

УДК 004:378.147.88

Цирульник Сергій Михайлович

Кандидат технічних наук, доцент кафедри комп'ютерних наук та економічної кібернетики

Вінницький національний аграрний університет, м. Вінниця, Україна

sovnm@ukr.net

ORCID: 0000-0002-5703-9761

МОБІЛЬНІ ДОДАТКИ ТА ОНЛАЙН ПЛАТФОРМИ МОНІТОРИНГУ ДАНИХ WI-FI МЕТЕОСТАНЦІЇ

Анотація. Світ швидко покривається мережами, які дозволяють цифровим пристроям взаємоз'єднуватися та передавати дані. Завдяки мережам можливе підключення всіх мобільних пристроїв, електронних датчиків, електронних вимірювальних приладів, медичних приладів та датчиків. Вони відслідковують, обмінюються, оцінюють та в деяких випадках автоматично коригують дані, які збираються та передаються. Поняття «Інтернет речей» є комплексним та має кілька рівнів: кінцеві пристрої (датчики, виконавчі пристрої), транспортний рівень (телекомунікаційне середовище, що включає дротові та бездротові мережі) та рівень роботи з даними (збір, зберігання та обробку). Ринкове середовище формує вимоги до молодих фахівців, а конкуренція між закладами вищої та фахової передвищої освіти надає можливість підготувати висококваліфікованого фахівця, який здатний вивчати та створювати сучасне апаратно-програмне забезпечення функціонування розумних електронних пристроїв та систем, які є вузлами мережі «Інтернет речей». У статті розглядаються питання, які пов'язані з особливостями створення простих пристроїв в рамках концепції Internet of Things на базі популярного Wi-Fi-модуля ESP8266 та впровадженням цих досліджень у освітній процес. Розкрито технічні можливості, особливості підключення та взаємодії модуля ESP8266 для моніторингу метеоданих. Показана організація доступу модуля до мережі Інтернет, відправлення даних та їх моніторинг з використанням популярного мобільного додатку Blynk, Virtuino та хмарного IoT-сервісу ThingSpeak. Проаналізована їх робота в некомерційних задачах та зручність користування освітнім закладом. У статті наводяться вихідні коди програм для Wi-Fi-модуля ESP8266 з цифровим датчиком BME280.

Ключові слова: Інтернет речей; модуль IoT; онлайн сервіс; мобільний додаток; вільне програмне забезпечення; моніторинг даних; метеостанція

Постановка проблеми. Internet of Things (IoT) це галузь, що стрімко зростає, до якої, відносяться такі технології, як «Розумний та безпечний будинок», «Інтелектуальне місто», «Віртуальний маркетинг», «Комунікаційна та навігаційна техніка», «Віртуальні інженерні виробництва». Такі технології вже охоплюють практично кожний сегмент у сфері промисловості, бізнесу, охорони здоров'я та споживчих товарів. IoT – одна з головних тенденцій IT-індустрії. Розробкою та проектуванням апаратно-програмного забезпечення для Internet of Things займаються embedded-програмісти.

Embedded-програміст – це фахівець, який займається розробкою, супроводом, тестуванням програмно-апаратних засобів. Ця спеціальність лежить на межі апаратної інженерії та програмування. Embedded-розробники працюють не тільки з Software, а й з Hardware. Embedded-розробник знайомий з базовими поняттями електроніки, схемотехніки, теорії обробки сигналів, математики, алгоритмів, мови програмування C++. Тому актуальним є аспекти практичної підготовки висококваліфікованого embedded- фахівця.

Завданням творчого конкурсу десятої Всеукраїнської олімпіади з радіоелектроніки для студентів коледжів та технікумів України була розробка автономної Wi-Fi метеостанції на модулі ESP8266 з можливістю моніторингу даних на мобільному пристрої (смартфон, планшет). Однією з головних задач для команди Вінницького технічного коледжу став вибір безкоштовного мобільного додатку або онлайн платформи з можливостями реалізувати свій власний інтерфейс.

Мета статті – дослідити сучасні тенденції застосування технологій інтернет речей засобами вільного програмного забезпечення для моніторингу даних Wi-Fi метеостанції на модулі ESP8266 мобільними додатками та онлайн платформами.

Аналіз досліджень та публікацій. Українськими університетськими командами Національного аерокосмічного університету ім. М. Є. Жуковського «Харківський авіаційний Інститут», Одеського національного політехнічного університету, Запорізького національного технічного університету, Східноукраїнського національного університету ім. Володимира Даля, Чернівецького національного університету імені Юрія Федьковича за підтримки колег з академічних закладів країн ЄС, що входять до консорціуму проекту ALIOT підготовлені наукові та методичні матеріали в рамках проекту Internet of Things: Emerging Curriculum for Industry and Human Applications / ALIOT, 573818-EPP-1-2016-1-UK-EPPKA2-CBHE-JP, 2016-2019, що фінансується програмою ЄС ERASMUS [1]. Однак, у цих матеріалах не достатньо розкриті питання, які мобільні додатки та онлайн платформи можна використовувати для своїх інженерних досліджень та практичних конструкцій студентами, аматорами, аспірантами. У роботі [2] розглянуті мобільні додатки Blynk, RemoteXY, Cayenne, Virtuino. Одним з відомих дослідників є Шварц М. Так у роботі [3] наводяться приклади застосувань онлайн сервісів Dweet.io, Freeboard.io, Temboo, IFTTT, aREST. Використання хмарних IoT-сервісів: Narodmon, ThingSpeak, Xively, Weaved, Blynk, Wylidrin наводяться у роботі [4].

Виклад основного матеріалу. До складу автономної Wi-Fi метеостанції на модулі ESP8266 входить модуль WeMos D1 mini (NodeMCU або WittyCloud), комбінований датчик температури та вологості BME280 (HDC1080 або DHT22), Li-Ion акумулятор, сонячна батарея та контролер заряду на мікросхемі TP4056. Структурна схема метеостанції наведена на рис.1.

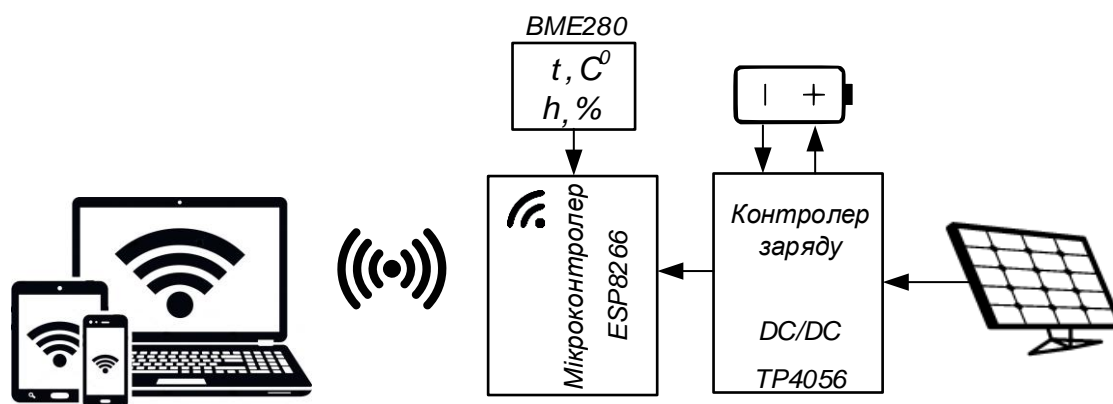


Рис. 1. Структурна схема Wi-Fi метеостанції

Якщо апаратна частина Wi-Fi метеостанції більш уніфікована і може відрізнитись типом датчиків, то програмна частина більш гнучка і від її вибору залежить зручність користування отриманими даними. Програмне забезпечення складається з програми для мікроконтролера ESP8266 та передбачає її взаємозв'язок з мобільним сервісом або онлайн сервісом. При виборі потрібно визначитись який функціонал хоче отримати користувач: модуль Wi-Fi є точкою доступу і дані можна отримати тільки встановивши зв'язок з модулем по безпроводному каналу; модуль приєднаний до мережі Wi-Fi і передає дані до мобільного додатку та/або в хмару, що дозволить здійснити відділений моніторинг даних; інформація про температуру та вологість є приватними чи можна ними ділитися; моніторинг даних тільки з мобільних пристроїв та/або персональних комп'ютерів; як часто отримувати дані; потрібно зберігати дані та як довго. І остання

вимога – мобільний додаток та онлайн платформа має бути безкоштовна або умовно-безкоштовна та дозволяє встановити власний логотип або емблему.

Одним з популярних додатків для IoT є Blynk [2, 5]. Основна мета Blynk – створення доступної платформи для бездротового управління електронними пристроями смартфоном. Першим кроком є встановлення додатку Blynk на смартфон з Play Market або App Store. Після установки мобільного додатка Blynk необхідно зареєструватися і вибрати пристрій, яким необхідно управляти (ESP8266). Далі, щоб не втратити, відправляємо собі на електронну пошту унікальний маркер авторизації (Token Auth), який використовується в скетчі для автентифікації на серверах Blynk.

У вікні, створеного проекту в правому верхньому кутку 3 кнопки: глобальні настройки проекту, додавання елементів (віджетів) на форму і запуск. Необхідно на форму додатка додати елементи управління і зробити їх налаштування. На рис.2 наведений інтерфейс проекту VTC у додатку Blynk.

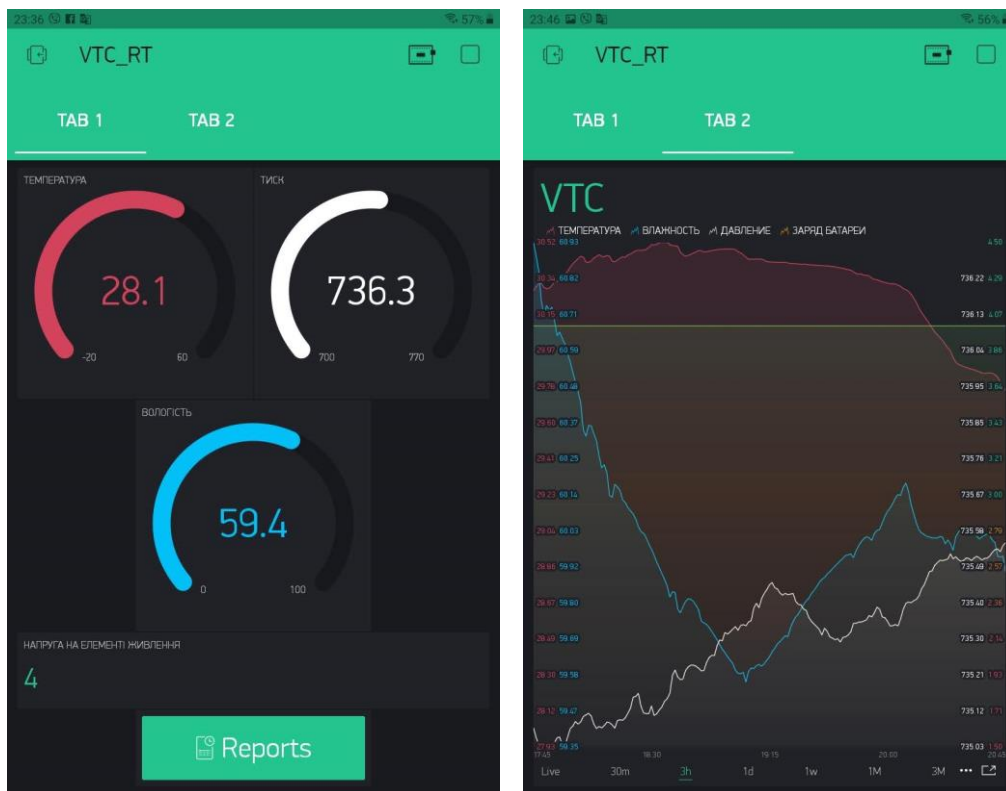


Рис. 2. Інтерфейс проекту VTC у додатку Blynk

У проекту VTC використовувалися такі елементи:

- три візуальні шкали (Gauge) для відображення значення температури, вологості, тиску (Virtual Pins 0, 1, 2);
- цифрова шкала (Value Display) для відображення напруги акумулятора (Virtual Pins 3);
- чотири графіка (Chart) температури, вологості, тиску, напруги акумулятора.

На рис. 3 наводяться скріншоти, що ілюструють настройку деяких елементів програми.

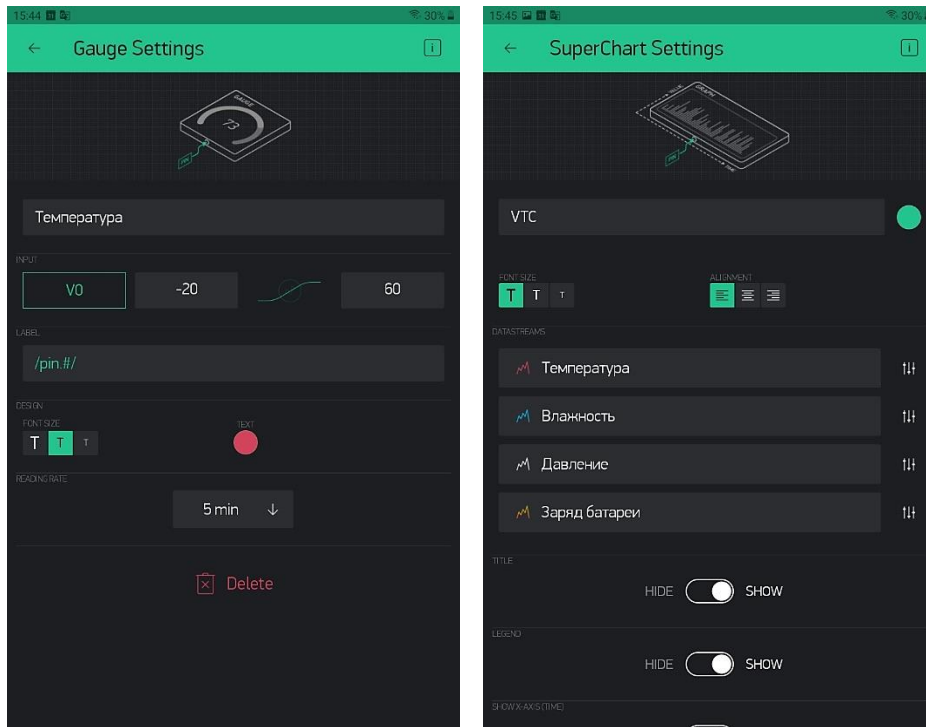


Рис. 3. Налаштування віджетів Gauge (Температура) та Graph

Головний плюс розробки програмного забезпечення для ESP8266 в середовищі Arduino IDE – можливість поєднати в одному скетчі абсолютно різні бібліотеки. Для того щоб використовувати ESP8266 з бібліотекою Arduino, треба використовувати Arduino IDE з підтримкою плат ESP8266. Для роботи Arduino IDE з Blynk треба встановити бібліотеку Blynk [6]. Також необхідно встановити бібліотеку Adafruit для роботи з датчиком BME280. Скетч програми:

```
#include <ESP8266WiFi.h>
#define BLYNK_PRINT Serial // Enables Serial Monitor
#include <BlynkSimpleEsp8266.h>
#include <Adafruit_Sensor.h>
#include <Wire.h>
#include <Adafruit_BME280.h>

unsigned int raw=0;
float volt=0.0;
Adafruit_BME280 bme; // I2C (SDA - D2; SCL - D1)
char auth[] = "*****";
char ssid[] = "*****";
char pass[] = "*****";

void setup() {
  Serial.begin(115200);
  while (!bme.begin(0x76)){
    Serial.println("BME280 not find, check wiring!");
    delay (1000);};
  Blynk.begin(auth, ssid, pass);
  pinMode(A0, INPUT);
```

```

raw = analogRead(A0);
volt= raw *4,2/1023;}

void loop() {
  Blynk.run();
  Send();// відправлення даних та сон
}

void Send(){
  float h = bme.readHumidity(); // значення вологості
  float t = bme.readTemperature(); // значення температури
  float p = (bme.readPressure() / 133.3223684F); // значення тиску

  Blynk.virtualWrite(V0, t); //вивід значення температури до Blynk
  Blynk.virtualWrite(V1, p); // вивід значення тиску до Blynk
  Blynk.virtualWrite(V2, h); // вивід значення вологи до Blynk
  Blynk.virtualWrite(V3, volt); // вивід значення заряду батареї до Blynk
  delay (300);
  ESP.deepSleep(20e6); //глибокий сон на 20 секунд (перемичка між D0 та Reset)
}

```

Основною бібліотекою для роботи з Blynk є бібліотека BlynkSimpleEsp8266.h. Функція Blynk.begin (auth, «ssid», «pass») отримує параметри підключення до Wi-Fi точки – її SSID та пароль, а також Token Auth пристрою для підключення до Blynk server.

Процес підключення до Blynk server починається тільки після того, як в програмі буде вперше викликана функція Blynk.run ().

Однією з особливостей Blynk є використання віртуальних контактів для передачі різної інформації між мобільним додатком та пристроєм. Віртуальні контакти не мають прямого зв'язку з портами ESP8266 і використовуються переважно для взаємодії з інтерфейсом мобільного додатка.

У кодї скетчу відправлення даних до віртуального контакту з номером n відбувається за допомогою коду Blynk.virtualWrite(n, value).

Основною бібліотекою для роботи з датчиком BME280 є бібліотека Adafruit_BME280.h. Команда Adafruit_BME280 bme створює новий екземпляр класу Adafruit_BME280. Функція bme.begin(0x76) виконує ініціалізацію датчика. Функції bme.readTemperature(), bme.readHumidity(), bme.readPressure() дозволяють зчитати значення температури, вологості, тиску, що вимірює датчик.

ESP8266 – це досить енергоємний пристрій, тому у скетчі використовується функцію глибокого сну ESP.deepSleep().

Сам по собі сервіс Blynk є дуже зручним і простим для освоєння інструментом, який легко інтегрується з модулем ESP8266 в конкретний проект з найменшими зусиллями. Мінусами використання такого рішення є менша гнучкість в налаштування проекту під себе (не можна додати власний логотип); для реалізації інших проектів, цей потрібно розібрати або докупити частину «енергії» (концепція умовно-безкоштовного додатку); не можна здійснювати моніторинг даних з комп'ютера та є певні складності зробити отримані дані публічними.

Сервіс ThingSpeak.com дозволяє публікувати дані від датчиків на вебсайт та виводити їх на графік з позначкою дати та часу. Це дозволяє отримати доступ до своїх даних з будь якої точки світу [4].

Спочатку необхідно створити обліковий запис на ThingSpeak. Далі створюємо

новий канал у своєму обліковому запису ThingSpeak (рис. 4).

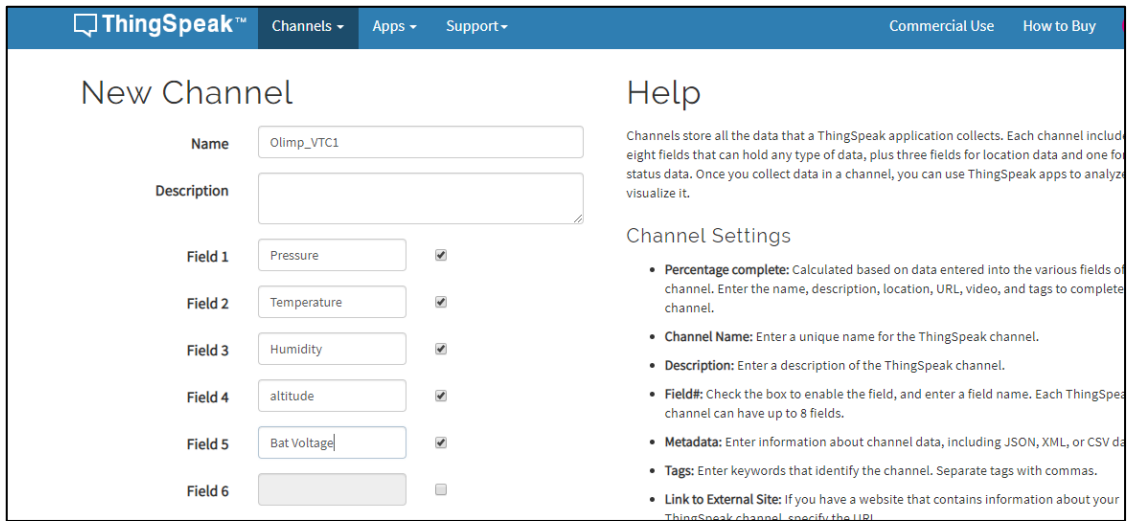


Рис. 4. Створення нового каналу Olimp_VTC1 в обліковому запису ThingSpeak

У своєму обліковому запису ThingSpeak вибрати “Channel”, а потім “My Channel”. Клацніть на назві каналу. Виберіть вкладку “API Keys” та скопіюйте “Write API Key” (рис. 5).

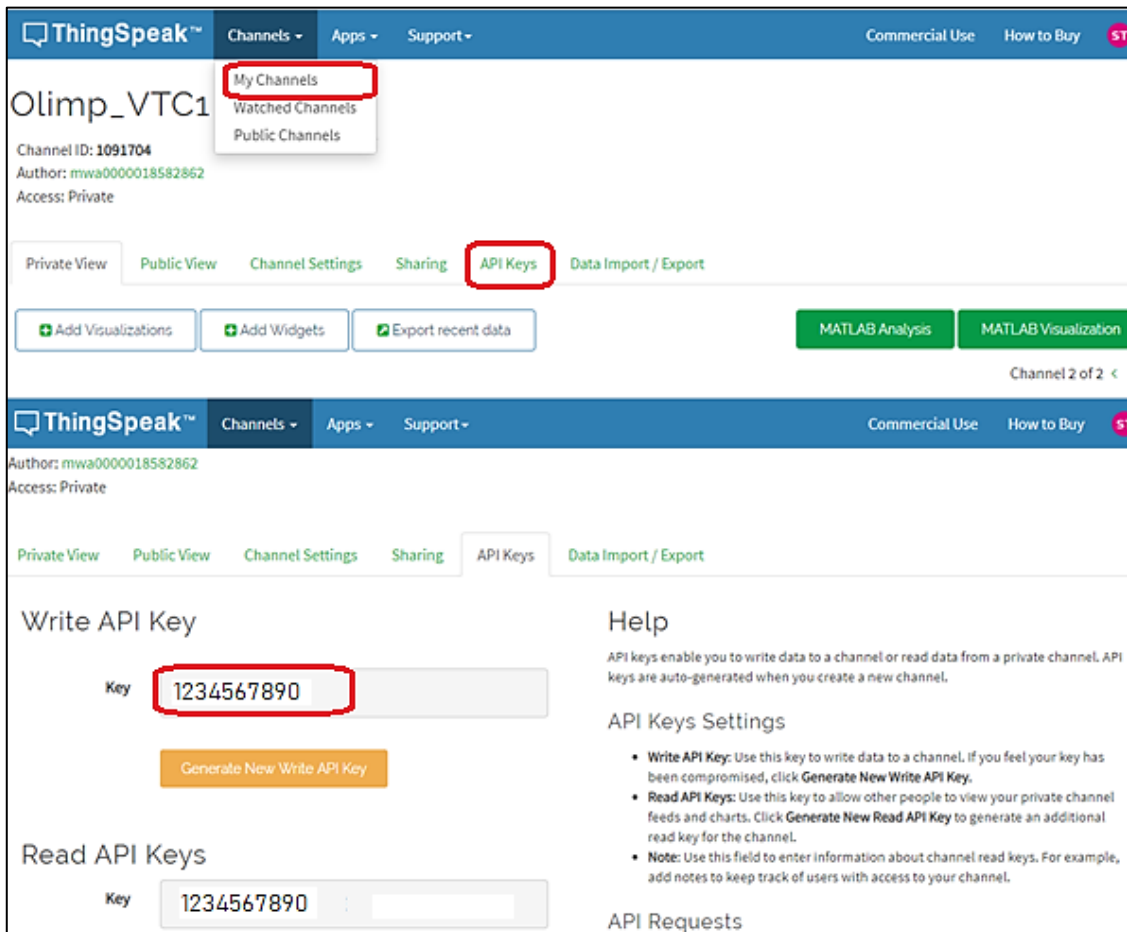


Рис. 5. Визначення Write API Key каналу Olimp_VTC1

На вкладці Private View додаємо віджети (Widgets) та графіки (Visualizations) для зручності моніторингу даних. Скетч програми:

```

const bool bme280Debug = 0; // controls serial printing
const long measurementInterval = 30000; // measurement interval in ms
const String App = "Thingspeak";
#include <ESP8266WiFi.h>
#include <farmerkeith_BMP280.h>
#include <Wire.h>
#include "PTHsleep.h" // tab file

bme280 bme0 (0, bme280Debug) ; // creates object bme0 of type bme280, base address
char ssid[] = "*****"; // WiFi Router ssid
char pass[] = "*****"; // WiFi Router password

// Thingspeak Write API
const char* server = "api.thingspeak.com";
const char* api_key = "*****"; // API write key

void setup() {
  Serial.begin(115200);
  WiFi.begin(ssid, pass);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print("-");
  }
  Serial.println("WiFi connected");

  unsigned long baseEventTime = millis();
  byte temperatureSamples = pow(2, osrs_t - 1);
  byte pressureSamples = pow(2, osrs_p - 1);
  byte humiditySamples = pow(2, osrs_h - 1);
  Serial.print ("Temperature samples=");
  Serial.print (temperatureSamples);
  Serial.print (" Pressure samples=");
  Serial.print (pressureSamples);
  Serial.print (" Humidity samples=");
  Serial.println (humiditySamples);

  bme0.begin(osrs_t, osrs_p, 1, 0, 0, 0, osrs_h);
  // parameters are (osrs_t, osrs_p, mode, t_sb, filter, spi3W_en, osrs_h)

  measurementEvent();
  bme0.updateF4Control16xSleep(); goToSleep();
}

void loop() {
}

```

Бібліотека – це частина програмного забезпечення, яка надає інтерфейс

прикладного програмування (API) для програміста, який може використовувати можливості пристрою без необхідності розбиратися з усіма подробицями. API-інтерфейс повинен бути простим для початківців з простими вимогами до початку роботи, забезпечуючи при цьому повне використання можливостей пристрою. У проєкті з Blink використовувалася бібліотека `Adafruit_BME280.h`. Однак, бібліотека не підтримує здатність модуля зберігати декілька біт даних, поки пристрій і його мікропроцесор керування знаходяться в сплячому режимі. Тому у проєкті з Thingspeak використовується бібліотека `farmerkeith_BMP280.h`. Для створення програмного об'єкта `BMP280` використовується команда `bme280 bme0 [6]`. Перед використанням необхідно прочитати параметри калібрування з датчика і налаштувати його для будь-якого режиму вимірювання. Ініціалізація датчика виконується командою `bme0.begin (osrs_t, osrs_p, 1, 0, 0, 0, osrs_h)`, де `osrs_t = 5` (16 вимірювань температури), `osrs_p = 5` (16 вимірювань тиску), `mode = 1` (нормальний), `t_sb = 0` (сон 0,5 мс між набори вимірювань), `filter = 0` (фільтрація відсутня), `spiw_en = 0` (SPI відключений, використовується I2C), `osrs_h = 5` (16 вимірів вологості) [6]. Якщо набір вимірювань включає в себе кілька вимірів з використанням нормального режиму, `BME280` необхідно перевести в сплячий режим. Це можна зробити за допомогою команди `bme0.updateF4Control16xSleep ()`. Використання команди дозволяє зберігати 12 біт інформації в регістрах `BME280 F4` і `F5`. Дані, що зберігаються в регістрах `F4` і `F5` відновлюються, коли мікроконтролер `ESP8266` активується кожні 30 секунд.

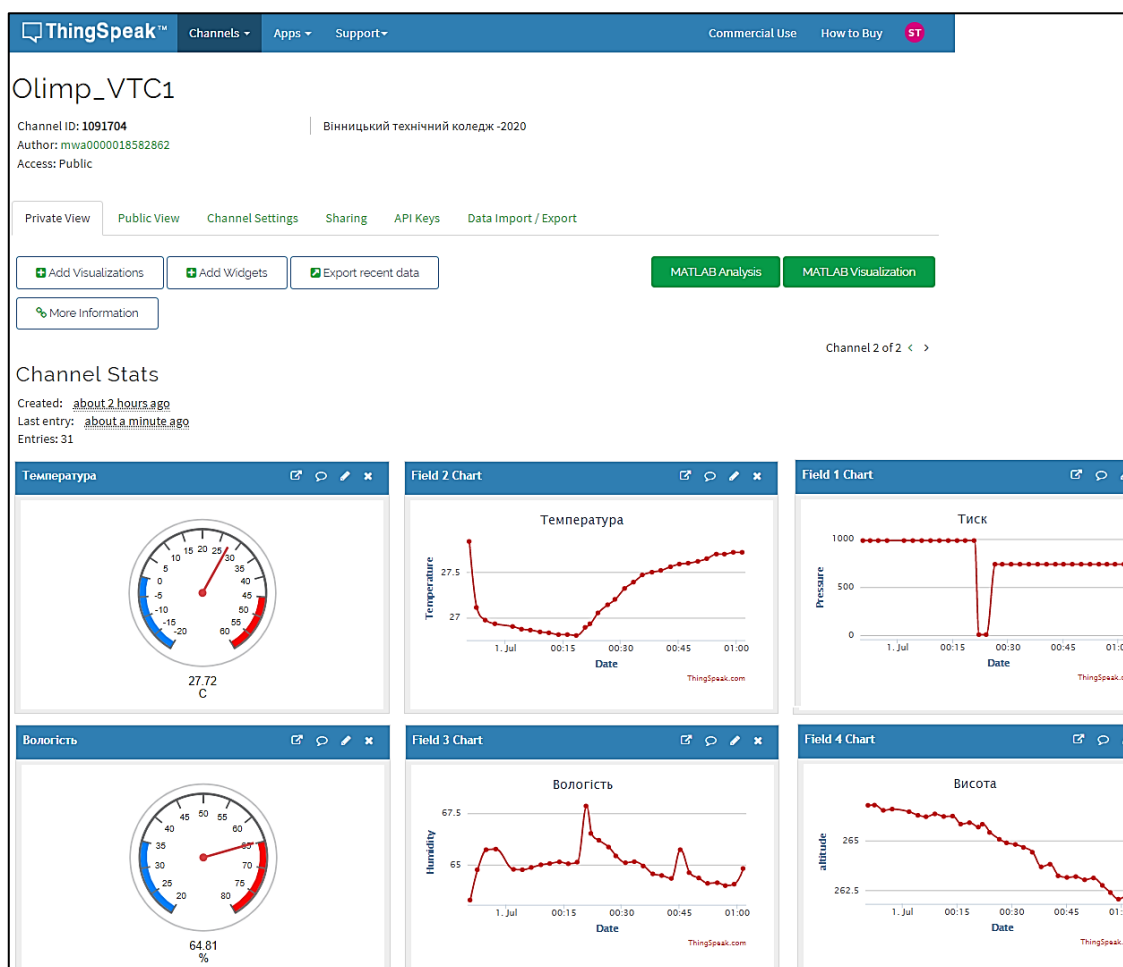


Рис. 6. Моніторинг даних WI-FI метеостанції в онлайн сервісі ThingSpeak

Існує зв'язок між атмосферним тиском та висотою над рівнем моря. Основні функції читання температури, тиску, вологості, розрахунок висоти та передавання їх до сервісу ThingSpeak знаходяться в бібліотеці PTHsleep.h.

Результати моніторингу даних Wi-Fi метеостанції в онлайн сервісі ThingSpeak наведені на рис. 6. Результати моніторингу можна поділитися та зробити проект публічним. Це робиться на вкладці Public View [7].

Сервіс ThingSpeak безкоштовний, зручний у використанні, тому що дозволяє отримати доступ до даних з будь-якого мобільного пристрою та комп'ютера, що має доступ до Інтернету з будь якої точки світу. Сервіс дозволяє додати карту, де можна вказати точне місце встановлення Wi-Fi метеостанції, однак, не вдалося додати власний логотип.

Мобільний додаток Virtuino як і Blynk використовує технологію додатки в додатку. Це означає, що проект залишається в «пісочниці» Virtuino і не можуть існувати окремо від Virtuino. У Virtuino розробник може скористатися сервісом ThingSpeak як проміжний сервер. Комунікація між додатком та метеостанцією здійснюється за допомогою веб-сервера на мобільному пристрої. Для створення мобільного додатку моніторингу даних необхідно встановити додаток Virtuino. У контекстному меню додатку вибрати режим «Server Setting», натиснути кнопку «+» та у спадному меню вибрати «Додати сервер IoT (канал Thingspeak)» (рис. 7)

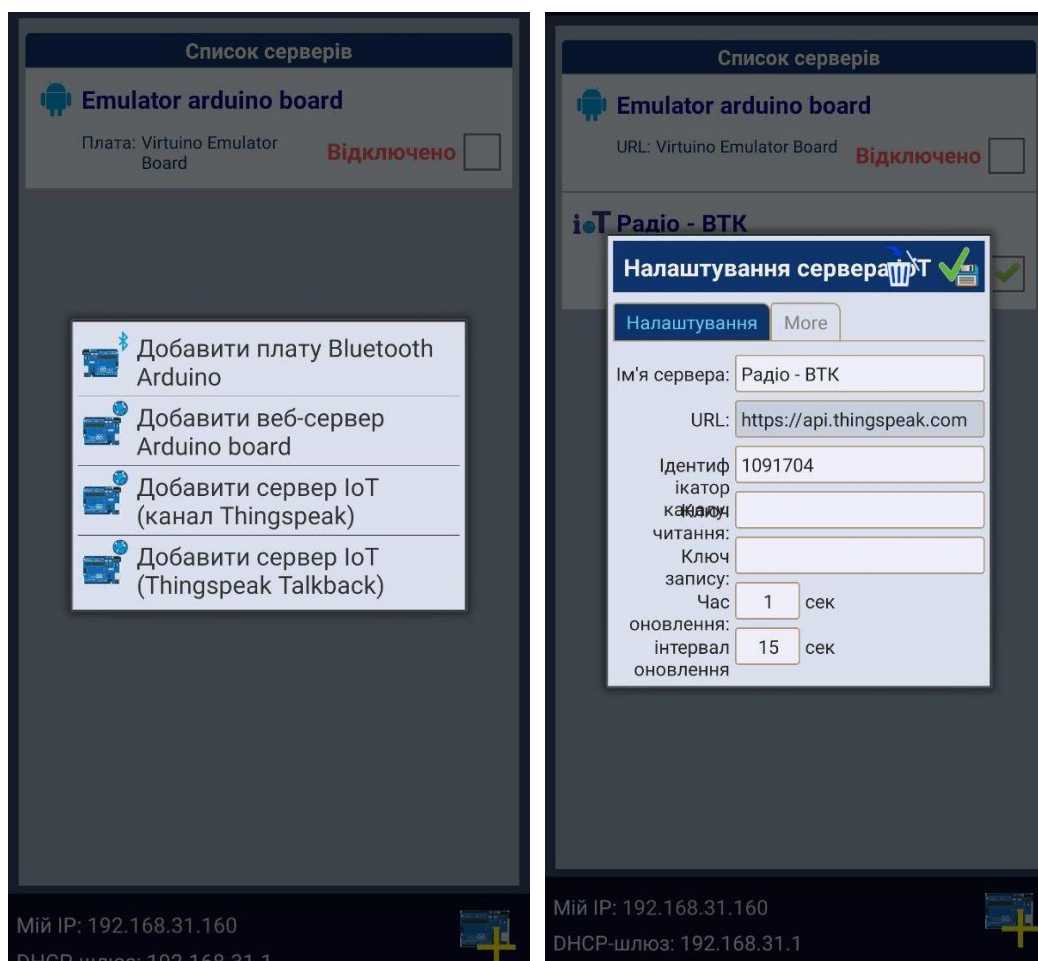


Рис. 7. Налаштування роботи Virtuino з сервером ThingSpeak

Наступним кроком є додавання на робоче поле проекту по аналогії з Blynk

аналогових шкал, кольорових шкал, цифрових індикаторів, графіків, текстових полів, картинок та налаштування їх (рис. 8).

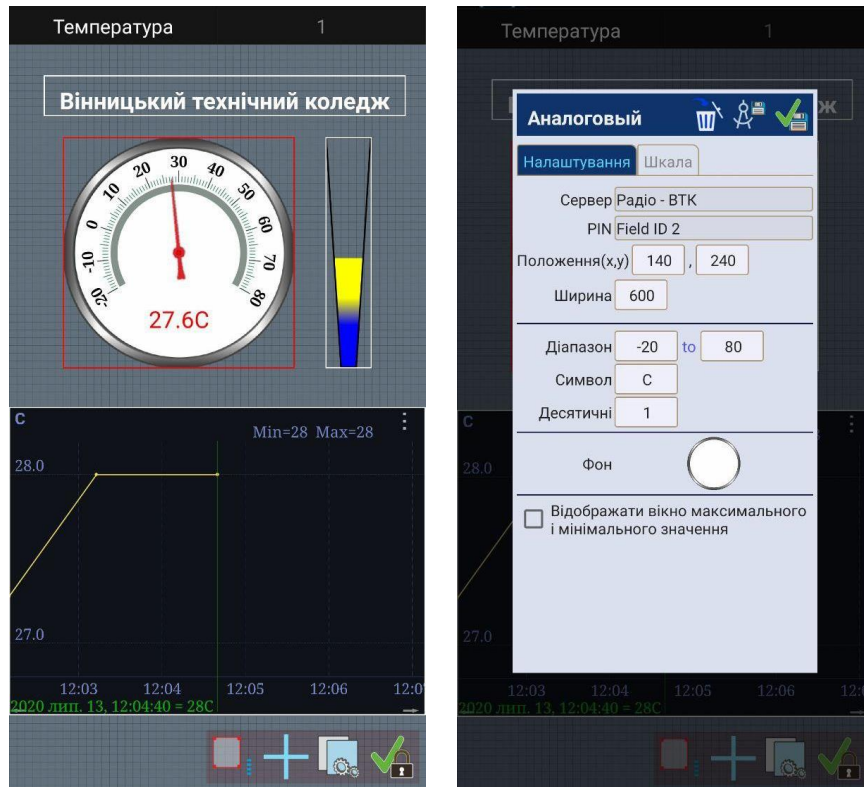


Рис.8. Проектування інтерфейсу мобільного додатку у Virtuino

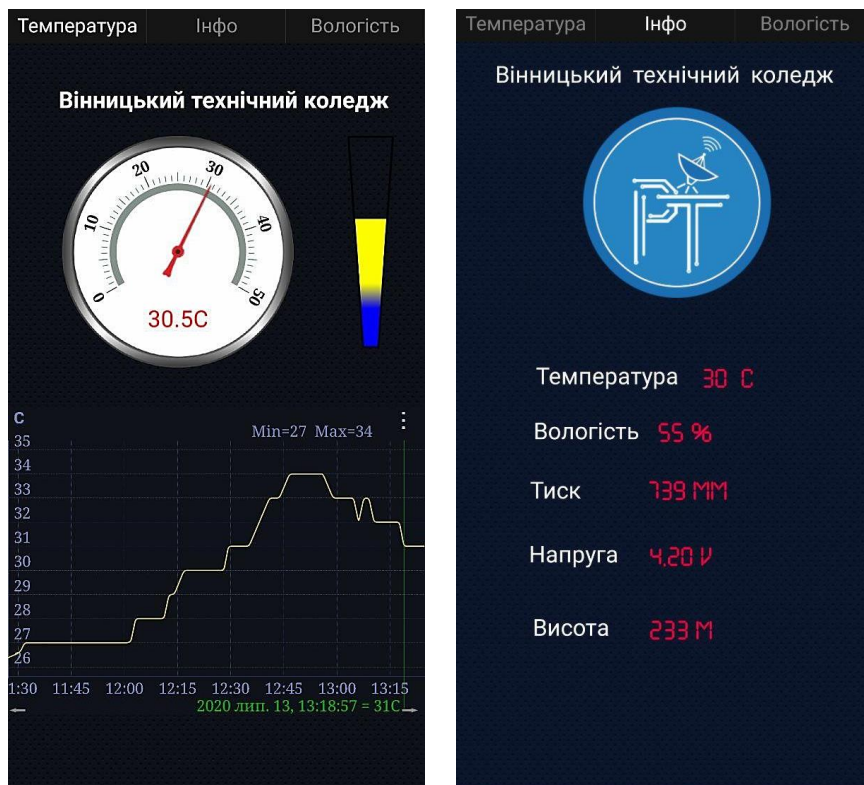


Рис. 9. Інтерфейс мобільного додатку моніторингу даних Wi-Fi метеостанції

Додаток моніторингу даних Wi-Fi метеостанції містить 3 вкладки: температура, вологість та інфо (рис. 9). Мобільний додаток Virtuino дозволив спроектувати зручний інтерфейс програми моніторингу даних з можливістю встановити свій власний логотип. Крім того програма дозволяє зберегти цей проект як окремий файл з розширенням vrt і поділитись ним з усіма бажаними. Для перегляду даних моніторингу на будь-якому смартфоні або планшеті достатньо встановити мобільний додаток Virtuino та відкрити в ньому файл.

На даний час у Вінницькому технічному коледжі студентами ведуться роботи над реалізації проекту моніторингу даних мікроклімату навчальних приміщень освітнього закладу з метою ефективного використання ресурсів під час опалювального сезону.

Висновки. Для проектів Internet of Things постає питання вибору мобільного додатку або онлайн сервісу. Для некомерційних проектів найбільш оптимальною є зв'язка он-лайн сервісу ThingSpeak та додатку Virtuino для Android платформи. Вони дозволять отримати приватний або публічний онлайн доступ з будь-якого пристрою, що має вихід до мережі Інтернет та можливість мати свій власний інтерфейс мобільного додатку. Такий зв'язок дозволить не тільки виконувати моніторинг даних, а ще і віддалене або автоматичне керування об'єктом.

Ринкове середовище формує вимоги до молодих фахівців, а конкуренція між закладами вищої та фахової передвищої освіти надає можливість підготувати висококваліфікованого фахівця. Досвід набутий при реалізації матеріалів статті є корисним для професійного зростання та отримання практичних навичок з розробки та проектування апаратно-програмного забезпечення для embedded систем з використанням Internet of Things.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Кафедра комп'ютерних систем, мереж і кібербезпеки. Навчальні матеріали. URL: <https://csn.khai.edu/nauka/projects/erasmus-aliot/navchalni-materiali-erasmus-aliot> (дата звернення: 03.09.2020).
2. Цирульник С. М., Роптанов В. І. Як долучитись до Internet of things. Збірник праць 11 МНПК «Інтернет-освіта-наука 2018». Вінниця. ВНТУ. 2018. С. 80-82.
3. Шварц М. Интернет вещей с ESP8266. Санкт-Петербург: БХВ-Петербург, 2019. 224 с.
4. Петин В. А. Arduino и Raspberry Pi в проектах Internet of Things. Санкт-Петербург: БХВ-Петербург, 2019. 432 с.
5. Практический опыт использования Blynk для датчика CO2. Часть 1. URL: <https://habr.com/ru/post/122601> (дата звернення: 25.05.2020).
6. Библиотека для BMP280 и BME280: 7 шагов – 2020 – How ToDo Well. URL: <https://ru.howtodowell.com/25114-Library-for-BMP280-39> (дата звернення: 25.05.2020).
7. ThingSpeak. Channels. Olimp_VTC1. URL: <https://thingspeak.com/channels/1091704> (дата звернення: 05.08.2020).
8. Shevchuk, A., Romanov, O. Temperature monitoring system in IOT network based on ESP8266 microcontroller and ThingSpeak service. Збірник матеріалів Міжнародної науково-технічної конференції «Перспективи телекомунікацій». 2020. URL: <http://conferenc.its.kpi.ua/proc/article/viewFile/202173/202083> (дата звернення: 10.06.2020).
9. Nasution, T., Muchtar, A., Seniman, S., Siregar, I. Monitoring temperature and humidity of server room using Lattepanada and ThingSpeak. In Journal of Physics: Conference Series. Vol. 1235, №. 1, 2019.

MOBILE APPLICATIONS AND ONLINE WI-FI MONITORING PLATFORMS OF WEATHER STATIONS

Tsyurulnyk Serhii

PhD (technical sciences), Assistant Professor of the Department of Computer Science and Economic Cybernetics, Vinnytsia National Agrarian University, Vinnytsia, Ukraine

sovnm@ukr.net

ORCID: 0000-0002-5703-9761

Abstract. Networks that allow digital devices to connect and transmit data are covering the world fast. Thanks to the networks, it is possible to connect all mobile devices, electronic sensors, electronic measuring devices, medical devices and sensors. They track, share, evaluate, and in some cases automatically adjust the data that is collected and transmitted.

The concept of "Internet of Things" is complex and has several levels: end devices (sensors, actuators), transport layer (telecommunications environment, including wired and wireless networks) and the level of data processing (collection, storage and processing). The market environment creates requirements for young professionals, and competition between higher education institutions and vocational education institutions provides an opportunity to train a highly qualified specialist who can study and create modern hardware and software for smart electronic devices and systems that are nodes of the Internet of Things network. The article deals with issues related to the peculiarities of creating simple devices within the concept of the Internet of Things based on the popular Wi-Fi module ESP8266 and the introduction of this research into the educational process. The technical possibilities, features of connection and interaction of the ESP8266 module for meteorological monitoring are revealed. The organization of the module's access to the Internet, data sending and their monitoring using the popular mobile applications Blynk, Virtuino and the cloud IoT service ThingSpeak is shown. Their work in non-commercial tasks and ease of use for educational institutions are analyzed. The article provides the source codes of programs for the Wi-Fi module ESP8266 with a digital sensor BME280.

Keywords: augmented reality; AR-platform; 3D model; educational process; mobile app; free software.

REFERENCES (TRANSLATED AND TRANSLITERATED)

1. Department of Computer Systems, Networks and CyberSecurity (2020). Training materials. September 3, 2020. <https://csn.khai.edu/nauka/projects/erasmus-aliot/navchalni-materiali-erasmus-aliot> (in Ukrainian).
2. Tsyurulnyk, S. & Roptanov, V. (2018). How to join the Internet of things. In Proceedings of the eleventh international scientific-practical conference INTERNET-EDUCATION-SCIENCE-2018, Vinnytsia, 22-25 May, 2018: 80-82, Vinnytsia (in Ukrainian).
3. Schwartz, M. (2019). Internet of Things with ESP8266. Saint Petersburg: BHV-Peterburg. 224 p. (in Russian).
4. Petin, V. (2019). Arduino and Raspberry Pi in the Internet of Things projects. Saint Petersburg: BHV-Peterburg. 432 p. (in Russian).
5. Practical experience of using Blynk for a CO2 sensor (2020). Part 1. May 25, 2020. <https://habr.com/ru/post/122601> (in Russian).
6. Library for BMP280 and BME280 (2020). 7 Steps - How To Do Well. May 25, 2020. <https://ru.howtodowell.com/25114-Library-for-BMP280-39> (in Russian).
7. ThingSpeak. Channels. (2020). Olimp_VTC1. August 5, 2020. <https://thingspeak.com/channels/1091704> (in English).
8. Shevchuk, A. & Romanov, O. (2020). Temperature monitoring system in IOT network based on ESP8266 microcontroller and ThingSpeak service. In Proceedings of the eleventh international scientific-practical conference «Modern challenges in telecommunications», Kyiv, 13-17 April 2020, Kyiv (in Ukrainian).
9. Nasution, T., Muchtar, M., Seniman, S., & Siregar, I. (2019, June). Monitoring temperature and humidity of server room using LattePanda and ThingSpeak. In Journal of Physics: Conference Series (Vol. 1235, No. 1, p. 012068). IOP Publishing.