

МЕТОДИЧНІ РЕКОМЕНДАЦІЇ
для проведення лабораторних занять (заочна форма)
з навчальної дисципліни
«АРХІТЕКТУРА І ТЕХНОЛОГІЇ ІОТ»

Загальні відомості

ESP8266 та ESP32 – це недорогі модулі, які ідеально підходять для самостійних проєктів у галузі Інтернету речей (IoT). Плати оснащені інтерфейсом Wi-Fi і підключаються до комп'ютера через WiFi або USB порт. Писати програмні коди (скетчі) під ці мікроконтролери можна в середовищі розробки «Arduino IDE». Всі завдання для лабораторних робіт передбачають використання мікроконтролера ESP8266. Однак можливо їх виконання також із використанням мікроконтролера ESP32. Для виконання лабораторних робіт використовуються приклади розробки проєктів IoT за посиланням <https://randomnerdtutorials.com>.

Виконання лабораторних робіт можливо без придбання мікроконтролерів з використанням онлайн симулятора для Arduino «Wokwi». Він імітує як роботу мікроконтролера ESP32, так і деяких датчиків і дозволяє зібрати віртуальну схему з модулів та запустити на ній програму.

Як звіт із виконання лабораторних робіт можливо надіслати посилання на запис відео із порядком виконання роботи. Приклад звіту з лабораторної роботи 1 можна подивитися за посиланням:

<https://drive.google.com/file/d/1zYZQmH9ytyczwVjOYYwH6hK43Z0u2Z6V/view?usp=sharing>

Можливо також надання звітів у форматі PDF із копією екрану з результатами роботи проєкту та текстом коду програми із висновками. Надішліть звіт (посилання на Google диск) на електронну пошту викладача. Кожна завершена робота оцінюється на 12 балів.

Лабораторне заняття 1. Створення проєкту IoT «Розумна розетка, що синхронізується за NTP-сервером».

Порядок виконання

1. Встановіть середовище Arduino IDE на комп'ютері (наприклад, версія 1.8.5). У середовищі «Arduino IDE» є «Менеджер плат», який допоможе додати плати, що цікавлять, до списку.

2. Встановлення плат ESP8266 та ESP32 в середовищі Arduino IDE.

Відкрийте «Arduino IDE» та виберіть пункт меню «Файл» > «Параметри». У вікні заповніть поле «Додаткові посилання для Менеджера плат», ввівши в нього рядок: http://arduino.esp8266.com/stable/package_esp8266com_index.json та натисніть на кнопку «Ok». Запустіть "Менеджер плат", вибравши пункт "Інструменти" > "Плата" > "Менеджер плат". У вікні "Менеджер плат" виберіть пункт "esp8266 by ESP8266 Community" зі списку збірок і натисніть на кнопку "Установка". Після успішного встановлення, у списку, навпроти назви "esp8266 by ESP8266 Community" з'явиться фраза "INSTALLED". Закрийте «Менеджер плат», натиснувши кнопку «Закрити».

Для встановлення плати ESP32 в середовищі Arduino IDE виберіть пункт меню «Файл» > «Параметри». У вікні заповніть поле «Додаткові посилання для Менеджера плат», ввівши в нього рядок:
https://dl.espressif.com/dl/package_esp32_index.json і натисніть кнопку «ОК». Запустіть «Менеджер плат», вибравши пункт «Інструменти» > «Плата» > «Менеджер плат». У вікні "Менеджер плат" введіть у пошуку "ESP32" і натисніть кнопку "Установка" для "ESP32 by Espressif Systems". Після успішного встановлення, у списку, навпроти назви "ESP32 by Espressif Systems" з'явиться фраза "INSTALLED". Закрийте «Менеджер плат», натиснувши кнопку «Закрити».

На цьому налаштування Arduino IDE для роботи з платами ESP завершено. Тепер у списку плат є розділ «ESP32» та «ESP8266 Boards».

3. **Для роботи з реальними платами** потрібно вибрати COM-порт, який з'явився у списку доступних портів після підключення плати до комп'ютера через порт USB. Визначити номер порту можна в меню «Комп'ютер» > «Властивості системи» > «Диспетчер пристроїв» (рис.1.1):



Рис.1.1

Якщо після підключення плати новий порт не з'явився, то не встановлено драйвер для чіпа CH340G / CP2104. Встановіть драйвер, назва якого вказано на рис.1.1.

4. Перевіряємо працездатність плати ESP8266 шляхом завантаження скетчу миготіння світлодіодом, вбудованим у плату.

Для перевірки працездатності плати ESP 8266 (NodeMCU 1.0 (ESP-12E Module)) через миготіння світлодіодом, вбудованим у плату. Для роботи з платою 8266 потрібно в пункті меню "Інструменти" > "Плата" > "ESP8266" вибрати плату "ESP 8266 (NodeMCU 1.0 (ESP-12E Module))". Завантажте скетч для миготіння світлодіодом, встановленим на платі з прикладів у меню «Файл» > «Приклади» > «Basics» > «Blink». Перевірте працездатність коду шляхом компіляції.

Натисніть кнопку «Завантаження» у середовищі Arduino IDE. Зачекайте, поки код компілюється і завантажується в плату. Після завантаження на платі повинен заблимати світлодіод.

5. Перевірка працездатності інтерфейсу Wi-Fi плати ESP 8266.

Відкрийте наступний приклад у меню «Файл»>«Приклади»>«ESP8266WiFi» > «WiFi Scan». Проаналізуйте порядок команд у коді для проведення сканування точок доступу Wi-Fi. Перевірте працездатність коду шляхом компіляції. *Натисніть кнопку «Завантаження» у середовищі Arduino IDE. Зачекайте, поки код компілюється і завантажується в плату. Після завантаження відкрийте вікно послідовного COM-порту «Монітор порту» середовища Arduino IDE та налаштуйте швидкість передачі на 115200 бод. Якщо все пройшло, як очікується, після повідомлення компілятора «Завантаження завершено», у вікні послідовного COM-порту буде виведено повідомлення, наведене на рис.1.2.*

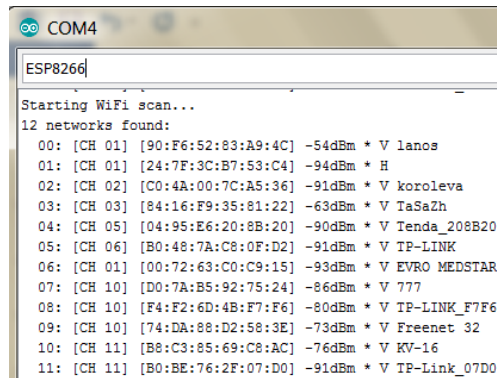


Рис.1.2

6. Створення проекту «Розумна розетка, що синхронізується за NTP-сервером».

Завантаження дати та точного часу з NTP-серверу та його відображення у моніторі послідовного порту для ESP8266. У меню «Інструменти» > «Менеджер плат» виберіть плату ESP8266. Завантажте скетч прикладу вбудованої бібліотеки NTPClient "IsTime.ino" для плати ESP8266, який дозволяє корегувати час та вимикати світлодіод у визначений час. Змініть першу константу у команді «NTPClient timeClient(ntpUDP,"pool.ntp.org", 10800, 60000);» та номер вихідного GPIO для керування світлодіодом змініть на «2», а константи «hour» та «minute» задайте такими, що відповідають бажаному часу вимкнення приладу. Відкрийте монітор послідовного порту і спостерігайте за поточним часом і моментом вимкнення світлодіода.

Переробіть цей код для керування світлодіодом (розеткою при наявності силового комутатора), що підключений до виходу «2» так, щоб він вмикався у визначений Вами час та вмикався через визначений інтервал часу. Збережіть скетч з ім'ям «Surname_Smart.ino» і після успішної компіляції перевірте правильність його роботи.

Якщо Ви використовуєте контролер, то після повідомлення компілятора «Завантаження завершено», у вікні послідовного COM-порту буде виведено повідомлення про поточний час, а світлодіод, який імітує вмикання перемикача, запрацює згідно завдання.

Лабораторне заняття 2. Віддалений запит даних датчика вологості та температури і керування світлодіодом через бот Telegram.

Загальні відомості.

Telegram Messenger – це хмарний сервіс обміну миттєвими повідомленнями та IP-телефонії. Telegram дозволяє створювати ботів, які можна використовувати для надсилання повідомлень. Боти – це сторонні програми, які працюють у Telegram. Користувачі можуть взаємодіяти з ботами, надсилаючи їм повідомлення, команди та запити. Ви керуєте своїми ботами за допомогою HTTPS-запитів до Telegram Bot API. Для взаємодії з ботом Telegram зручно використовувати універсальну бібліотеку ботів Telegram, яка надає досить простий інтерфейс для Telegram Bot API.

Порядок виконання.

1. Для створення коду, що дозволяє запитувати дані датчика через бот Telegram, візьмемо за основу проект, який дозволяє запитувати показання датчика BME280 та надсилати повідомлення до боту Telegram (<https://randomnerdtutorials.com/telegram-request-esp32-esp8266-nodemcu-sensor-readings/>). Створіть **свого бота** Telegram за інструкціями, що містяться на сайті. Поточні показання температури та вологості датчика виводяться у бот за запитом «/readings», який додано до вітального повідомлення. При потребі перевірте правильність роботи створеного проекту.

2. Приклад програмного коду для керування вбудованим у плату діодом з боту Telegram наведено за посиланням: <https://randomnerdtutorials.com/telegram-control-esp32-esp8266-nodemcu-outputs/>.

Використовуючи цей код *доробіть проект* віддаленого запиту показань датчика BME280 так, щоб після запуску програми в чат приходило повідомлення про початок роботи бота з переліком команд управління ботом:

/start - команда, за якою виводиться вітальне повідомлення із зазначенням команд для керування платою;

/led_on и /led_off - команда, за якою плата ESP включає/виключає світлодіод та виводить поточний стан;

/state – команда, за якою ESP посилає в бот поточний стан світлодіода;

/readings - повертає поточні показання температури та вологості датчика BME280.

Збережіть скетч з ім'ям «Surname_Telegram.ino» і після успішної компіляції перевірте його працездатність.

Лабораторне заняття 3. Система сигналізації з відправкою повідомлень про спрацювання датчика руху у бот Telegram.

Загальні відомості.

PIR (Pyroelectric Infrared) датчик по суті складається з чутливого піроелектричного елемента, який вловлює рівень інфрачервоного випромінювання. PIR датчики відмінно підходять для проектів, у яких необхідно визначати наявність або відсутність людини в межах певного робочого простору. Дуже часто використовуються у системах сигналізації. Для забезпечення необхідного кута огляду 110 x 70 ° датчик забезпечений лінзою. У більшості датчиків PIR використовується лінза Френеля, яка виготовляється з пластику, хоча він і додає шумів у вимірювання. PIR-датчики добре працюють в діапазоні хвиль від 8 до 14 мкм, цей діапазон характерний для випромінювань людського тіла. Діапазон чутливості до 6 метрів.

Порядок виконання.

1. Підключіть PIR датчик руху до ESP8266: вихід V підключіть до живлення +3,3V, вихід G – до загальної шини Gnd, а сигнальний вихід S датчика – до GPIO16. Для створення проекту перейдіть до прикладу за посиланням:

<https://randomnerdtutorials.com/telegram-esp8266-nodemcu-motion-detection-arduino/>

Даний проект надсилає у чат з ім'ям бота повідомлення про початок роботи "Bot started up", а потім при виявленні руху щоразу надходитиме повідомлення "Motion detected!". У серійний монітор порту, крім того, буде виводитися інформація про підключення до точки доступу WiFi та IP address, а також повідомлення про спрацювання датчика руху "Motion detected!" та повернення у вихідний стан "Motion ended!".

Збережіть скетч з ім'ям «Surname_ Telegram_PIR.ino» і після успішної компіляції перевірте його працездатність.

Лабораторне заняття 4. Система сигналізації з відправкою повідомлень про спрацювання датчика диму у бот Telegram.

Загальні відомості.

Модуль димового сповіщувача MQ-2 живиться від джерела з напругою 3,...,5В. Напруга на цифровому виході «D0» змінюється з високого рівня на низький при перевищенні певного порогу, що встановлюється змінним резистором модулю.

Порядок виконання.

Підключіть датчик MQ-2 до ESP8266: вихід V підключіть до живлення +3V, вихід G – до загальної шини Gnd, а цифровий вихід «D0» датчика – до GPIO16. Для створення проекту перейдіть до прикладу за посиланням:

<https://randomnerdtutorials.com/telegram-esp8266-nodemcu-motion-detection-arduino/>

При відсутності спрацювання датчика диму MQ-2 на його борту світиться лише зелений світлодіод, який свідчить про подачу живлення до модуля. При спрацюванні датчика диму MQ-2 на його борту вмикається червоний світлодіод, а на цифровому виході «D0» з'являється низький потенціал, за яким мікроконтролер надсилає повідомлення про небезпечне перевищення СО у приміщенні до мобільного додатку «Telegram». Після відновлення вихідного стану датчика диму червоний світлодіод вимикається, що свідчить про те, що сповіщувач перестав виявляти джерело небезпеки.

Перевірте правильність створеного проекту.

Лабораторне заняття 5. Використання хмарного сервісу ThingSpeak для віддаленого добового контролю якості повітря у приміщенні.

Загальні відомості

ThingSpeak - це хмарна платформа IoT, яка дозволяє збирати, відображати та аналізувати потокові дані. Зберігання даних у хмарі забезпечує швидкий доступ до них. ThingSpeak дозволяє інженерам та вченим будувати IoT-системи без налаштування серверів та розробки веб-додатків.

Порядок виконання роботи

1. Передавання вимірів температури до хмарного сервісу ThingSpeak. Для дистанційного моніторингу стану повітря у приміщенні будемо використовувати датчик температури та вологості BME280 . Проект будемо реалізовувати на базі

мікроконтролера ESP8266 з графічним відображенням зміни контрольованих параметрів у хмарній платформі IoT “ThingSpeak”.

Необходимо зайти на сайт <https://thingspeak.com/> та створити обліковий запис. Після реєстрації Вам потрібно буде створити новий канал, користуючись інструкціями за посиланням <https://randomnerdtutorials.com/esp8266-nodemcu-thingspeak-publish-arduino/>.

Після створення каналу буде визначено ID каналу і буде доступна закладка ключів API. Натисніть "Ключ API" та скопіюйте ключ API для запису даних. Налаштуйте середовище Arduino IDE на роботу з платою ESP8266. Для підключення до системи ThingSpeak підключіть бібліотеку ThingSpeak.h. Завантажте програмний код передачі даних вимірів температури у хмарний сервіс ThingSpeak ESP8266 «Publish Sensor Readings to ThingSpeak» до програмного середовища Arduino IDE. Замініть поля CH_ID та WRITE_APIKEY своїми даними створеного каналу ThingSpeak.

У цій програмі функція ThingSpeak() повертає код помилки, який міститься у змінних httpCode1. Якщо повертається код 200, то це означає, що дані успішно передані та отримані. Так само необхідно, щоб дані які надсилаються, приходили з паузами мінімум в 20 секунд, інакше система їх ігноруватиме.

Збережіть скетч «**ThingSpeak.ino**» та проведіть його перевірку шляхом компіляції.

2. Передавання вимірів температури та вологості до хмарного сервісу ThingSpeak

Для додавання ще одного каналу до раніше створеного проекту можна скористатися прикладом “Sending Multiple Fields (Temperature, Humidity, and Pressure)” передавання даних у хмарну платформу IoT “ThingSpeak” за посиланням: <https://randomnerdtutorials.com/esp8266-nodemcu-thingspeak-publish-arduino/>.

Зайдіть у свій акаунт ThingSpeak та створіть додатковий канал для виведення вимірів вологості та тиску. Додайте до раніше створеного скетчу «**ThingSpeak.ino**» необхідні рядки для читання крім температури ще і значення вологості та тиску.

Проведіть компіляцію створеного скетчу "**Surname_ TS_ Quality.ino**" та після успішної компіляції перевірте працездатності розробленого проекту.

Якщо Ви використовуєте мікроконтролер, відкрийте свій обліковий запис «Thingspeak Account» і перейдіть до приватного перегляду. Дані вимірів у «Thingspeak Account» повинні співпасти з переданими.

Перелік літератури

1. Андреев О.В. Архитектура і технології IoT: конспект лекцій [Електронне видання] / О.В. Андреев . – Житомир: Державний університет "Житомирська політехніка", 2022. - 128 с.
2. Архитектура і технології IoT: Методичні вказівки до виконання лабораторних робіт [Електронне видання] / О.В. Андреев . – Житомир: Державний університет "Житомирська політехніка", 2022. - 39 с.