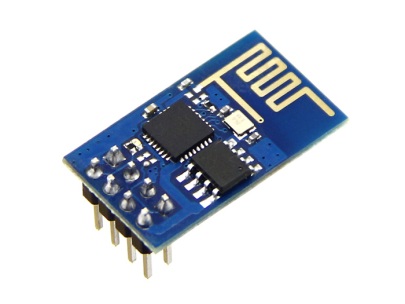
# Передача даних в інформаційних системах

В даній роботі я хочу показати передачі даних між мікроконтроллерами на прикладі I2C та передачі за допомогою радіомодулів на 433MHz. Структура даної інформаційної системи буде наступною: модуль esp8266 передає певні дані по I2C модулю arduino uno. До arduino uno буде підключений передавач на 433MHz. Аналогічний приймач буде підключений до модулю arduino pro mini, який в свою чергу буде підключений до реле. В даній роботі до реле підключений світлодіод, але оскільки в електромагнітних реле керуюче коло гальванічно не з’єднане з керованим (сухий контакт), то замість світлодіоду до реле можуть бути підключені контакти домашньої електромережі (в теорії. На практиці був світлодіод).

# Апаратна частина

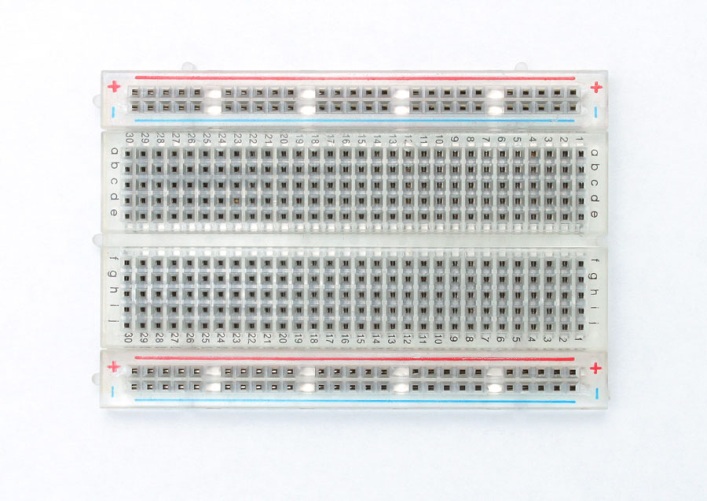
Нам знадобляться:

1. Модуль esp8266. В даній роботі використовувався esp8266-01.



2. Блок живлення для esp8266 на 3.3V.

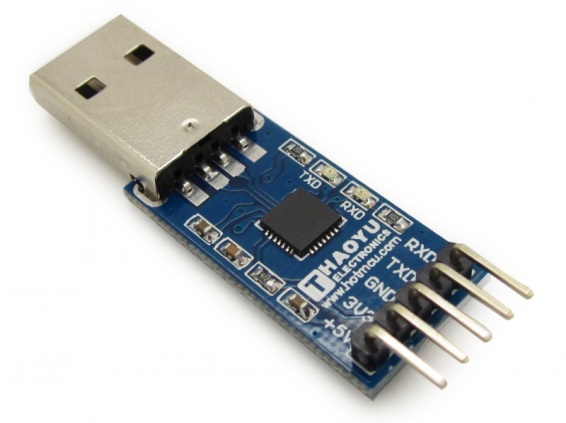
3. Макетна плата



4. Перемички



5. 2 usb-uart модуля



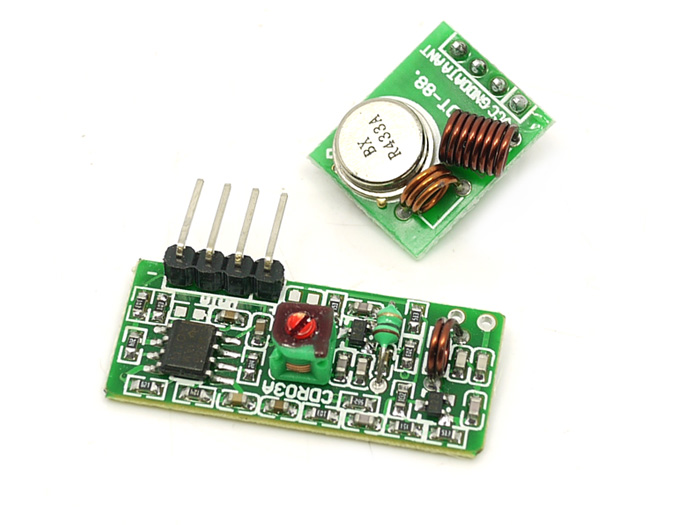
6. Світлодіод



7. Резистор до нього (номінал резистора визначається за формулою R=(VS-VL)/I, де VS = напруга живлення, VL = пряма напруга, розрахована для кожного типу діодів (як правило від 2 до 4 вольт) I = струм світлодіода (наприклад 20мA), він повинен бути менший максимально допустимого для вашого діода. Однак якщо у вас немає документації на діод, чи вам ліньки шукати ці параметри, візьміть резистор номіналом 100-300 Ом (можна більше, але тоді діод світитиме менш яскраво)).



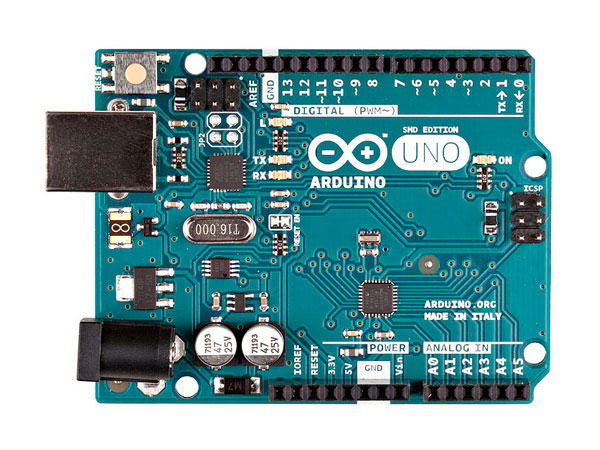
8. Модулі передавача (зверху) і приймача (знизу) на 433 MHz



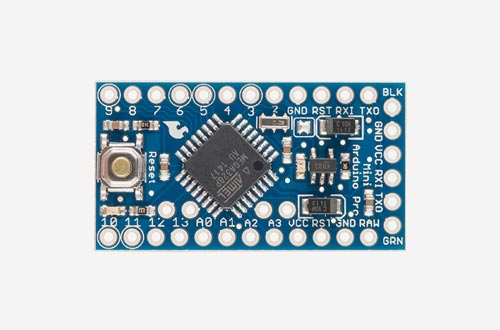
9. Електромагнітне реле (в даній роботі було використано реле рівнем керуючого сигналу – логічна «1»)



10. Модуль arduino uno (можна використовувати будь-який модуль у якого є підтримка I2C та бібліотеки VirtualWire

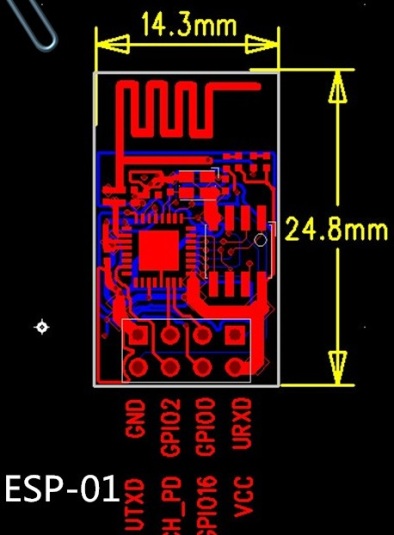


11. Моуль arduino pro mini на 5V, чи будь-який інший модуль, що підтримує бібліотеку VirtualWire та видає на пінах живлення 5V (якщо остання вимога не дотримується, потрібно додатково мати джерело живлення на 5V).



# Підключення та програмування

1. Почнемо з модуля esp8266. Для програмування цього модуля нам знадобится один з модулів usb-uart. Для живлення схеми використовуйте блок живлення на 3.3V. Підключення зручно робити на макетній платі за допомогою перемичок.



Виведіть від блоку живлення землю і живлення. Підключіть до живлення Vcc та CH\_PD. До землі підключіть GND esp8266 та GND usb-uart. Підключіть TX esp8266 до RX usb-uart, а RX esp8266 до TX usb-uart. Для прошивки модуля esp8266 підключіть GPIO0 до землі та відключіть і знову підключіть CH\_PD (це необхідно для перезапуску модуля). Загрузіть наступний скетч:

#include <ESP8266WiFi.h>

#include <WiFiClient.h>

#include <ESP8266WebServer.h>

#include <ESP8266mDNS.h>

#include <Wire.h>

typedef unsigned short int usi;

#define sda 0

#define scl 2

#define ardaddr 10

#define tnum 5

#define addr1on 11

#define addr1off 10

MDNSResponder mdns;

// Replace with your network credentials

const char\* ssid = "ntwrk";

const char\* password = "1qaz2wsx";

ESP8266WebServer server(80);

String webPage = "";

void setup(void){

webPage += "<h1>ESP8266 Web Server</h1><p>Socket #1 <a href=\"socket1On\"><button>ON</button></a>&nbsp;<a href=\"socket1Off\"><button>OFF</button></a></p>";

//pinMode(sndp,OUTPUT);

Serial.begin(115200);

Wire.begin(0,2);

WiFi.begin(ssid, password);

Serial.println("");

// Wait for connection

while (WiFi.status() != WL\_CONNECTED) {

delay(500);

Serial.print(".");

}

Serial.println("");

Serial.print("Connected to ");

Serial.println(ssid);

Serial.print("IP address: ");

Serial.println(WiFi.localIP());

if (mdns.begin("esp8266", WiFi.localIP())) {

Serial.println("MDNS responder started");

}

server.on("/", [](){

server.send(200, "text/html", webPage);

});

server.on("/socket1On", [](){

server.send(200, "text/html", webPage);

Wire.beginTransmission(ardaddr);

Wire.write(addr1on);

Wire.endTransmission();

});

server.on("/socket1Off", [](){

server.send(200, "text/html", webPage);

Wire.beginTransmission(ardaddr);

Wire.write(addr1off);

Wire.endTransmission();

});

server.begin();

Serial.println("HTTP server started");

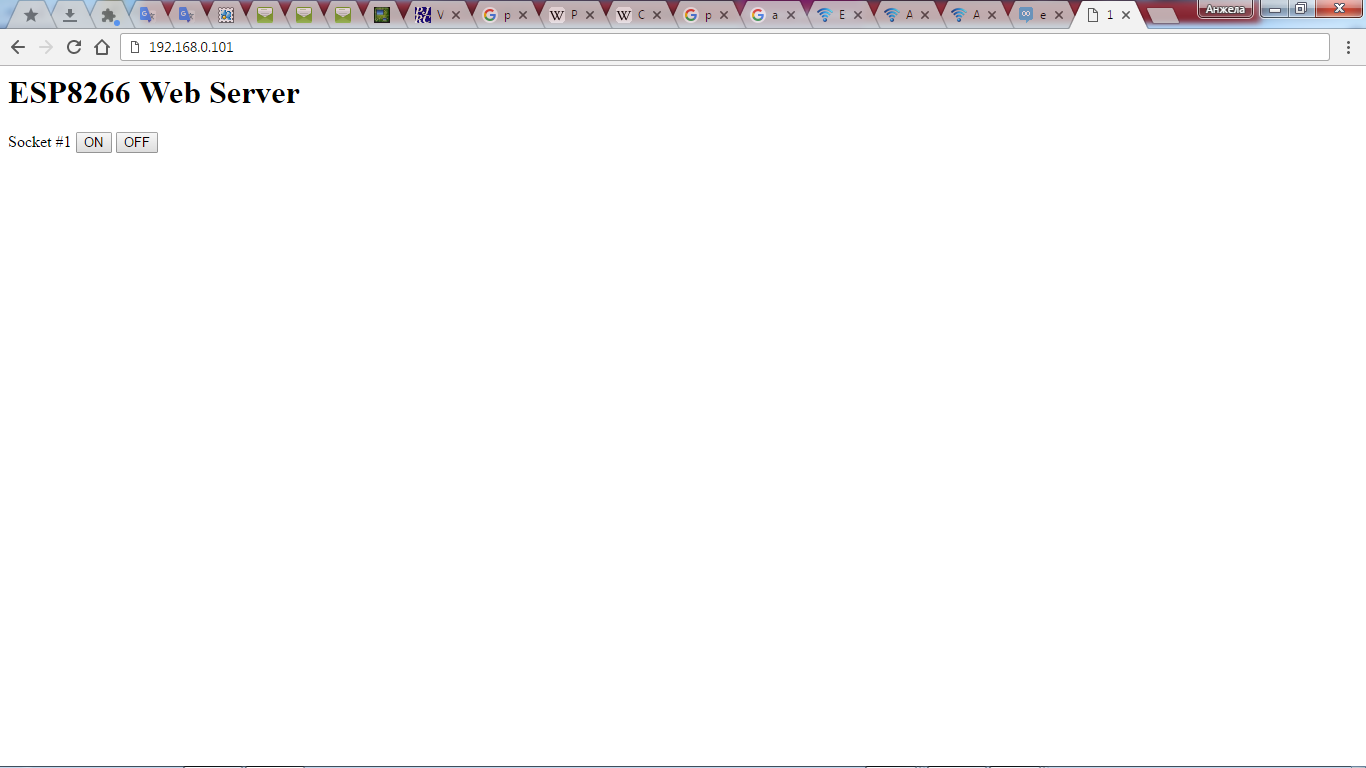
}

void loop(void){

server.handleClient();

}

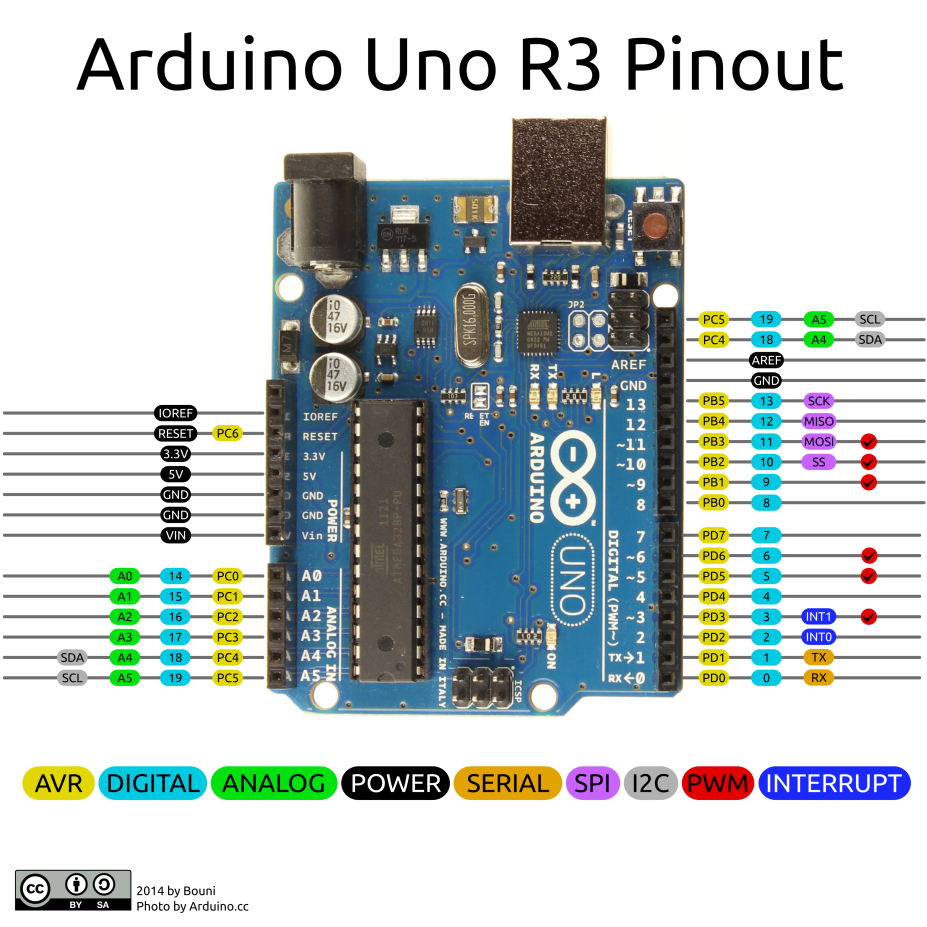
Відкривши монітор порта знайдіть рядок з IP адресою серверу (чотри числа розділені крапками після надпису «IP address:»).Ввівши в браузері цю адресу ви побачите щось таке:



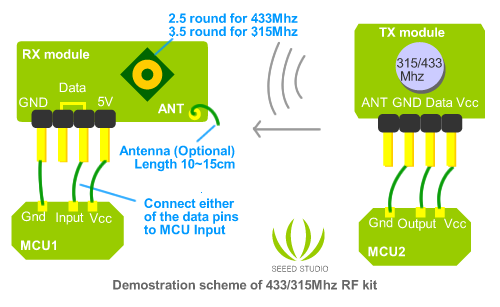
Запалювати/виключати світлодіод будемо відповідно натискаючи кнопки on/off.

Для передачі даних по I2C була використана бібліотека «Wire». Метод Wire.begin(0,2); ініціював піни 0 як sda, 2 – як scl. Тому відключіть GPIO0 від землі.

2. Arduino uno. Підключіть GPIO0 esp8266 до A4 arduino uno (sda-sda) та GPIO0 esp8266 до A5 arduino uno (scl-scl). Також підключіть до землі блоку живлення (за допомогою макетної плати та перемичок) землю arduino uno (GND pin) та землю передавача.



Передавач працює на вхідній напрузі 3-12V (чим вища напруга, тим більша потужність вихідного сигналу). В даній роботі передавач був підключений до живлення блоку живлення. Проте якщо потрібно передавати дані на більші відстані передавач потрібно підключити до більш потужного живлення (земля всіх гальванічно-зв’язаних модулів повинна бути спільною). Підключіть інформаційний пін передавача до 2-го піну arduino uno.



Загрузіть наступний скетч:

#include <Wire.h>

#include<VirtualWire.h>

typedef unsigned short int usi;

#define ardaddr 10

#define txpin 2

String cmnd="";

bool i2crecv=false;

void setup() {

Serial.begin(115200);

Wire.begin(ardaddr);

Wire.onReceive(onesprecv);

vw\_set\_tx\_pin(txpin);

vw\_set\_ptt\_inverted(true); // Required for DR3100

vw\_setup(2000);

}

void loop() {

usi strtt=millis();

if(i2crecv){

/\*Serial.println(cmnd.c\_str());

Serial.println(cmnd.length());\*/

vw\_send((uint8\_t \*)cmnd.c\_str(),cmnd.length());

vw\_wait\_tx(); // Wait until the whole message is gone;

i2crecv=false;

Serial.println("trtd");

}

}

void onesprecv(int amnt){

cmnd=""; i2crecv=true;

while (Wire.available()){

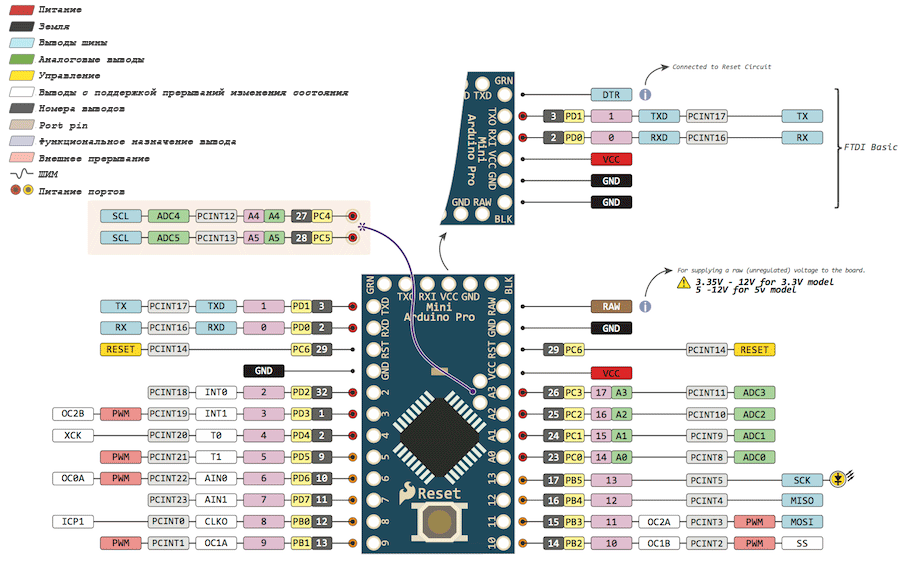
cmnd+=Wire.read();

}

Serial.println(cmnd);

}

3. Arduino pro mini. Підключіть один змодулів usb-uart до arduino pro mini (ftdi basic) (Vcc 5V – Vcc5; GND – GND; Tx-Rx; Rx-Tx; DTR-DTR). Існують два видимодулів arduino pro mini: на 3.3V 5V. Нам необхідний модуль на 5V тому, що живлення для приймача на 433MHz саме 5V. Підключіть приймач на 433MHz до . Arduino pro mini (GND – GND; Vcc – Vcc; data\_pin (будь-який з двох центральних пінів на приймачі – кожен з них data\_pin) – arduino\_pin\_17 (A3)). Підключіть до arduino pro mini реле (Vcc – Vcc; GND – GND; IN – arduino\_pin\_16 (A2)). Існують реле в яких керуючий сигнал – логічна «1» і ті, в яких керуючий сигнал – логічний «0». Якщо використовувати реле з керуючим сигналом «0», то в коді необхідно поміняти місцями рядки digitalWrite(rlp,LOW); та digitalWrite(rlp,HIGH);. Підключіть світлодіод до реле і окремого живлення (порядок наступний: живлення => резистор =>анод діоду (довший контакт) => незамкнуте коло реле). До іншої чистини незамкнутого кола реле відповідно підключається земля arduino pro mini.



Загрузіть наступний скетч:

#include<VirtualWire.h>

#define recvp 17

#define rlp 16

#define cmdl 2

#define ascii0 48

#define ascii1 49

void dcdr(uint8\_t buf[]){

switch(buf[0]){

case ascii1:

if(buf[1]==ascii0)

digitalWrite(rlp,LOW);

else if(buf[1]==ascii1)

digitalWrite(rlp,HIGH);

break;

}

}

void setup(){

Serial.begin(115200);

pinMode(rlp,OUTPUT);

vw\_set\_rx\_pin(recvp);

vw\_set\_ptt\_inverted(true); // Required for DR3100

vw\_setup(2000); // Bits per sec

vw\_rx\_start(); // Start the receiver PLL running

}

void loop() {

uint8\_t buf[VW\_MAX\_MESSAGE\_LEN];

uint8\_t buflen = VW\_MAX\_MESSAGE\_LEN;

if (vw\_get\_message(buf, &buflen)){

for(byte i=0;i<cmdl;i++) Serial.print((buf[i]==48)?0:1);

dcdr(buf);

Serial.println();

}

}

І можете перемикати світлодіод з браузеру.