & \dots & \dots & \dots \\ m\text{стр} \cdot 1\text{стб} & m\text{стр} \cdot 2\text{стб} & \dots & m\text{стр} \cdot p\text{стб} \end{pmatrix}.$$

Наприклад

$$C = A \cdot B = \begin{pmatrix} 1 & 3 & 4 & 2 \\ 3 & 2 & 0 & -1 \\ 0 & 1 & -1 & 2 \end{pmatrix} \cdot \begin{pmatrix} 1 & 3 \\ 2 & 2 \\ 10 & 0 \\ 12 & -1 \end{pmatrix} = \begin{pmatrix} 1 \cdot 1 + 3 \cdot 2 + 4 \cdot 10 + 2 \cdot 12 & 1 \cdot 3 + 3 \cdot 2 + 4 \cdot 0 + 2 \cdot (-1) \\ 3 \cdot 1 + 2 \cdot 2 + 0 \cdot 10 - 1 \cdot 12 & 3 \cdot 3 + 2 \cdot 2 + 0 \cdot 0 - 1 \cdot (-1) \\ 0 \cdot 1 + 1 \cdot 2 - 1 \cdot 10 + 2 \cdot 12 & 0 \cdot 3 + 1 \cdot 2 - 1 \cdot 0 + 2 \cdot (-1) \end{pmatrix} = \begin{pmatrix} 71 & 7 \\ -5 & 14 \\ 16 & -1 \end{pmatrix}.$$

Багато властивостей, властивих операціям над числами, справедливі і для операцій множення матриць.

В алгебрі матриць немає дії ділення. Вираз A/B не має сенсу. Його замінюють два різні вирази $B^{-1} \times A$ і $A \times B^{-1}$, якщо існує B^{-1} .

Для квадратних матриць можлива операція піднесення до степеня. Після визначення, вважають, що $A^0 = E$ і $A^1 = A$. Цілим позитивним ступенем $A^m (m > 1)$ квадратною матрицею A називається добутком m матриць, рівних A , тобто, :

$$A^m = A \cdot A \cdot \dots \cdot A \quad m \text{ разів.}$$

Для знаходження добутку двох матриць в **Excel** використовується функція “**МУМНОЖ**”, яка обчислює добуток матриць.

Рішення системи рівнянь у MS Excel

Рішення системи рівнянь за допомогою знаходження зворотної матриці. Нехай дана лінійна система рівнянь.

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = b_2 \\ \dots \\ a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n = b_n \end{cases}$$

Цю систему рівнянь можна представити в матричній формі:

$$A \times X = B,$$

де:

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix}$$

Матриця невідомих обчислюється за формулою $X = A^{-1}B$.

5.2 Порядок виконання роботи

При виконанні лабораторної роботи дотримуйтесь рекомендованого порядку:

- ознайомтесь з матеріалом , який наведено у підрозділі 3.1;
- отримайте від викладача варіант завдання;
- створіть новий проект в Microsoft Office Excel 2003;
- виконайте завдання згідно варіанту та прикладу, що наведений у підрозділі 3.3. Варіанти завдань наведено у підрозділі 3.5;
- оформіть звіт.

5.3 Приклад виконання завдання

Завдання.Рішити систему рівнянь, а також провести транспонування і обчислення визначника матриці.

$$\begin{cases} 2x_1 + 3x_2 + 4x_3 + 5x_4 = 5 \\ 4x_1 + 5x_2 + 2x_3 + 7x_4 = 2 \\ 3x_1 + 5x_2 + 6x_3 + 4x_4 = 7 \\ 2x_1 + 6x_2 + 7x_3 + 5x_4 = 2 \end{cases}$$

1) Вводимо значення елементів матриць A і B рівняння у комірці рисунок 5.1.

2) Виділяємо блок комірок під зворотною матрицю.

3) Викликаємо діалогове вікно “*Мастер функций*” і в робочому полі “*Категория*” обираємо “*Математические*”, а в робочому полі “*Функция*” – ім'я функції “*МОБР*”. Після цього клацаємо по кнопці **ОК**.

4) Вводимо діапазон початкової матриці у робоче поле (“*Массив*”) і натискаємо кнопку **ОК**.

	A	B	C	D	E	F	G	H	I
1									
2	A=	2	3	4	5		B=	5	
3		4	5	2	7			2	
4		3	5	6	4			7	
5		2	6	7	5			2	
6									

Рисунок 5.1 – Функція в табличному вигляді

5) Потім натискаємо поєднання клавіш **Ctrl+Shift+Enter**. Зворотна матриця готова.

6) Перемножуємо зворотну матрицю A^{-1} на матрицю B за допомогою матричної функції “**МУМНОЖ**” (порядок множення важливий - першою повинна йти матриця A^{-1} , а другою - B). Отриманий вектор-стовпець X є результатом рішення системи рівнянь.

7) Транспонування матриці A . Для цього виділяємо блок комірок під транспоновану матрицю.

8) Запускаємо “**Мастер функцій**” і в робочому полі “**Категория**” вибираємо “**Ссылки и массивы**”, а в робочому полі “**Функция**” - ім'я функції **ТРАНСП**, після цього клацаємо на кнопці ОК.

9) Після появи діалогового вікна **ТРАНСП** виділяємо матрицю і тиснемо **Enter**. Після цього натискаємо поєднання клавіш **Ctrl+Shift+Enter**.

10) Обчислимо визначник матриці A^{-1} . Для цього вибираємо комірку під визначник Δ . Запускаємо “**Мастер функцій**” і в робочому полі “**Категория**” обираємо “**Математические**”, а в робочому полі “**Функция**” - ім'я функції **МОПРЕД**.

11) Після появи діалогового вікна **МОПРЕД** виділяємо матрицю і тиснемо **Enter**. Готовий приклад показана на рис. 5.2 .

	A	B	C	D	E	F	G	H
1								
2	A=	2	3	4	5		B=	5
3		4	5	2	7			2
4		3	5	6	4			7
5		2	6	7	5			2
6								
7	A ⁻¹ =	0,010417	0,03125	0,6875	-0,60417		X=	3,71875
8		-0,59375	0,21875	-0,1875	0,4375			-2,96875
9		0,260417	-0,21875	0,1875	-0,10417			1,96875
10		0,34375	0,03125	-0,3125	0,0625			-0,28125
11								
12	A' =	2	4	3	2		Δ =	96
13		3	5	5	6			
14		4	2	6	7			
15		5	7	4	5			

Рисунок 5.2 – Зображення готового завдання.

5.4 Зміст звіту

Звіт з лабораторної роботи повинен містити:

- мету роботи;
- завдання на виконання роботи;
- результати виконаної роботи ;
- висновки.

5.5 Завдання на практичну роботу

Вирішити систему рівнянь, провести транспонування і обчислення визначника матриці. Варіанти завдань наведено в табл.5.1.

Таблиця 5.1 –Варіанти завдань:

Вар-т	Завдання	Вар-т	Завдання
1	2	3	4
1	$\begin{cases} 27x_1 + 3,3x_2 + 1,3x_3 = 21 \\ 3,5x_1 - 17x_2 + 2,8x_3 = 17 \\ 4,1x_1 + 5,8x_2 - 17x_3 = 8 \end{cases}$	2	$\begin{cases} 17x_1 + 2,8x_2 + 1,9x_3 = 7 \\ 2,1x_1 + 34x_2 + 1,8x_3 = 11 \\ 4,2x_1 - 1,7x_2 + 13x_3 = 28 \end{cases}$
3	$\begin{cases} 31x_1 + 2,8x_2 + 1,9x_3 = 2 \\ 1,9x_1 + 31x_2 + 2,1x_3 = 21 \\ 7,5x_1 + 3,8x_2 + 48x_3 = 56 \end{cases}$	4	$\begin{cases} 91x_1 + 5,6x_2 + 7,8x_3 = 98 \\ 3,8x_1 + 51x_2 + 2,8x_3 = 67 \\ 4,1x_1 + 5,7x_2 + 12x_3 = 58 \end{cases}$
5	$\begin{cases} 33x_1 + 2,1x_2 + 2,8x_3 = 8 \\ 4,1x_1 + 37x_2 + 4,8x_3 = 57 \\ 2,7x_1 + 1,8x_2 + 11x_3 = 32 \end{cases}$	6	$\begin{cases} 76x_1 + 5,6x_2 + 4,7x_3 = 101 \\ 3,8x_1 + 41x_2 + 2,7x_3 = 97 \\ 2,9x_1 + 2,1x_2 + 38x_3 = 78 \end{cases}$
7	$\begin{cases} 32x_1 - 2,5x_2 + 3,7x_3 = 65 \\ 0,5x_1 + 34x_2 + 1,7x_3 = -2,4 \\ 1,6x_1 + 2,3x_2 - 15x_3 = 43 \end{cases}$	8	$\begin{cases} 54x_1 - 2,3x_2 + 3,4x_3 = -35 \\ 3,4x_1 + 17x_2 + 2,3x_3 = 27 \\ 3,4x_1 + 2,4x_2 + 74x_3 = 19 \end{cases}$
9	$\begin{cases} 36x_1 + 1,8x_2 - 4,7x_3 = 38 \\ 2,7x_1 - 36x_2 + 1,9x_3 = 4 \\ 1,5x_1 + 4,5x_2 + 33x_3 = -16 \end{cases}$	10	$\begin{cases} 56x_1 + 2,7x_2 - 1,7x_3 = 19 \\ 3,4x_1 - 36x_2 - 6,7x_3 = -24 \\ 0,8x_1 + 1,3x_2 + 37x_3 = 12 \end{cases}$
11	$\begin{cases} 27x_1 + 0,9x_2 - 1,5x_3 = 35 \\ 4,5x_1 - 28x_2 + 6,7x_3 = 26 \\ 5,1x_1 + 3,7x_2 - 14x_3 = -14 \end{cases}$	12	$\begin{cases} 45x_1 - 3,5x_2 + 7,4x_3 = 25 \\ 3,1x_1 - 6x_2 - 2,3x_3 = -15 \\ 0,8x_1 + 7,4x_2 - 5x_3 = 64 \end{cases}$
13	$\begin{cases} 38x_1 + 6,7x_2 - 1,2x_3 = 52 \\ 6,4x_1 + 13x_2 - 2,7x_3 = 38 \\ 2,4x_1 - 4,5x_2 + 35x_3 = -6 \end{cases}$	14	$\begin{cases} 54x_1 - 6,2x_2 - 0,5x_3 = 5,2 \\ 3,4x_1 + 23x_2 + 0,8x_3 = -8 \\ 2,4x_1 - 1,1x_2 + 38x_3 = 18 \end{cases}$
15	$\begin{cases} 78x_1 + 5,3x_2 + 4,8x_3 = 18 \\ 3,3x_1 + 11x_2 + 1,8x_3 = 23 \\ 4,5x_1 + 3,3x_2 + 28x_3 = 34 \end{cases}$	16	$\begin{cases} 38x_1 + 4,1x_2 - 2,3x_3 = 48 \\ -2,1x_1 + 39x_2 - 5,8x_3 = 33 \\ 1,8x_1 + 1,1x_2 - 21x_3 = 58 \end{cases}$
17	$\begin{cases} 17x_1 - 2,2x_2 + 30x_3 = 18 \\ 2,1x_1 + 19x_2 - 2,3x_3 = 28 \\ 4,2x_1 + 3,9x_2 - 31x_3 = 51 \end{cases}$	18	$\begin{cases} 28x_1 + 3,8x_2 - 32x_3 = 45 \\ 2,5x_1 - 2,8x_2 + 3,3x_3 = 71 \\ 6,5x_1 - 7,1x_2 + 48x_3 = 63 \end{cases}$
19	$\begin{cases} 33x_1 + 3,7x_2 + 4,2x_3 = 58 \\ 2,7x_1 + 23x_2 - 2,9x_3 = 61 \\ 4,1x_1 + 4,8x_2 - 50x_3 = 70 \end{cases}$	20	$\begin{cases} 71x_1 + 6,8x_2 + 6,1x_3 = 70 \\ 5x_1 + 48x_2 + 5,3x_3 = 61 \\ 8,2x_1 + 7,8x_2 + 71x_3 = 58 \end{cases}$

5.6 Контрольні питання

1. Що таке матриця? Види матриць. Розмірність матриць.
2. Методика рішення системи рівнянь у **MS Excel**.
3. Що таке зворотня матриця? Як її обчислити у **MS Excel**?

ПРАКТИЧНА РОБОТА №6

Тема: Створення базових таблиць в ACCESS

Мета: Побудова інтерактивних звітів

MS Access є додатком Windows. У СУБД Access передбачено багато додаткових сервісних можливостей. Майстри допоможуть створити таблиці, форми або звіти з наявних заготовок. Вирази використовуються в Access, наприклад, для перевірки допустимості введеного значення. Макроси дозволяють автоматизувати багато процесів без програмування, тоді як вбудований в Access мову VBA (Visual Basic for Applications) дає можливість досвідченому користувачеві програмувати складні процедури обробки даних.

6.1. Структура таблиці і типи даних

При розробці структури таблиці, перш за все, необхідно визначити назви полів, з яких вона повинна складатися, типи полів і їх розміри. Кожному полю таблиці присвоюється унікальне ім'я, яке не може містити більше 64 символів. Ім'я бажано робити таким, щоб функція поля дізнавалася по його імені. Далі треба вирішити, дані якого типу будуть міститися в кожному полі. У MS Access можна вибрати будь-які з основних типів даних. Один з цих типів даних повинен бути присвоєний кожному полю. Значення типу поля може бути поставлено лише в режимі конструктора.

В Access існує декілька способів створення порожньої таблиці:

- майстер таблиць;
 - введення даних безпосередньо в порожню таблицю в режимі таблиці;
- визначення всіх параметрів макета таблиці в режимі конструктора.

6.1.2. Об'єкти бази даних

Об'єктами бази даних є:

Таблиці - сукупність записів, де зберігається основна інформація. Форма являє собою спеціальний формат екрану, використовуються для введення даних в таблицю і перегляду однієї записи.

Запит - це інструмент для аналізу, вибору і зміни даних. За допомогою

Access можна створювати кілька видів запитів.

Звіти - це засіб організації даних при виведенні на друк.

З усіх типів об'єктів тільки таблиці призначені для зберігання інформації. Решта використовуються для перегляду, редагування, обробки і аналізу даних - інакше кажучи, для забезпечення ефективного доступу до інформації.

6.1.3. Типи даних

Текстовий - найбільш чисто використовуваний в Access тип даних. Цей тип даних підходить для зберігання адрес, для полів з коротким описом,

для числових даних, які не потребують розрахунків, таких, як телефонні номери та поштові індекси. Довжина - 255 символів.

Поле Мемо - призначений для полів, довжина яких перевищує 255 символів. Приклад: довге поле опису. Поле Мемо може зберігати до 65 535 символів, що приблизно дорівнює 32 сторінок тексту.

Числовий. Дані, що використовуються для математичних обчислень, за винятком фінансових розрахунків (для них слід використовувати тип «Грошовий»).

Дата час. Значення дат і часу. Зберігає 8 байтів. Можна вводити дати з 1 січня 100 року по 31 грудня 9999 року. Access пропонує кілька різних форматів дат.

Грошовий. Використовується для грошових значень і для запобігання округлення під час обчислень, для виконання обчислень над полем, яке містить числа, в лівій частині яких не більше 15 знаків, а праворуч від коми не більше чотирьох знаків.

Лічильник. Автоматична вставка унікальних послідовних (збільшуються на 1) або випадкових чисел при додаванні запису з використанням цього типу даних або, вибравши відповідний пункт у властивості Нове значення цього поля. Якщо видалити одну з послідовних записів, цей тип поля не запам'ятав і не перенумеруйте віддалене значення. Це значення буде просто відсутні.

Логічний (Так / ні). Дані, що приймають тільки одне з двох можливих значень, таких як «Так / Ні», «Істина / Брехня», «On / Off». Значення Null не допускаються.

Поле об'єкта OLE. Об'єкти OLE (такі як документи Microsoft Word, електронні таблиці Microsoft Excel, малюнки, звукозапис або інші дані в довічнім форматі), створені в інших програмах, що використовують протокол OLE.

Гіперпосилання. Гіперпосилання може мати вигляд шляху UNC або адреси URL.

Майстер підстановок. Створює поле, що дозволяє вибрати значення з іншої таблиці або зі списку значень, використовуючи поле зі списком.

6.1.4. Несумісні дані

Необхідно дотримуватися для одних найменувань полів даних однаковий тип даних.

Для створення нової бази даних: завантажте MS Access, у вікні виберіть пункт «**Нова база даних**»;

Задайте ім'я вашої бази. За замовчуванням MS Access пропонує вам ім'я бази db1 (Database1.accdb), а тип файлу - Бази даних Access. Ім'я задайте «**Стоматологія**».

Порядок виконання:

6.2 У вікні бази даних виберіть вкладку **Створення**, а потім клацніть по кнопці <**Конструктор таблиць**> (рис. 6.1).

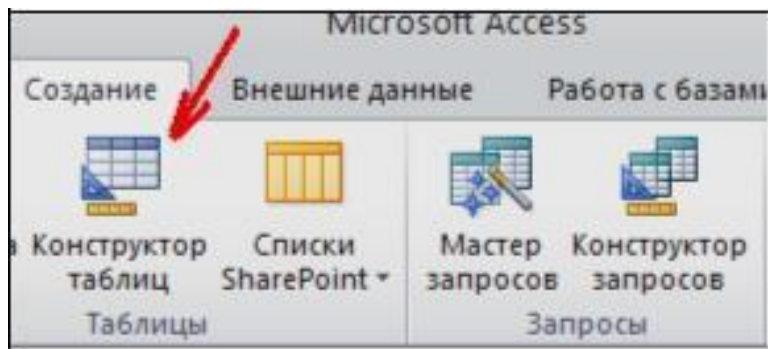


Рис 6.1. Вибір способу створення таблиці

6.3. Для визначення полів таблиці: введіть в рядок стовпця Ім'я поля ім'я першого поля – «**ID лікаря**».

У рядку стовпця «**Тип даних**» натисніть на кнопку списку і виберіть тип даних Числовий. Поля вкладки Загальні залиште такими, як пропонує Access (див. рис. 6.2).

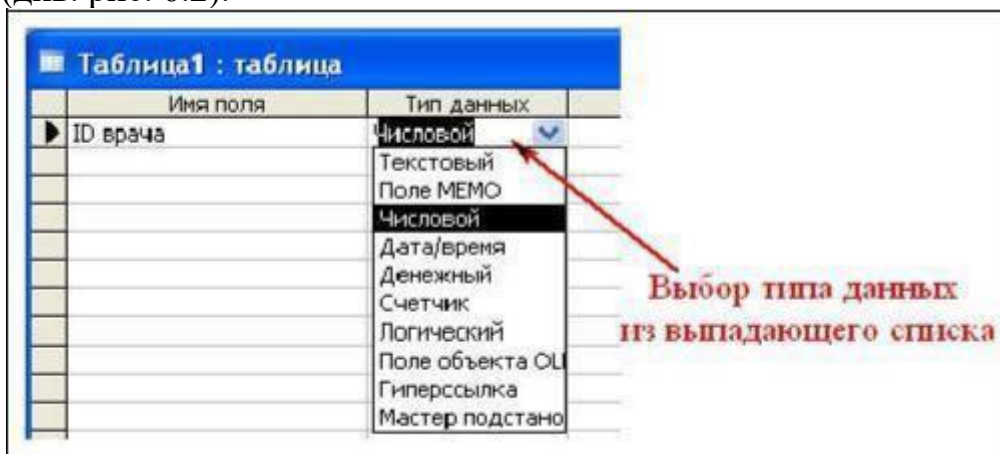


Рис 6.2. Вибір типу даних

Для визначення всіх інших полів таблиці бази даних «Лікарі»

виконайте дії, аналогічні зазначеним вище відповідно до табл. 6.1.

Табл.6.1 – Властивості таблиці Лікарі

Ім'я поля	Тип даних	Розмір поля (вкладка Загальні)
ID лікаря	Числовой	
П.І.Б.	Текстовый	20
Посада	Текстовый	20

№ кабінету	Числовий	Довге ціле (по замовчуванню)
Термін прийому	Текстовий	12
Вид наданих послуг	Текстовий	50 (Максимальне значення)

Примітка : (дані вибрані мовою програмного продукту)

6.4. Введіть умови для поля «№ кабінета».

Нехай за умовою в поліклініці 2 поверху і значення номера кабінету знаходиться в межах від 100 до 130 і від 200 до 230. У рядку «Умова на значення» перейдіть в режим Будівника виразів (рис. 4) і введіть умову:>
=

100 And <= 130 Or >= 200 And <= 230 (див.. рис. 6.3).

У рядку «Повідомлення про помилку» введіть: «Такого номера кабінету немає».



Рис 6.3. Кнопка
переходу

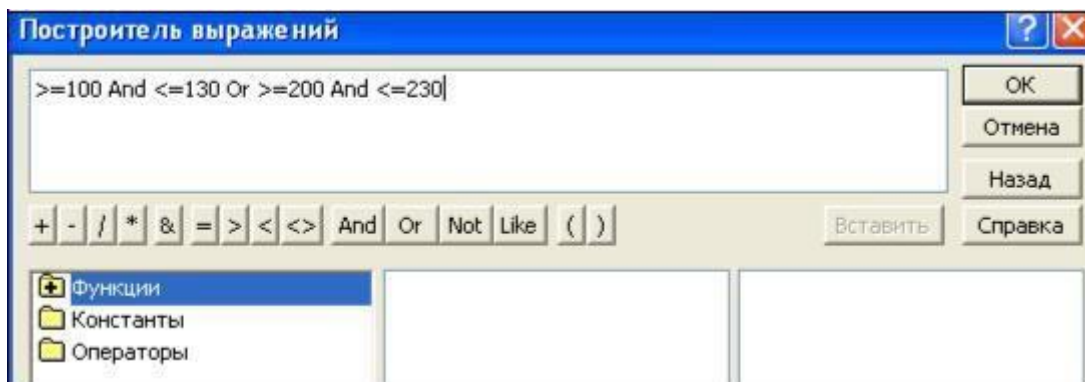


Рис.6.4. Вигляд «Будувача виразів»

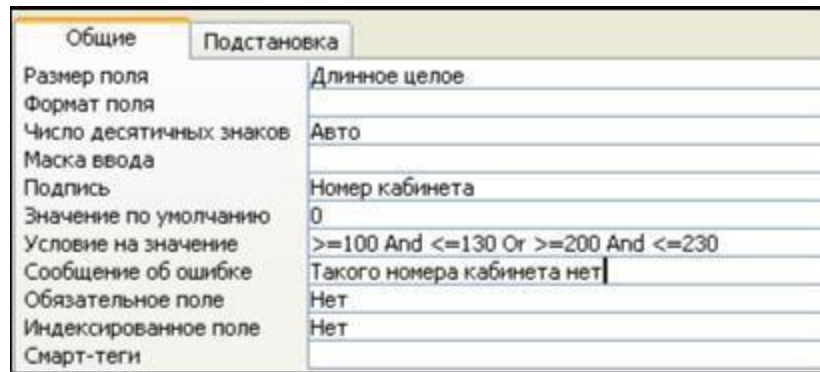


Рис 6.5. Властивості « Будувача виразів»

1.5. Створення ключових полів.

Первинний ключ - це поле (або ряд полів), яке однозначно ідентифікує кожний запис у таблиці. Access не допускає повторних значень в поле первинного ключа.

Зробіть поле **ID** лікаря ключовим. Для цього відкрийте таблицю

«Лікарі» в режимі **Конструктора** і, помістивши курсор на ім'я цього поля, клацніть по кнопці - Ключове поле (рис. 6.6):

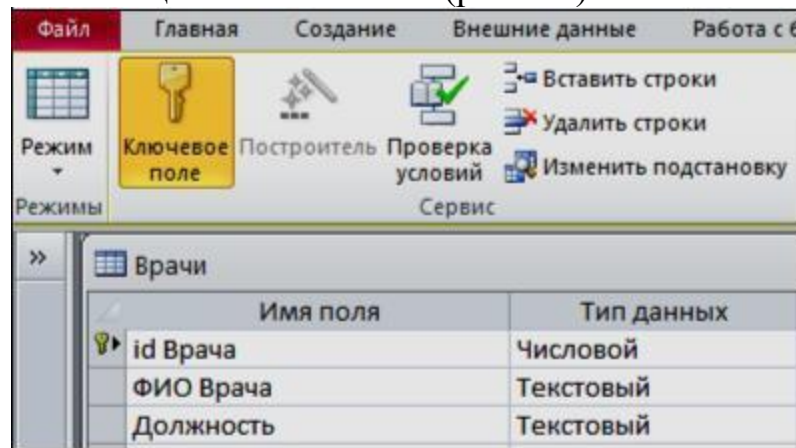


Рис.6.6 Вибір ключового поля

Закрийте таблицю. MS Access запропонує зберегти таблицю (за замовчуванням «Таблица1»). Задайте ім'я таблиці «Лікарі».

2. Заповнення таблиць

2.1. Створення таблиць-підстановок.

Для спрощення введення даних в таблицю « Лікарі » створіть таблиці-підстановки для полів: « **Термін прийому** » і «**Вид наданих послуг**». Для поля «Посада» дані будуть підставлятися з таблиці «Штатний розклад».

Підстановка спрощує режим введення. При цьому немає необхідності набирати довгі назви кілька разів. Досить буде вибрати необхідний запис зі списку.

2.1.1. Таблиця «**Зміна-подстановка**» (використовується для поля «**Термін прийому**»). Поля таблиці (див.табл.6.2):
Таблиця 6.2.Опис властивостей полів таблиці «**Зміна-подстановка**»

Смена_подстановка : таблица	
Имя поля	Тип данных
НомерСмены	Текстовый
ВремяПриема	Текстовый

Примітка 1. В результаті клацання по кнопці <OK> MS Access запропонує вам задати ключове поле. Для створюваної таблиці натисніть на кнопку <Ні>.

2.1.2 Таблиця «**Найменування послуг_підстановка**»

(використовується для поля «**Вид надаваних послуг**») створіть в Excel, щоб імпортувати її в нашу базу даних. Поля таблиці (див.табл.6.3):

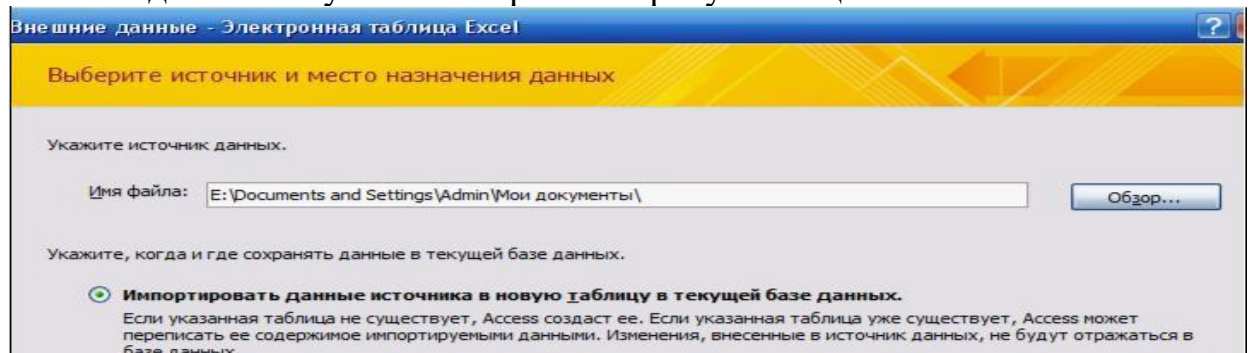
Таблиця 6.3 Зразок найменування

Найменування напрямки послуги
терапевтична
хірургічна
ортопедичне

Імпорт як самих таблиць, так і полів таблиць можна виробляти з різних джерел і файлів різних форматів. Визначте формат осередків: для поля **Найменування послуг** - текстовий. Збережіть таблицю в тій же папці, де знаходиться ваша БД «Стоматологія».

У базі даних «Стоматологія» на панелі інструментів виберіть «Зовнішні дані Excel».

У діалоговому вікні виберіть створену таблицю:



У наступному вікні «Імпорт електронної таблиці» виберіть Лист 1, натисніть <Далі>. встановіть прапорець «Перший рядок містить заголовки

стовпців», далі: «Дані необхідно зберегти в новій таблиці», далі підтвердить опис і назва полів, далі виберіть «Не створювати ключ», **Готово**.

Таблиця «**Найменування послуг**» буде імпортована в вашу БД. Подивіться її в режимі Конструктора. Чи всі поля відповідають типу даних?

2.1.3. Таблиця «Штатний Розклад»

(дані поля «**Найменування посади**» будуть використовуватися для підстановки в таблицю «**Лікарі**» (див.табл. 6.4).

Таблиця 6.4. Опис властивостей полів таблиці «Штатний розклад»

Штатное Расписание : таблица	
Имя поля	Тип данных
Должность	Текстовый
Оклад	Денежный
Надбавка	Денежный

Для поля «Оклад» у вкладці Загальні в рядку «Формат поля» вибрати «Грошовий».

Для поля «Посада» задати значення Ключове поле

2.2. Заповнення таблиць даними.

Для заповнення таблиць даними необхідно перейти з режиму Конструктора в режим Таблиці або відкрити таблицю подвійним клацанням.

Увага! (Дані в таблицю можна ввести тільки в режимі Таблиці).

Заповніть таблицю «**Зміна_підстановка**» даними (див.рис. 6.7).

Смена_подстановка	
НомерСмег	ВремяПрие
1	08.00-12.00
2	13.00-19.00

Рис.6.7 Приклад заповнення таблиці

Заповніть таблицю «**Штатний Розклад**» даними (див.рис. 6.8).

Штатное Расписание : таблица			
	Должность	Оклад	Надбавка
+	врач-стоматолог	7 000,00р.	700,00р.
+	ортопед	14 000,00р.	1 200,00р.
+	хирург-терапевт	10 000,00р.	1 000,00р.

Рис.6.8 Приклад заповнення таблиці

2.3. Створення підстановлювального поля.

Для створення підстановлювального поля використовується три способи: режим Конструктора. Цей спосіб використовується в тому випадку, якщо поле для підстановки стоїть першим в підстановочній таблиці, використання Майстра підстановок, або використання мови запитів SQL (за допомогою будівника запитів).

- Відкрийте таблицю «**Лікарі**» в режимі Конструктора. Для поля «Посада» у вкладці Підстановка виберіть в рядку Тип елемента керування:

«Поле зі списком»; в рядку Тип джерела рядків: «Таблиця або запит» (задається за замовчуванням); в рядку Джерело рядків зі списку вибрати таблицю «**Штатний Розклад**» - тобто таблицю, з якої дані будуть підставлятися в таблицю «Лікарі» (див. рис. 6.9):

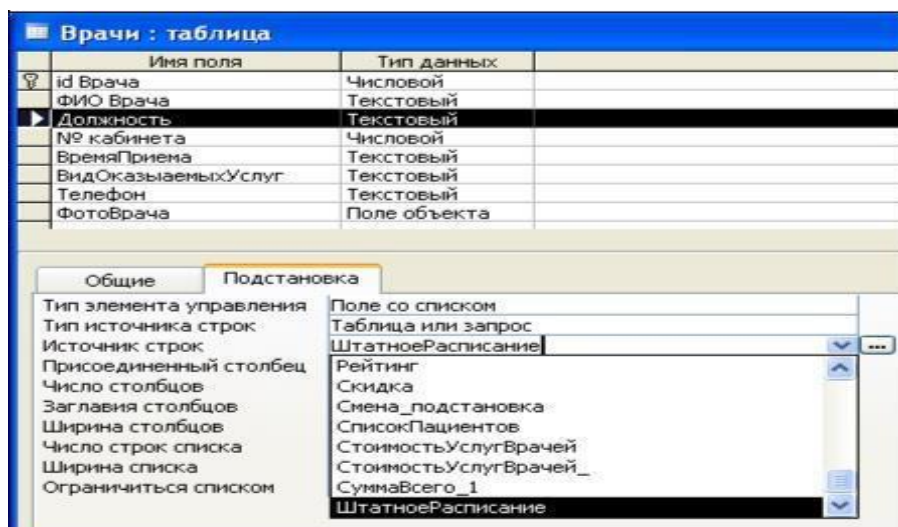


Рис. 6.9. Підстановка в режимі Конструктора

- Для створення підстановки для поля «**Термін Прийому**» в стовпці Тип даних виберіть зі списку «Майстер підстановки». У наступному вікні натисніть кнопку <Далі>, в діалоговому вікні виберіть «**Таблиця: Зміна підстановка**». Натисніть <Далі>.

У наступному вікні «Створення підстановки» зі списку «Доступні поля» виберіть за допомогою кнопки поле «**Термін Прийому**» (дані з цього поля будуть підставлятися в нашу таблицю), натисніть кнопку <Далі>, потім <Готово> .

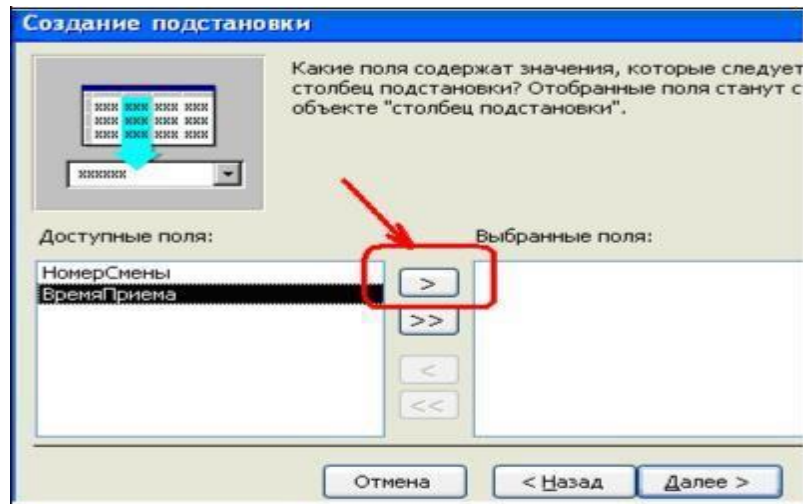


Рис.6.10. Використання Майстра підстановки

- Використання мови запитів SQL. Для використання будівника запитів для поля «**Термін Прийому**» виберіть вкладку Підстановка: в рядку Тип елемента керування - «Поле зі списком», в рядку Тип джерела рядків - «Таблиця або запит» в рядку Джерело рядків натисніть на кнопку Будівника виразів і в діалоговому вікні Інструкція SQL: будівник запитів на панелі інструментів виберіть: Вид Режим SQL.

запит матиме вигляд:

SELECT Зміна_підстановка.Термін Прийому FROM Зміна_підстановка;

2.4. Введення обмеження на дані в поле «ID лікаря».

Для поля «**ID лікаря**» в таблиці «**Лікарі**» виконаємо умови:

- коди лікарів не повинні повторюватися,
- повинна бути забезпечена можливість їх зміни (через останнього умови в цьому полі можна використовувати тип даних Лічильник, в якому дані не повторюються). Для виконання цих умов зробіть наступне: встановіть курсор на рядку «**ID лікаря**»; у вкладці Загальні клацніть по рядку параметра Індексовані поле; виберіть зі списку «Так (збіги не допускаються)»; перейдіть в режим Таблиця.

Примітка: Індекс - це засіб MS Access, що прискорює пошук і сортування даних в таблиці.

2.5. Заповніть таблицю «**Лікарі**» даними.

Порядок виконання практичної роботи

Розробка зведеної таблиці для таблиці бази даних.

Відкрити таблицю «**Журнал**» в режимі зведеної таблиці. Щоб відкрити таблицю в цьому режимі, виберіть її у вікні бази даних, натисніть кнопку «Відкрити», а потім для переходу з режиму таблиці в режим зведеної таблиці виберіть на панелі Вид «**Вільна таблиця**».

При першому відкритті в її робочу область не входять поля таблиці (див. Рис. 6.11):

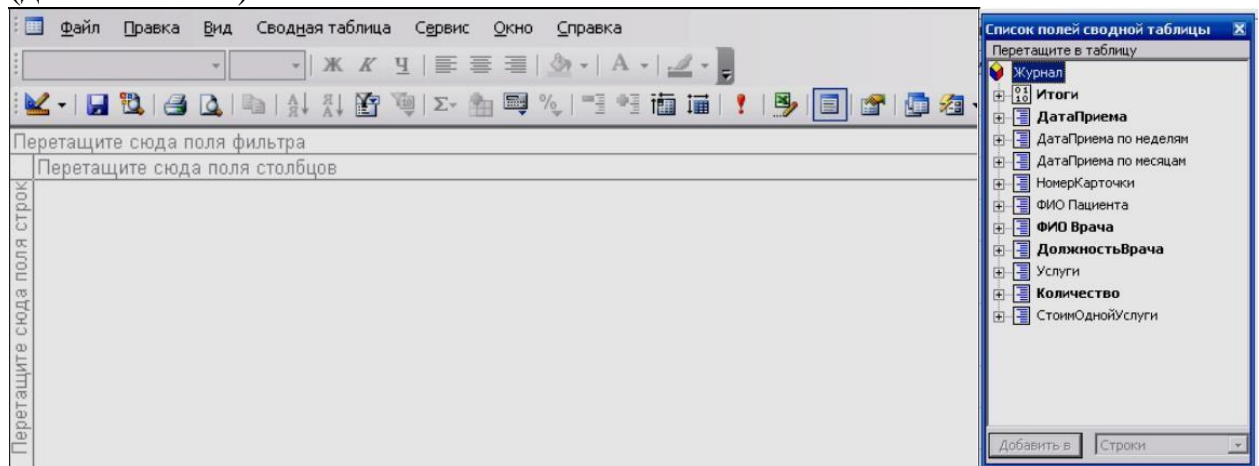


Рис. 6.11. Порожній макет зведеної таблиці «Журнал» і список полів джерела записів

Нехай необхідно підрахувати кількість виконаних послуг кожним лікарем за кожен день (за певний період). Якщо список полів цієї таблиці не відображається, натисніть на панелі інструментів піктограму «Список полів».

Унікальні значення прізвища лікарів складуть заголовки рядків. Перетягніть в цю область поле «**П.І.Б Лікарів**». В область полів стовпців перетягніть поле «**ДатаПрийому**». Нехай у нашому прикладі фільтрація складе за посадами. Перетягніть в область фільтра - поле **Посада Лікаря**

ДолжностьВрача	ДатаПриема	01.01.2011	02.01.2011	03.01.2011	04.01.2011	06.01.2011	09.01.2011	01.02.2011	09.02.2011	Общие итоги
ФИО Врача	Количество	Количество	Количество	Количество	Количество	Количество	Количество	Количество	Количество	Нет итогов
Валиуллин Д.К.	+	1	2					1		
Галимянова О.А.	+	5							1	
Дюкина Я.Б.	+	4								
Елизарова Е.П.	+					2		2		
Жуков А.Е.	+			2						
Карбышев Г.О.	+			2			3	1		
Леонидова М.Р.	+				3			2		
Накипов Н.Д.	+		2					1		
Никоненко П.С.	+					2		1		
Родинова Ф.Я.	+			1						
Сидорович М.Л.	+					4		1		
	+					2				
	+					1				
Халимов И.Ш.	+					2				1
Харитонов Н.Н.	+						4			
Шабернева Ф.Н.	+							1		2
Общие итоги	+									

Рис. 6.12. Зведена таблиця з прізвищами лікарів і датами прийому

Для підрахунку підсумкових значень клацніть в області даних правою кнопкою миші і виберіть Автовизначення Сума (див. Рис. 6.13).

ФИО Врача	01.01.2011	02.01.2011	03.01.2011	04.01.2011	06.01.2011	09.01.2011	01.02.2011	09.02.2011	Общие итоги
Валиуллин Д.К.	1	2					1		4
Галимянова О.А.	5							1	6
Дюкина Я.Б.	5							1	6
Елизарова Е.П.	4					2	2		4
Жуков А.Е.			2			2	2		4
Карбышев Г.О.			4						4
Леонидова М.Р.				3		3			6
Накипов Н.Д.			2	3					5
Никоненко П.С.			2				1		3
Родионова Ф.Я.					2	1	1		3
Сидорович М.Л.				1				1	2
					4				8
					2				2

Рис. 6.13. Зведена таблиця з обчислюваними полями.

САМОСТІЙНО виконайте фільтрацію даних і виведення результатів за посадами, використовуючи поле «Посада Лікаря», вибравши **ВСЕ** або окремі посади.

Також можна отримати зведену діаграму по таблиці, вибравши на панелі інструментів Вид Зведена діаграма. (див.рис. 6.14).

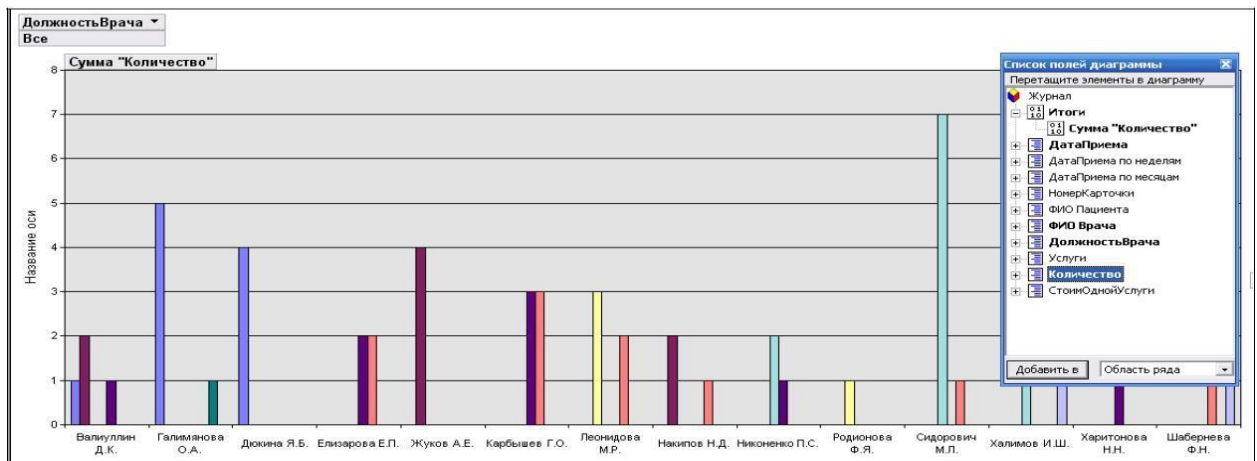


Рис. 6.14. Таблица «Журнал» в режимі зведеної діаграми

2. Створення звітів бази даних

Звіти використовуються для відображення інформації, що міститься в таблицях, в вигляді, який легко зчитується як на екрані комп'ютера, так і на папері.

Звіт можна відобразити в трьох режимах: в режимі конструктора, що дозволяє змінити зовнішній вигляд і макет звіту, в режимі перегляду зразка, де можна переглянути всі елементи готового звіту, але і в скороченому вигляді; в режимі попереднього перегляду.

Звіт можна створити в режимі конструктора і в режимі майстра.

Для створення звіту перейдіть у вкладку Звіти Створення за допомогою майстра. Виберіть таблицю «Штатний розклад», поля Посада, Оклад, Надбавка.

З таблиці «Лікарі» виберіть поля: «ІД, П.І.Б. Лікаря, Вид наданих послуг». Натисніть кнопку <Далі>.

У наступному діалоговому вікні в рядку «Вид уявлення даних» буде обрана перша таблиця, в нашому прикладі «ШтатноеРасписание» Натисніть кнопку <Далі>. Додайте за допомогою стрілок в рівень представлення даних поле Должность. У наступних вікнах виберіть дизайн звіту. <Готово>.

Приклад готового звіту представлений на рис 6.15.

ШтатноеРасписание					
Должность	Оклад	Надбавка	id	ФИО Врача	Оказываемые Услуги
врач-стоматолог					
	7 000,00р.	2 500,00р.			
			104	Жуков А.Е.	Терапевтическая
			102	Дюкина Я.Б.	Терапевтическая
			112	Харитонов Н.Н.	Терапевтическая
			109	Родионова Ф.Я.	Терапевтическая
			106	Леонидова М.Р.	Терапевтическая
			103	Елизарова Е.П.	Терапевтическая
			113	Шабернева Ф.Н.	Терапевтическая
			110	Сидорович М.Л.	Терапевтическая
ортопед					
	14 000,00р.	3 000,00р.			
			105	Карбышев Г.О.	Ортопедическая
			100	Валиуллин Д.К.	Ортопедическая
			108	Никоненко П.С.	Ортопедическая

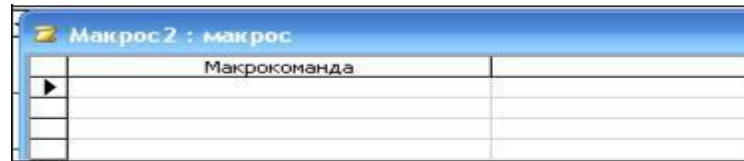
Рис. 6.15. Звіт бази даних

3. Створення макросів.

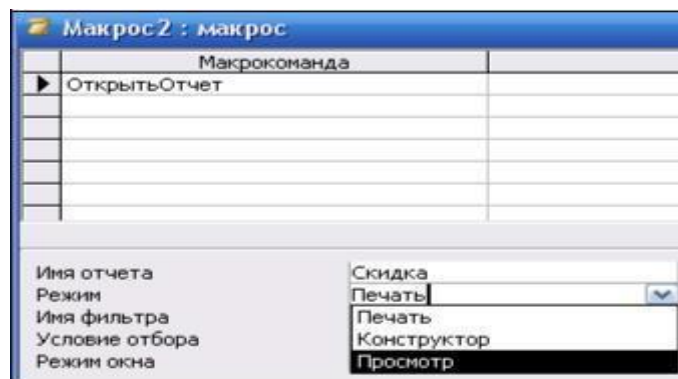
Макроси - це невеликі програми на мові макрокоманд СУКД MS Access, що складаються з послідовності певних команд (однієї або декількох макрокоманд). Макроси є найпростішими засобами автоматизації дій над об'єктами MS Access. Необхідно відзначити, що макроси легко створювати, і для цього не потрібно знати синтаксис мови програмування.

Для створення макросу, відкрийте головну форму в режимі конструктора і додайте кнопку. Нехай за допомогою новоствореної кнопки можна відкрити звіти для перегляду по діяльність бази даних «Стоматологія».

Натиснути правою кнопкою миші по полю кнопки і виберіть Обробка подій Макрос. Відкриється вікно конструктора макросів.



У цьому вікні треба вибрати макрокоманду зі списку, натиснувши на кнопці відкриття списку макрокоманд. У списку є близько 50 різних макрокоманд для виконання дій. Для нашого прикладу виберіть: Відкрити звіт.



Виберіть ім'я звіту і режим: Перегляд. Збережіть макрос. Всі створені макроси будуть відображатися в вкладці Макроси.

САМОСТІЙНЕ Додайте інші кнопки, доповнивши їх відповідними командами-макросами.

Контрольні питання

1. Призначення звіту, способи створення звітів.
2. Створення звітів у режимі конструктора, особливості виконання.
3. Поняття запиту, способи створення запитів.
4. Створення запиту у режимі конструктора, особливості виконання.
5. Виконання запиту з пошуковими реквізитами.
6. Створення за допомогою «Будувача виразів».
7. Призначення функцій Sum, Count, особливості їх застосування.
8. Призначення функції Var.
9. Поняття дисперсії та середнього відхилення від дисперсії.

ПРАКТИЧНА РОБОТА №7

Тема: Розрахунок параметрів та розробка технічного завдання для індивідуального проєкту

Мета :Створення пристрою моніторингу температури та атмосферного тиску на базі хмарних технологій

Перед створенням пристрою моніторингу треба зробити налаштування хмарних сервісів, які будуть використовуватись для вирішення поставлених завдань.

ESP8266 досить потужний процесор, має багато пам'яті, вбудований контролер WiFi. Але він не може самостійно візуалізувати значення температури або атмосферного тиску за деякій проміжок часу, наприклад, за місяць. Для вирішення цього питання будемо використовувати якийсь сервер - посередник, який прийматиме повідомлення від пристрою і передаватиме їх користувачу, і навпаки. Крім того, на сервер також лягає завдання відобразити графіки змін тиску та температури. Звичайно, є варіанти і прямого зв'язку, але вони не розглядаються в рамках курсу.

Раніше вже проводився аналіз можливості тих чи інших безкоштовних сервісів, які допоможуть вирішити поставлене завдання. Для вирішення поставленого завдання краще підходить сервіс ThingSpeak.

Це відкрита IoT-платформа із MATLAB-аналітикою. Сервіс компанії Math Works, створений для Інтернету речей, тісно пов'язаний із продуктами Math Works (MATLAB). ThingSpeak - це служба аналітичної платформи IoT, яка дозволяє збирати, візуалізувати та аналізувати потоки даних у реальному часі у хмарі. Ви можете надсилати дані до ThingSpeak зі своїх пристроїв, створювати миттєві візуалізації даних у реальному часі та надсилати оповіщення за допомогою веб-сервісів, таких як Twitter та Twilio. Завдяки аналітиці MATLAB усередині ThingSpeak ви можете писати та виконувати код MATLAB для виконання попередньої обробки, візуалізації та аналізу. ThingSpeak доступний як безкоштовний сервіс для некомерційних невеликих проєктів (<3 мільйони повідомлень на рік або ~ 8200 повідомлень на день). Сервіс дозволяє накопичувати дані в своїх сховищах, після чого їх можна зручно переглядати у вигляді графіків.

З особливостей сервісу він має інтеграцію з MatLab Analytics з підтримкою скриптів MatLab. У безкоштовному обліковому записі сервіс дозволяє створювати до чотирьох каналів, у кожному з каналів можна зберігати до 8 полів з різними даними. Ще в безкоштовному акаунті існує ще одне обмеження - не можна публікувати дані частіше одного разу на 15 секунд (рекомендується пауза в 20 секунд), це як би "захист" від перевищення ліміту 8200 повідомлень на день.

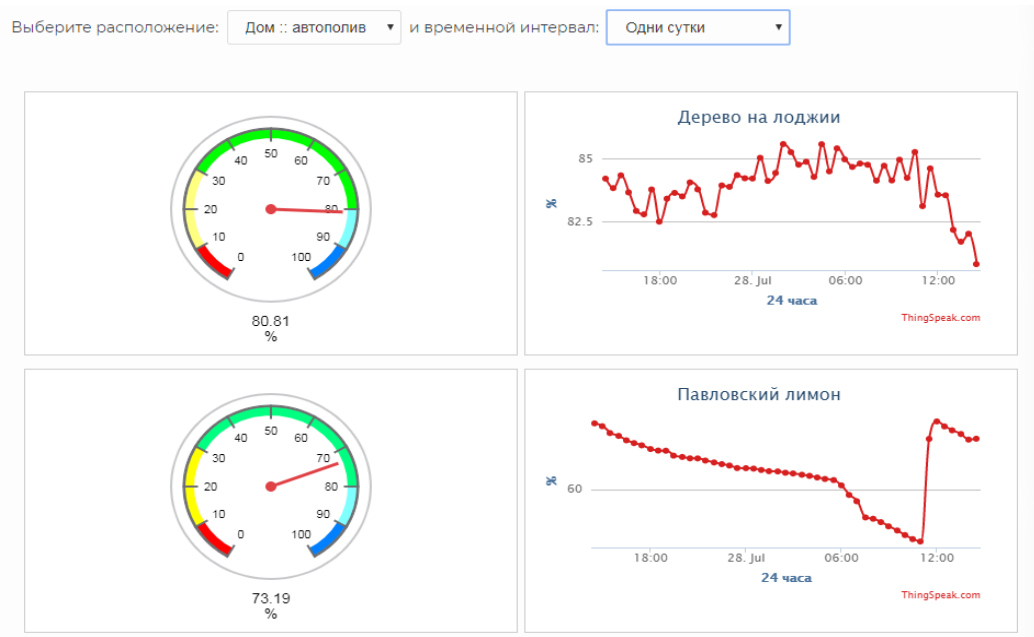


Рисунок 7.1 - Сервіс ThingSpeak

У принципі, пристроєм легко вписатися в цей ліміт для даного завдання. Є кілька програм – клієнтів для Android, але набагато менше, ніж для MQTT. Плюси: просте використання на ESP, зручний сайт. Мінуси: багато обмежень на безкоштовному обліковому записі.

Таким чином, робимо висновок про використання сервісу ThingSpeak для обробки отриманих даних та подальшої їх обробки.

Порядок виконання роботи

1. Зробити обліковий запис на запропонованому сервері ThingSpeak. Допустимо створювати обліковий запис на будь якому іншому сервері за побажанням, але методику треба засвоїти особисто і завантажити відповідний приклад використання.
2. Використовуючи плату на основі ESP8266 підключити датчики атмосферного тиску та температури до портів вводу-виводу мікроконтролера. Для даного прикладу використовуються датчики DS18B20, MS5611. Перший датчик підключається по інтерфейсу OneWire другий по I2C. Які порти обрати для підключення залежить від обраної плати.
3. Для отримання даних також потрібно знати час, з цим завданням допоможе модуль на основі DS3231.
4. Для швидкого створення пристрою буде використовуватись Arduino.
5. Створити новий проєкт та підключити до нього такі бібліотеки:


```
#include <Arduino.h>
#include <ESP8266WiFi.h>
```



```
#include <ESP8266Ping.h>
#include "ThingSpeak.h"
#include "uRTCLib.h"
#include <SPI.h>
#include <Wire.h>
#include <DS18B20.h>
#include <MS5611.h>
#include <I2C_eeprom.h>.
```

Підключення необхідних бібліотек треба робити з інтерфейсу програмної середовища Arduino. Це дозволяє завантажувати автоматично необхідні файли.

6. Додати до коду ініціалізацію ThingSpeak та WIFI


```
WiFi.mode(WIFI_STA);
WiFi.begin(ssid, pass);
ThingSpeak.begin(client); // Initialize ThingSpeak
WiFi.setAutoConnect(true);
WiFi.setAutoReconnect(true);
```
7. Додати код який буде зчитувати показники з датчиків та відправляти до серверу. Приклад коду наведено нижче:


```
int httpCode = ThingSpeak.writeFields(myChannelNumber,
myWriteAPIKey);
  if (httpCode == 200) {Serial.println("Channel write successful.");}
  else {Serial.println("Problem writing to channel. HTTP error code " +
String(httpCode));}
```
8. Для випадку використання серверу ThingSpeak забезпечити у програмі обмеження більше 25-30 секунд між відправками, в іншому випадку на безкоштовній основі серверне буде забезпечувати запис даних.
9. Оформити звіт з практичної роботи.

Рекомендована література

Основна:

1. Панченко Т.В. [та ін.] Функціонально активні та інтелектуальні матеріали: підручник. Дніпро: Дніпровський національний університет імені Олеся Гончара. – Акцент ПП, 2017. – 128 с.
2. Розорінов Г.М. , Соловійов Д.О.. Високошвидкісні волоконно-оптичні лінії зв'язку: навч. посіб.– 2-е вид., перероб. і допов. Київ.: Кафедра, 2020. – 344с.
3. Лепіх Я. І., Гордієнко Ю.О., Дзядевич С.В. та ін. Створення мікроелектронних датчиків нового покоління для інтелектуальних систем: Монографія . Одеса: “Астропринт”, 2018. – 296 с.
4. Ленков С. В. та ін. Мікроелектронні датчики для інформаційних (інтелектуальних) систем спеціального призначення. Наука і оборона, 2010 - №3- 55-58с.
5. Філяшкін М.К. Мікроелектронні інформаційні системи: Навчальний посібник – К.: НАУ, 2019. – 276 с.

Додаткова:

1. Методологія інформаційних систем та баз даних: теоретичний і практичний підходи: навч. посібник / уклад. Ю.О. Ушенко, М.Л. Ковальчук, М.С. Гавриляк, А.Л. Негрич.– Чернівці : Чернівецький нац. ун-т ім. Ю. Федьковича, 2021. 240 с.
2. Бройдо, В.Л. Обчислювальні мікроелектронні системи, мережі та телекомунікації: посібник. Київ, 2006 .703 с.
3. Проценко І. Ю., Н. І. Шумакова. Наноматеріали і нанотехнології в електроніці : Підручник. Суми : Сумський державний університет, 2018. – 155 с.
4. Основи електроніки з елементами мікроелектроніки : навч. посіб. / Укл. : П. Г. Стахів, В. І. Коруд, О. Є. Гамола та ін. – Львів : Магнолія-2006, 2019. – 225 с.
5. Ленков С. В., Селюков О. В. Деякі проблеми з області розробки й модернізації військової техніки. Погляд з середини. Наука і оборона, 2007. – № 2 – 38–41с.
6. Готра З.Ю. Фізичні основи електронної техніки.: Підручник. Львів: Бескид Біт. , 2017.- 55с.
7. Кулаков, Ю.О. Комп'ютерні мережі : підручник для ВНЗ. Київ: Юніор, 2018. 395 с.
8. Мюллер, С. Модернізація та ремонт ПК: пер. з англ. «Вільямс», 2019.1180 с.
9. Arduino. Статті 2018. URL: <https://www.arduino.cc/en/Guide/HomePage>.
10. Y. Wu, D. B. Farmer, W. Zhu, S.J. Han, C. D. Dimitrakopoulos, A. A. Bol, P. Avouris, and Y. M. Lin, "Three-Terminal Graphene Negative Differential Resistance Devices", ACS Nano, vol. 6 (3)/ 2012, pp 2610-2616.

Інформаційні ресурси:

- 1.Наукова бібліотека Запорізького національного університету. URL: <http://library.znu.edu.ua/>
- 2.Система електронного забезпечення навчання ЗНУ. URL: <https://moodle.znu.edu.ua/>
- 3.Національна бібліотека України імені В. І. Вернадського. URL: <http://www.nbuv.gov.ua/>
4. AnyLogic: імітаційне моделювання для бізнесу URL: <https://www.anylogic.com/>
5. Електронні кромпоненти. URL:<https://uk.wikipedia.org/wiki>
- 6 Electronics Tutorials (Basic Electronics Tutorials and Revision) веб-сайт. URL: <http://www.electronics-tutorials.ws> (дата звернення 31.05.2023)