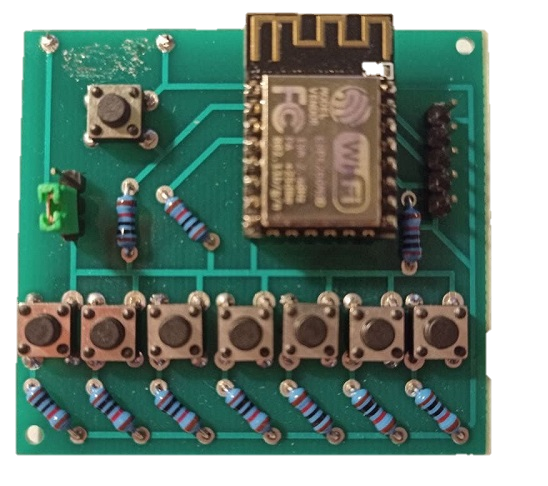
ЛІЦЕЙ №208 М. КИЄВА

**Створення пультової системи голосування**

****

**Виконав:**

Лагодич М. Д., 9-В клас

**Керівник:**

Смикало Я. Й.

Київ 2021

Зміст

[Зміст 2](#_Toc70794830)

[Вступ 3](#_Toc70794831)

[Ідея проекту 4](#_Toc70794832)

[Завдання проекту 4](#_Toc70794833)

[Хід роботи 5](#_Toc70794834)

[Створення пультів голосування та сервера 5](#_Toc70794835)

[Пульти 7](#_Toc70794836)

[Сервер 8](#_Toc70794837)

[Створення програмного забезпечення 9](#_Toc70794838)

[Створення засобів автоматизації 10](#_Toc70794839)

[Висновок 11](#_Toc70794840)

# Вступ

У сучасному світі є багато технологій презентування інформації. Зараз вже не є проблемою побачити на власні очі космічні станції, вимираючих рідкісних тварин та навіть ландшафти інших планет, на яких ми можемо ніколи і не побувати.

Але для якісного навчання недостатньо одного презентування. Потрібний ще і зворотній зв’язок, коли доповідач ставить запитання за щойно розказаним матеріалом. З таким підходом доповідач адаптує свою розповідь під конкретну аудиторію та навіть окремих слухачів, а аудиторія краще запам’ятовує та розуміє матеріал.

Зворотній зв’язок дуже просто отримати, коли вся аудиторія складається із 10-20 людей, тобто за допомогою піднятих рук і словесних відповідей. Та якщо доповідь ведеться на конференції на 200-300 людей, такий метод не буде ефективним. Для такого випадку зворотний зв’язок потрібно автоматизувати.

# Ідея проекту

1. Доповідач на етапі підготовки додає у свою презентацію слайди з питаннями та кількома (2-7) варіантами відповіді. Пізніше, під час доповіді, розповідач ставитиме ті запитання аудиторії. Зворотним зв’язком буде вважатися статистика вигляду «варіант – кількість людей, що обрали цей варіант».
2. Під час доповіді у кожного слухача є пристрій з кількома кнопками. Одна кнопка – один варіант відповіді. При натисканні кнопки пристрій без дротів передає обраний варіант на комп’ютер доповідача.
3. Комп’ютер доповідача малює діаграму, що буде результатом зворотного зв’язку.

Використовувати в якості таких пристроїв смартфони слухачів – не оптимальний варіант, адже не у кожного є смартфон з потрібним програмним і/або апаратним забезпеченням, яке дозволить без проблем запустити програму для зворотного зв’язку. Отже для даної задачі доцільніше створити окремі пульти з кнопками та модулями бездротового зв’язку.

Процес ззовні схожий на процес голосування, тому далі такі пульти будуть називатися «пультами голосування».

Далеко не кожен сучасний комп’ютер чи ноутбук має модуль бездротового зв’язку, тому для комп’ютера доповідача потрібно створити ще пристрій-приймач (далі – «сервер») та програмне забезпечення, що опрацювуватиме прийняті дані та виводитиме діаграму.

# Завдання проекту

Створити пульти голосування, сервер, програмне забезпечення для них та для комп’ютера доповідача. Також створити документацію проекту та автоматизовані засоби для простого повторення проекту іншими людьми.

# Хід роботи

## Створення пультів голосування та сервера

Пульти та сервер повинні мати бездротовий зв’язок, при чому такий, який дасть змогу підключити десяти пристроїв та передавати інформацію без втрат, а також бажано, щоб кожному пристрою-клієнту присвоювався певний ідентифікатор, за допомогою якого система зможе записувати відповіді окремих респондентів.

На сьогоднішній день дуже розповсюджений протокол Wi-Fi. Він цілковито підходить під мої потреби, а до того ж систему з Wi-Fi можна буде відлагоджувати за допомогою звичайного смартфона, адже багато сучасних телефонів сумісні з ним.

Отже кожен пульт має складатися із блоку керування та WiFi-модуля. Блок керування слідкуватиме за натисканням кнопок і транслюватиме номер натиснутої кнопки серверу через WiFi-модуль за допомогою протоколу HTTP (найрозповсюдженішого Інтернет-протоколу).

Такий алгоритм виходить занадто складний для виконання простою електричною схемою, тому блоком керування має виступити мікроконтролер – програмуєма мікросхема, що є повноцінним комп’ютером.

Існує мікроконтролер під назвою ESP8266, що є програмуємим модулем Wi-Fi, тобто один пристрій виконує одразу дві функції. Модуль достатньо відомий, для нього існує зручне середовище програмування та прості у використанні бібліотеки. Отже мій вибір упав на модуль ESP8266.

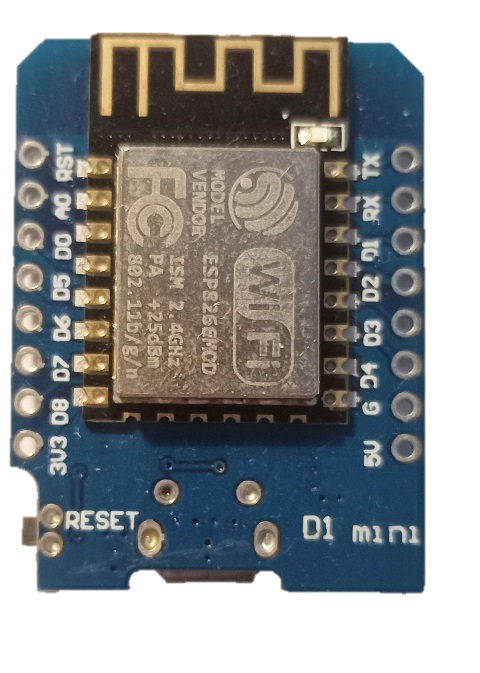


Рисунок 1. Модуль ESP8266 на платі-перехіднику

Слід зазначити, що такий модуль може виступати і в ролі керуючої частини пульта, і в ролі сервера. Будучи в ролі сервера, ESP8266 може створити власну WiFi-мережу (в режимі так званої *точки доступу*) або користуватися сторонньою мережою (в режимі *клієнта*), наприклад впровадженої звичайним WiFi-роутером. Можливість переходити в різні режими слід буде врахувати при написанні програмного забезпечення.

Сервер потребуватиме під’єднання до комп’ютера доповідача. Сьогодні де-факто стандартом комунікування периферичних пристроїв з персональними комп’ютерами є протокол USB, тому саме його матиме використовувати сервер.

Проте сам по собі модуль ESP8266 не може з’єднуватись по USB, бо потребує іншогої напруги живлення (3,3 В, а для USB застосовується 5 В) та розрахований на інший протокол обміну даними (UART). Для вирішення даної проблеми існують плати-перехідники (*рис. 1*), на які WiFi-модулі впаюються початково, на заводі. Такий комплекс вже повністю готовий до програмування та подальшої роботи.

Отже, тепер частина системи, периферична до комп’ютера доповідача, визначена.

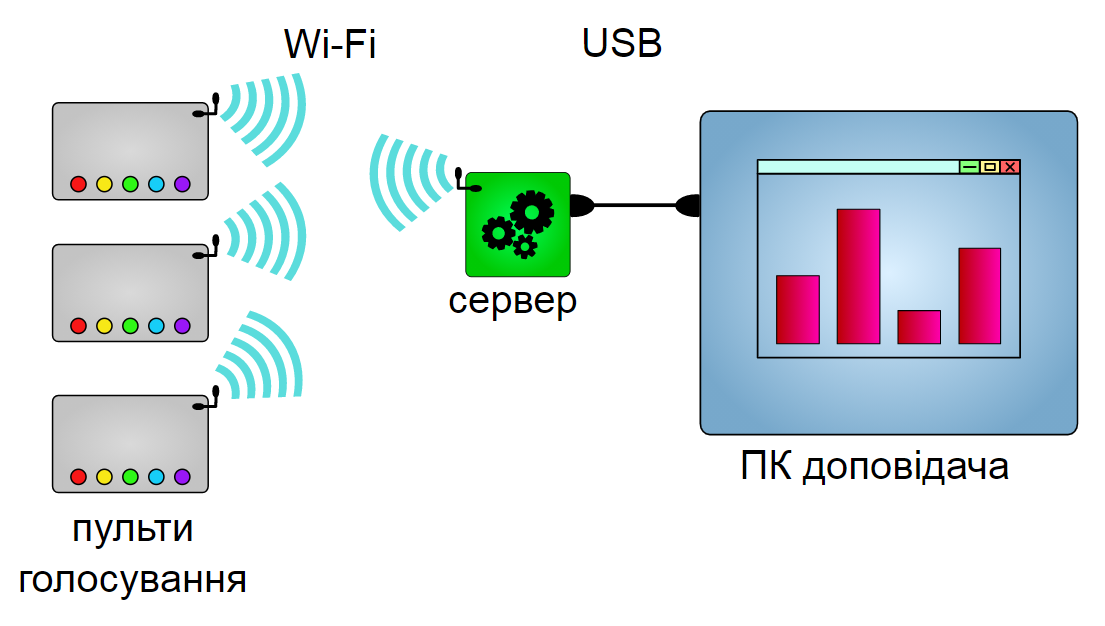


Рисунок 2. Загальна структура системи

### Пульти

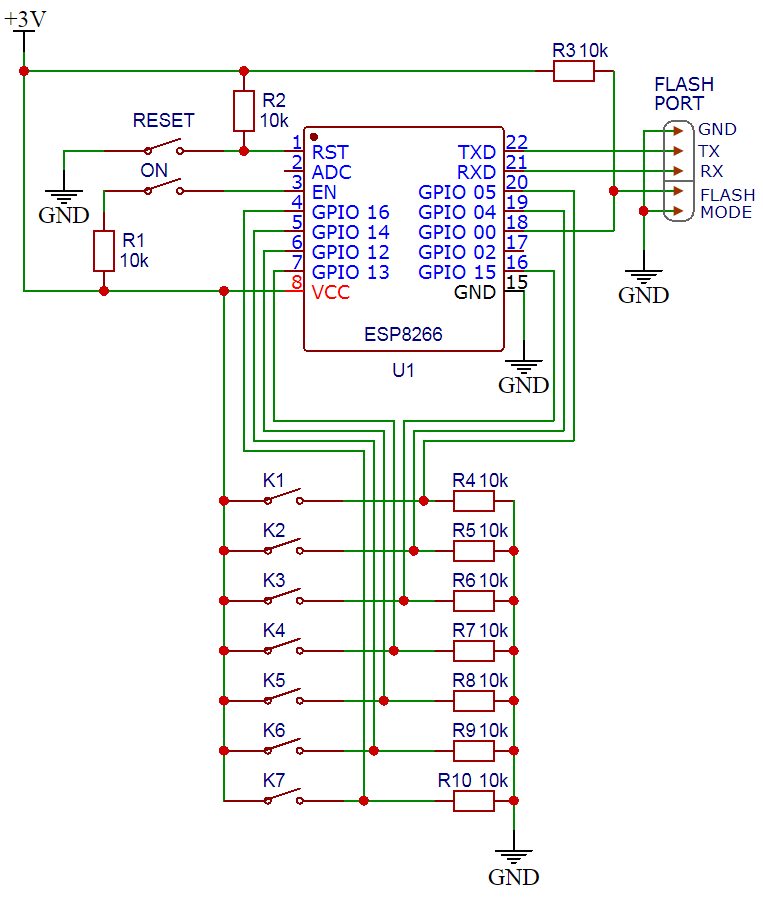


Рисунок 3. Схема пульта

Внутрішню будову пульта поясню за принциповою схемою (*рис. 3*).

*Живлення* до модуля ESP8266 (U1) під’єднується до контактів VCC та GND, але модуль для увімкнення потребує подачі високого рівня сигналу на контакт EN (від англ. Enable - увімкнути).

*Перезавантаження* необхідне для пульта, як і для будь-якого пристрою. Роль перезавантажуючого механізму виконує кнопка RESET, що при замиканні подає низький рівень на контакт RST (від англ. Reset - перезавантажити). Зауважимо, що для коректного спрацьовування необхідне підтягнення контакту RST до високого рівня резистором R2 (детальніше про підтягнення див. нижче).

*Перезапис програмного забезпечення* реалізується за допомогою порта FLASH (від англ. спалах, тут: швидкий запис). Контакти GND, TX та RX під’єднуються за допомогою UART-USB перехідника до комп’ютера із ПЗ, потрібним для перезапису. Контакти, об’єднані під назвою FLASH MODE, замикаються для введення модуля ESP8266 в режим програмування.

*Клавіатура* із кнопок К1-К7 слугує для введення відповіді. При натисканні кожна кнопка подає сигнал високого рівня на відповідний контакт GPIO (від англ. General Purpose Input/Output – загальний ввід/вивід). Для введення не використовуються контакти GPIO 00 та GPIO 02, бо вони є службовими (GPIO 00 при ініціалізації ESP8266 вводить його в режим програмування, якщо містить низький рівень, а до GPIO 02 під’єднаний світлодіод на корпусі, і його підтягнення викликає помилку ініціалізації).

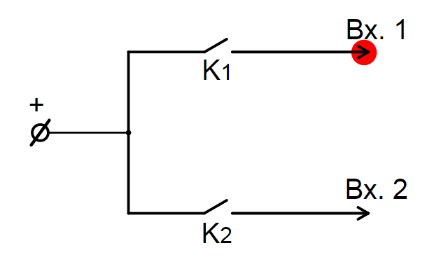
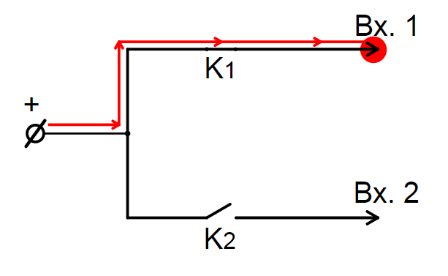
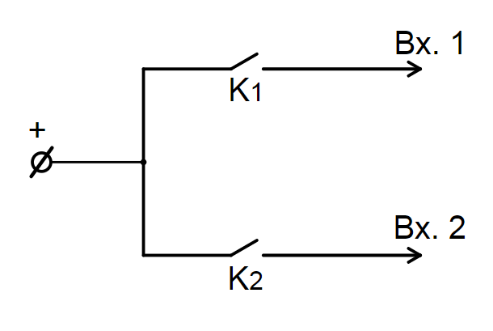
*Підтягнення* – це механізм втримання на контакті рівня сигналу, що є протилежним до рівня спрацьовування. Розберімо проблему на прикладі кнопок:

*(в)*

*(б)*

*(а)*

Рисунок 4. Поведінка пристрою без підтягування



Нехай існує кнопка К1 (*рис. 4.а*), що при замиканні подає високий рівень на вхід 1 (*рис. 4.б*). При розімкненні кнопки (*рис. 4.в*) на вході 1 лишається заряд, що сприймається мікроконтролером як сигнал високого рівня. Таким чином, кнопка може спрацювати лише один раз, що є проблемою.

Приєнаймо вхід 1 до негативного полюса джерела живлення через резистор R1 (такий резистор називається *підтягуючим*). Після розімкнення кнопки К1 залишковий заряд стіче до мінуса, і мікроконтролер сприйме відсутність заряду на вході 1 як низький рівень.

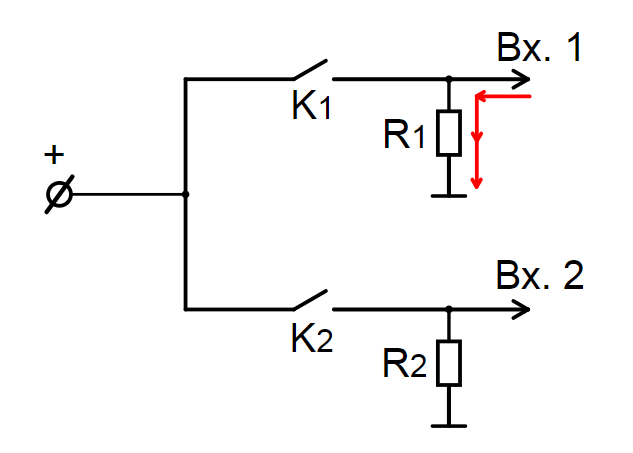


Рисунок 5. Функція підтягнення

Отже всі контакти, використані для введення сигналу, мають бути підтягненими.

### Сервер

Через те, що всі модулі ESP8266 з’єднані однією мережею Wi-Fi, кожному присвоюється свій ідентифікатор (IP-адреса).

Приймаючи натискання кнопок від пультів, сервер запам’ятовує і надісланий номер кнопки, і кожен такий ідентифікатор, щоб уникнути повторів: одна людина може відправити 100 натискань, кардинально змінюючи статистику, що призведе до хибного трактування результатів доповідачем. Отже, приймаючи чергове повідомлення від пульта, сервер повинен перевірити його ідентифікатор, і тільки якщо такого ще не має у списку, відповідь зараховується.

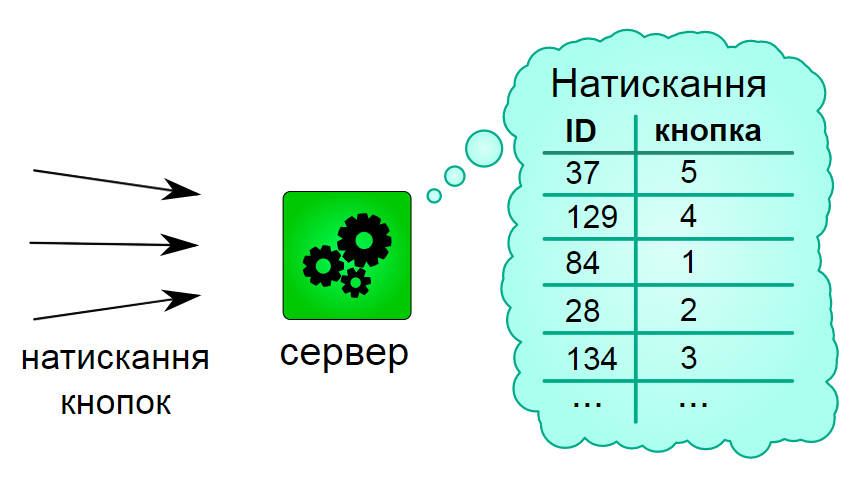


Рисунок 6. Пам'ять сервера

Порядок зв’язку сервера з ПК простий: ПК надсилає серверу команди («отримати результати» або «забути поточні результати»), а сервер надсилає ПК результати (натискання кнопок), якщо була відповідна команда.

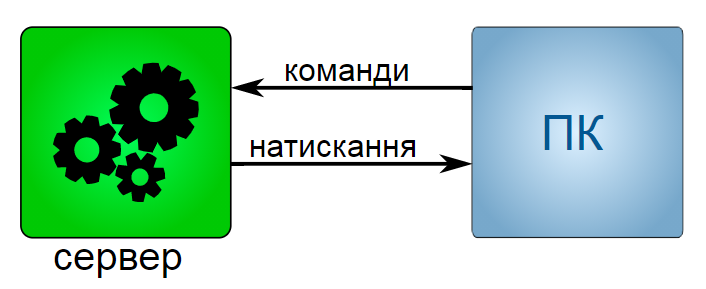


Рисунок 7. Обмін інформацією між сервером та ПК

## Створення програмного забезпечення

## Створення засобів автоматизації

# Висновок

Я створив систему бездротового пультового голосування, інтегровану в програму Microsoft PowerPoint. Найкраще така система підходить для аналізу якості сприйняття інформації аудиторією та для скринінгового бліц-опитування, коли користуватися власними гаджетами учасникам заборонено.

Під час виконання свого проекту я оволодів такими технологіями та знаннями:

* мовою програмування VBA;
* інструментами розробників для Microsoft Office;
* протоколами Wi-Fi, HTTP та форматом збереження даних JSON;
* особливостями роботи мікроконтролера ESP8266;
* особливостями програмування мікроконтролерів мовою C++ у середовищі Arduino IDE;
* системою контролю версій Git;
* онлайн-сервісом хостингу проектів GitHub;
* мовою розмітки Markdown для опису та документації проектів,

також я здобув такі навички:

* конструювання електронних схем за допомогою онлайн-сервісів на основі знань фізики;
* написання та налагодження програмного коду;
* створення комп’ютерних систем;
* автоматизації своєї роботи за допомогою скриптових мов програмування;
* виявлення недоліків сторонніх програм та їх обхід або виправлення,

і до того ж я поглибив свої знання мов програмування C++ та Python.