

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 відображати 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.

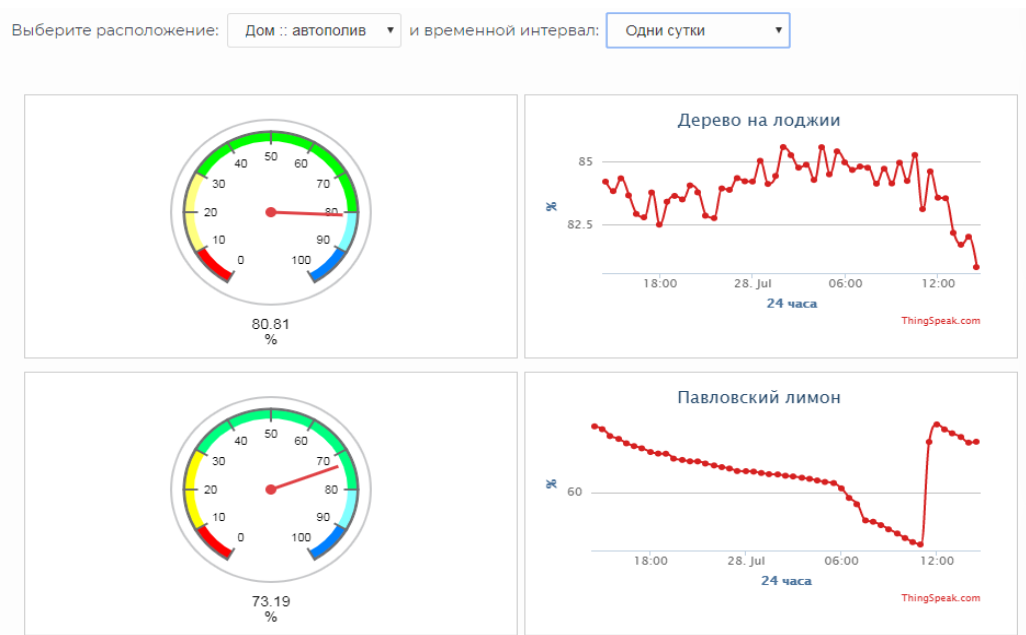


Рисунок 7.1 - Сервіс ThingSpeak

У принципі, пристроям 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 датчика 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.