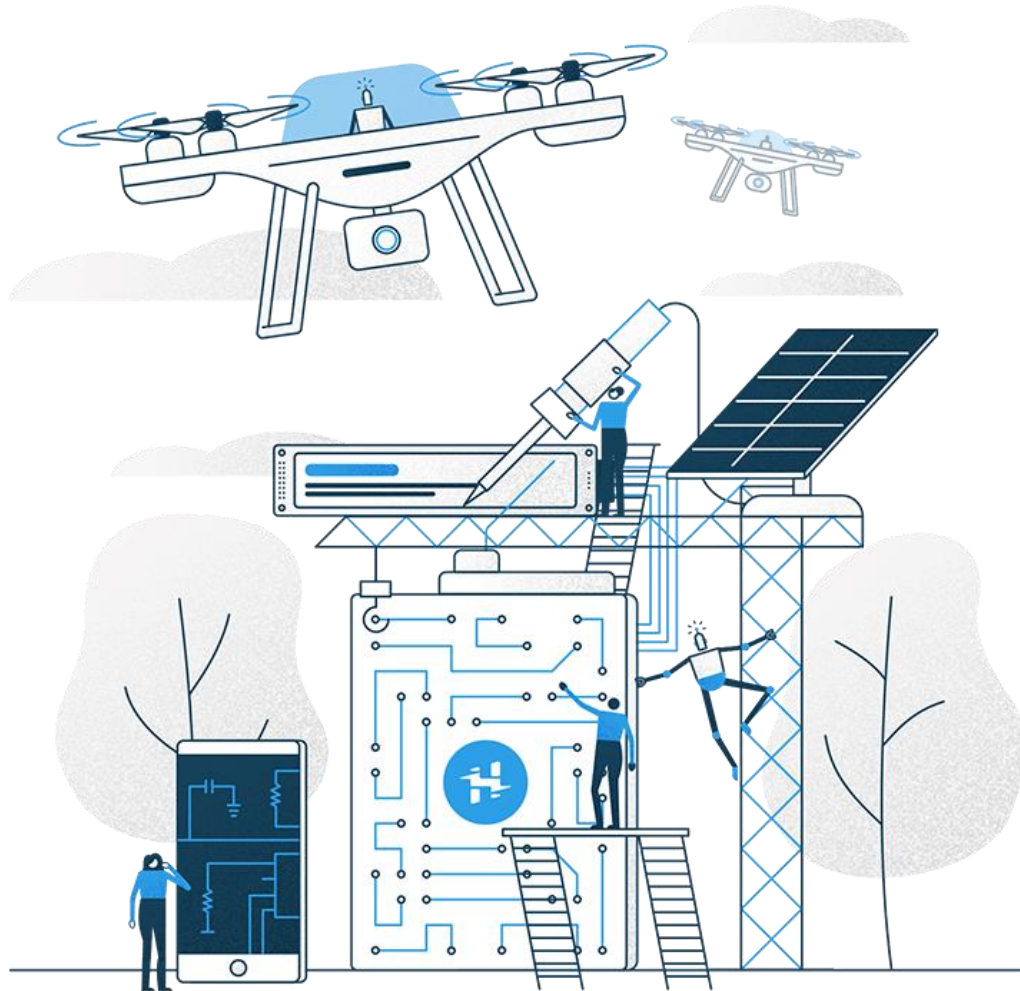


# МІКРОПРОЦЕСОРНІ СИСТЕМИ УПРАВЛІННЯ



Lesson 14

# Периферійний послідовний інтерфейс UART

У Ардуіно реалізована апаратна підтримка інтерфейсу послідовної передачі даних через порти 0, 1. Периферійний послідовний порта UART мікроконтролера ATmega 328 використовується для обміну інформацією з BT, WiFi модулем, ультразвуковим датчиком HC-SR04, ємнісним сенсором відбитків пальців Waveshare, датчиком CO2 K-33.

Його основні можливості: широкий діапазон швидкостей обміну даними; висока швидкість передачі при низькій частоті XTAL; 8 або 9-розрядний формат даних; виявлення помилок утрати даних при прийомі; виявлення помилок формату кадрів; виявлення помилкового стартового біта; три окремих переривання: по завершенню передачі, по порожньому регістру передавача і по завершенню прийому.

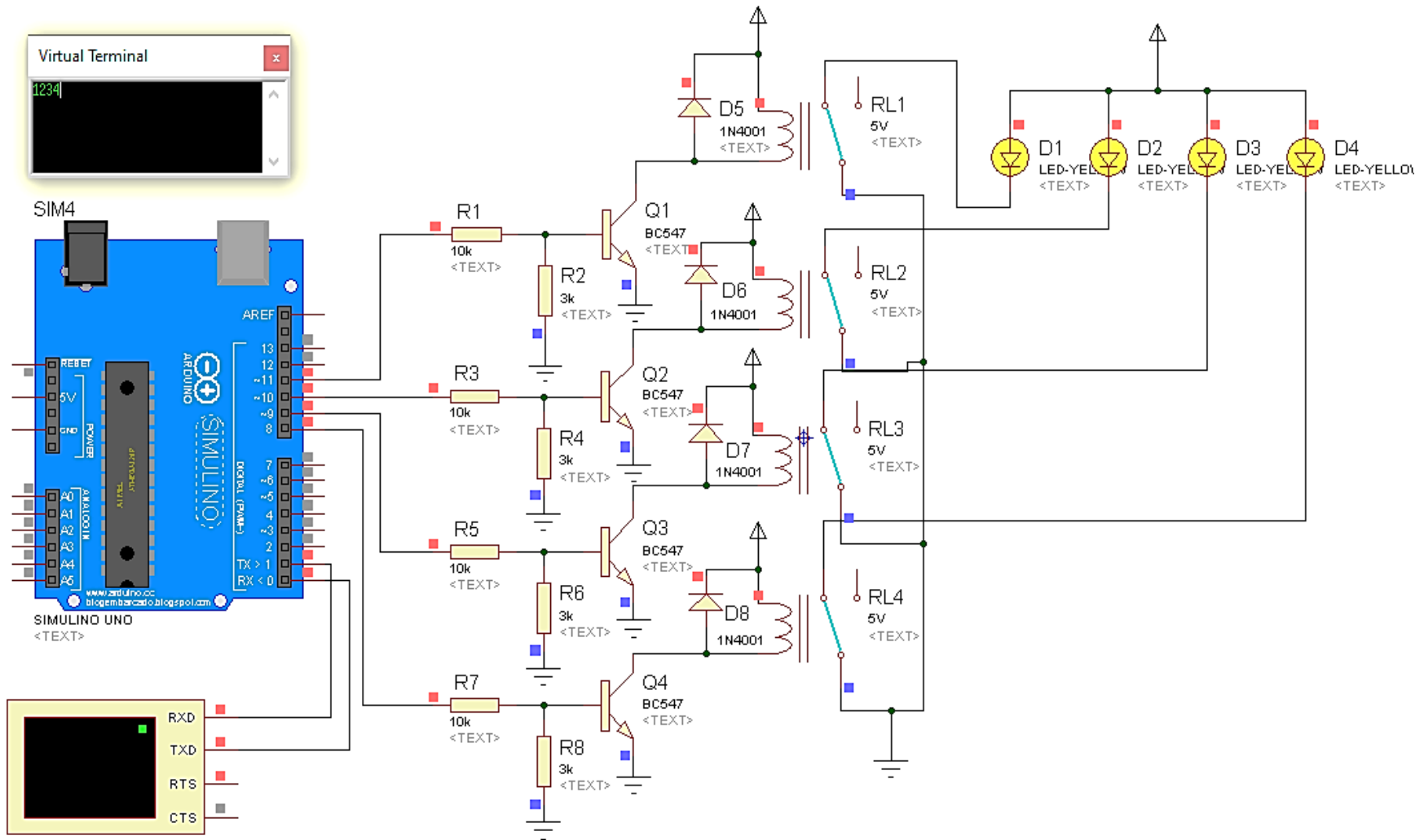
При передачі модуль UART додає до вхідного символу (8 або 9 біт) на початку — старт-біт (нуль), а в кінці — стоп-біт (одиниця), формуючи таким чином 10- або 11-бітову послідовність. Отримані значення передаються до регістра зсуву, який по черзі передає біти на вихід передавача TXD (вивід PD1). Швидкість видачі біт на вихід передавача визначається параметром baud rate (швидкість передачі інформації; вимірюється в бодах), яким можна керувати.

# Периферійний послідовний інтерфейс UART

Приймач модуля UART безперервно перевіряє стан входу RXD, на якому за відсутності даних встановлюється рівень «1». Приймач зчитує інформацію з входу в 16 разів швидше. При виявленні на виводі RXD рівня «0» (тобто можливого старт-біта) мікроконтролер пропускає шість відліків, а потім робить три вибірки. Ці вибірки доводяться на відлік 8, 9 і 10 для кожного біта, що приймається, і, таким чином, зчитування значення біта відбувається в середині інтервалу його передачі, що дозволяє працювати з сигналами, що мають фронти великої тривалості.

Якщо мікроконтролер виявляє, що на виводі RXD все ще присутній рівень «0», тобто прийшов стар-біт, модуль UART переходить в робочий режим і починає зчитувати байт. Якщо ж на виводі RXD вже присутній рівень «1», вважається, що перший відлік був просто шумом, і модуль переходить до очікування коректного символу. Якщо приймач визначив, що прийшов дійсний символ, він починає брати по три відліки кожного біта в середині інтервалу його передачі. Якщо значення всіх трьох відліків біта не збігаються, то значення біта набуває рівним значенню двох однакових відліків. На завершення модуль зчитує вибірки, що відносяться до стоп-біту. Для того, щоб було вирішено про коректний прийом символу, принаймні, дві з цих вибірок мають дорівнювати одиниці. Інакше модуль вважає символ за невірною кадрований і реєструє помилку кадркування (framing error)

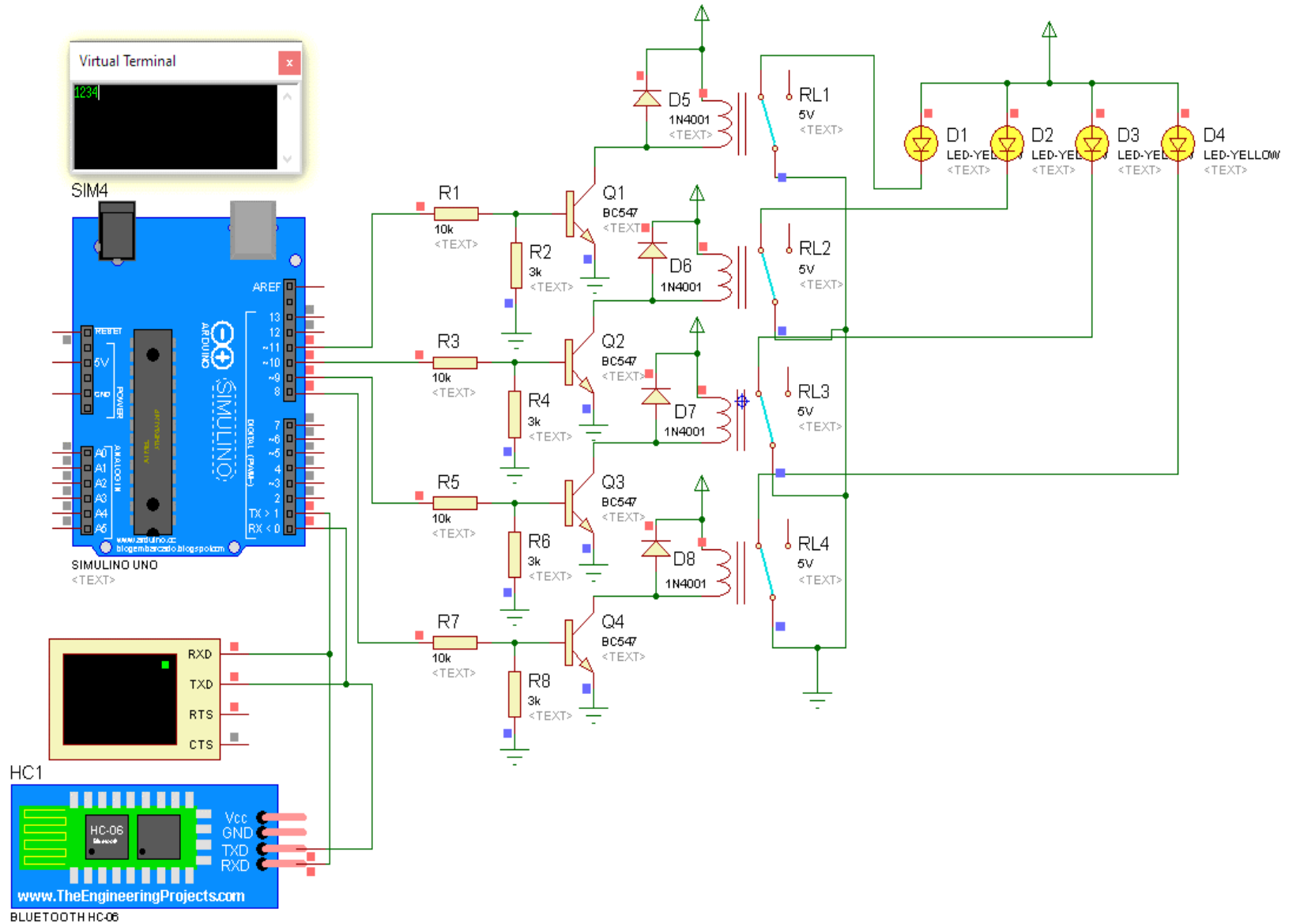
# Керування 4 виконавчими пристроями по UART



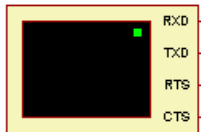
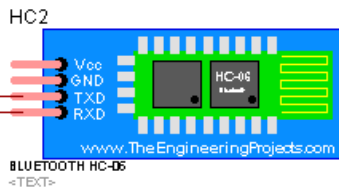
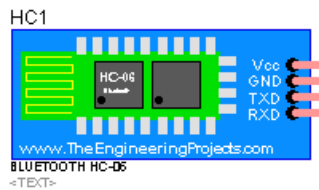
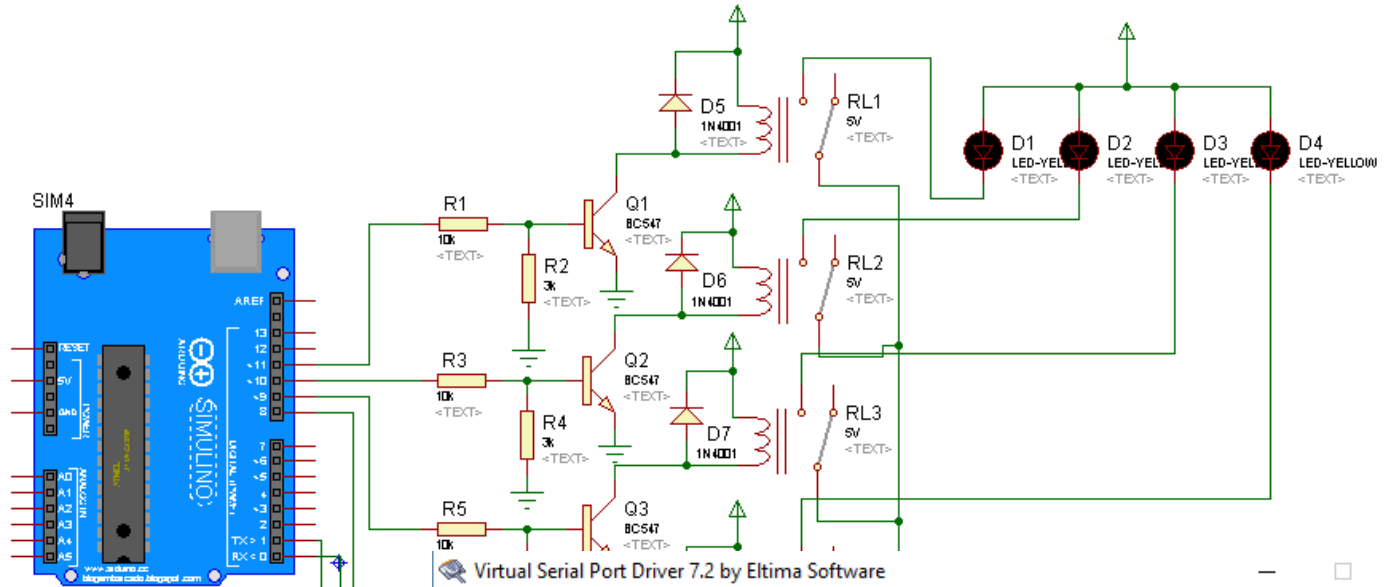
# Керування 4 виконавчими пристроями по UART

```
1 char incomingByte;
2 int b1=0;
3 int load1 = 8;
4 int load2 = 9;
5 int load3 = 10;
6 int load4 = 11;
7 void setup() {
8     Serial.begin(9600);
9     pinMode(load1, OUTPUT);
10    pinMode(load2, OUTPUT);
11    pinMode(load3, OUTPUT);
12    pinMode(load4, OUTPUT);
13 }
14 void loop() {
15     if (Serial.available() > 0) { //якщо є дані
16         incomingByte = Serial.read(); // читаємо байт
17         if ( incomingByte == '1') {PORTB^=0b00000001;}
18         //переключити D8, решта залишити без змін
19         if ( incomingByte == '2') {PORTB ^=0b00000010;}
20         if ( incomingByte == '3') {PORTB ^=0b00000100;}
21         if ( incomingByte == '4') {PORTB ^=0b00001000;}
22     }
23 }
```

# Керування 4 виконавчими пристроями по BT/UART



# Керування 4 виконавчими пристроями по BT



Virtual Serial Port Driver 7.2 by Eltima Software

Port pairs Options Help

Serial ports explorer

Virtual Serial Port Driver

Physical ports

- COM1

Virtual ports

- COM3 [4800-N-8-1]
- COM4 [4800-N-8-1]

Other virtual ports

Manage ports Port access list Custom pinout

VSPD by Eltima can create virtual serial ports with any names you like, so you are not limited to COMx names only. However, please, make sure that programs working with these ports support custom port names.

First port: COM2

Second port: COM5

Add pair

First port: COM3

Second port: COM4

Delete pair

Enable strict baudrate emulation

Break line/Restore connection

All virtual serial pairs will be removed from your system. Please, make sure all ports are closed.

Delete all

Pair COM3 <-> COM4 created successfully

# Онлайн сервіс RemoteXY



РЕДАКТОР ПРИКЛАДНА ПРОГРАМА ДОКУМЕНТАЦІЯ Приклади СООБЩЕСТВО БЛОГ УВІЙТИ



Створіть графічний інтерфейс для вашого пристрою

**ПРОБУВАТИ!**

Підтримка:  
Arduino, ESP8266, ESP32  
Android, iOS  
Bluetooth, Wi-Fi, Ethernet, USB  
Internet з будь-якого місця

Чотири кроки для досягнення мети

1. Розробіть інтерфейс.
2. Завантажте вихідний код.
3. Підключіть модуль зв'язку.
4. Встановіть мобільний додаток для управління.





# Онлайн сервіс RemoteXY

The screenshot shows the RemoteXY web interface. At the top, there is a dark green header with the RemoteXY logo on the left and navigation links: РЕДАКТОР, ПРИКЛАДНА ПРОГРАМА, ДОКУМЕНТАЦІЯ, Приклади, СООБЩЕСТВО, БЛОГ, УВІЙТИ. Below the header, the interface is divided into three main sections:

- елементи (Elements):** A sidebar on the left containing various UI components under the heading "управління" (Control). These include: кнопка (button), вимикач тель (toggle switch), Переключити тель (toggle switch), слайдер (slider), джойстик (joystick), RGB коло (RGB wheel), and Поле введення (input field) with an "edit" icon. Below these are "індикація" (indicator) and "оформлення" (styling) options.
- New project:** A central workspace with a grid background. It contains a toolbar with icons for file operations and a large green button labeled "Отримати вихідний код" (Get source code). In the center of the grid is a large smartphone mockup with a teal screen displaying the text "Перетящите сюди елементи управління" (Drag elements here).
- властивості (Properties):** A sidebar on the right with a red dashed border, containing a list of properties: конфігурація (configuration), підключення модуля (module connection), and екран (screen).

# Онлайн сервіс RemoteXY

з'єднання    Пристрій    модуль    середа

 Bluetooth     Arduino UNO     HC-05 (06) Bluetooth module     Arduino IDE

**сумісні модулі**

 HC-05 (06) Bluetooth module     HM-10 Bluetooth BLE module

**доступні модулі**

 ESP8266 Wi-Fi module     Ethernet shield W5100     USB to UART converter     WiFi on chip     Bluetooth on chip

застосувати    скасування

ВЛАСТИВОСТІ

конфігурація

 Bluetooth

 Arduino UNO

 HC-05 (06) Bluetooth module

 Arduino IDE

підключення модуля

екран

# Онлайн сервіс RemoteXY

The screenshot displays the RemoteXY online service interface. At the top, the logo "RemoteXY" is visible on the left, and navigation links "РЕДАКТОР", "ПРИКЛАДНА ПРОГРАМА", "ДОКУМЕНТАЦІЯ", "Приклади", "СООБЩЕСТВО", "БЛОГ", "УВІЙТИ" are on the right. The main workspace is titled "New project \*" and contains a central mobile phone simulator with the text "Перетящите сюди елементи управління" (Drag elements here). To the left is a panel "елементи" (elements) with categories "управління" (control) and "індикація" (indication). The "управління" category includes: кнопка (button), вимикач тель (toggle switch), Переключити тель (toggle), слайдер (slider), джойстик (joystick), and RGB коло (RGB wheel). Below these is "edit" and "Поле введення" (input field). The "індикація" category includes "оформлення" (styling). To the right is a panel "властивості" (properties) with sections: "конфігурація" (configuration) including Bluetooth, Arduino UNO, HC-05 (06) Bluetooth module, and Arduino IDE; and "підключення модуля" (module connection) which is highlighted with a red dashed box. The "підключення модуля" section includes: "Інтерфейс підключення:" (connection interface) with a dropdown set to "Hardware Serial"; "Порт Serial:" (Serial port) with a dropdown set to "Serial, pins 0 (RX) and 1 (TX)"; and "Швидкість обміну:" (baud rate) with a dropdown set to "9600". Below this is the "екран" (screen) section with "Колір фону:" (background color) set to black, "орієнтація:" (orientation) set to "вертикально" (vertical), and "Пароль доступу:" (access password).

# Онлайн сервіс RemoteXY

The image displays the RemoteXY online service interface. On the left is a control panel with various widgets: a button (кнопка), a toggle switch (вимикач тель), a toggle switch (Переключити тель), a slider (слайдер), a joystick (джойстик), and an RGB wheel (RGB коло). Below these are an edit button (edit X) and a text input field (Поле введення). Under 'індикація' (indication) and 'оформлення' (styling), there are a label (Label / мітка), a panel (панель), and a page (Page / сторінка).

The central area shows a mobile phone screen with four channels (CH1, CH2, CH3, CH4), each with an OFF/ON toggle switch. Above the phone are icons for file operations (document, trash, save, share) and a green button labeled 'Отримати вихідний код' (Get source code) with a blue arrow pointing to it.

On the right is the Arduino IDE configuration panel. It includes the Arduino IDE logo and the text 'підключення модуля' (module connection). The connection interface is set to 'Hardware Serial' on 'Serial, pins 0 (RX) and 1 (TX)' with a baud rate of '9600'. Under 'екран' (screen), the background color is black, orientation is 'вертикально' (vertical), and a password field is present. The 'елемент' (element) section, highlighted with a red dashed border, shows settings for a switch: color (orange), background color (black), variable name 'switch\_1', 'Напис включення' (ON), 'Напис виключення' (OFF), shape 'прямокутна' (rectangular), and 'Прив'язати до висновку' (bind to output) set to '8'.

# Онлайн сервіс RemoteXY

Сергій \*\*\*    Мои проекты    Мої токени

РЕДАКТОР    ПРИЛОЖЕНИЕ    ДОКУМЕНТАЦИЯ    ПРИМЕРЫ    СООБЩЕСТВО    БЛОГ

## Исходный код для проекту: Новый проект

1. **Завантажити вихідний код** програми, відкрийте його в Arduino IDE.
2. Установити **бібліотеку RemoteXY** для Arduino IDE.
3. Скомпілюйте вихідний код та завантажте плату Arduino за допомогою Arduino IDE.
4. Правильно підключіть **модуль Bluetooth** к плате Arduino.
5. Установити мобільне додаток **RemoteXY, вер. 4.7.12** для смартфона / планшета.
6. Підключіть к Arduino з мобільного додатка.

project.ino    **Завантажити код**    Загрузить библиотеку

```
/*
-- New project --

This source code of graphical user interface
has been generated automatically by RemoteXY editor.
To compile this code using RemoteXY library 2.4.3 or later version
download by link http://remotexy.com/en/library/
To connect using RemoteXY mobile app by link http://remotexy.com/en/download/
- for ANDROID 4.7.12 or later version;
- for iOS 1.4.7 or later version;
```

# Онлайн сервіс RemoteXY

```
1 |//////////////////////////////////////
2 |//          RemoteXY include library          //
3 |//////////////////////////////////////
4 |// определение режима соединения и подключение библиотеки RemoteXY
5 |#define REMOTEXY_MODE__HARDSERIAL
6 |#include <RemoteXY.h>
7 |#define REMOTEXY_SERIAL Serial
8 |#define REMOTEXY_SERIAL_SPEED 9600
9 |#pragma pack(push, 1)
10|uint8_t RemoteXY_CONF[] =
11| { 255,4,0,0,0,115,0,11,24,1,
12|   2,0,25,8,22,11,2,26,31,31,
13|   79,78,0,79,70,70,0,2,0,25,
14|   25,22,11,2,26,31,31,79,78,0,
15|   79,70,70,0,2,0,25,42,22,11,
16|   2,26,31,31,79,78,0,79,70,70,
17|   0,2,0,25,59,22,11,2,26,31,
18|   31,79,78,0,79,70,70,0,129,0,
19|   8,10,12,6,17,67,72,49,0,129,
20|   0,8,28,12,6,17,67,72,50,0,
21|   129,0,8,45,12,6,17,67,72,51,
22|   0,129,0,8,61,12,6,17,67,72,
23|   52,0 };
24|struct {
25|  // input variables
26|  uint8_t switch_1; // =1 если переключатель включен и =0 если отключен
27|  uint8_t switch_2; // =1 если переключатель включен и =0 если отключен
28|  uint8_t switch_3; // =1 если переключатель включен и =0 если отключен
29|  uint8_t switch_4; // =1 если переключатель включен и =0 если отключен
30|  // other variable
31|  uint8_t connect_flag; // =1 if wire connected, else =0
32|} RemoteXY;
33|#pragma pack(pop)
```

# Онлайн сервіс RemoteXY

```
38 #define PIN_SWITCH_1 8
39 #define PIN_SWITCH_2 9
40 #define PIN_SWITCH_3 10
41 #define PIN_SWITCH_4 11
42
43 void setup() {
44     RemoteXY_Init ();
45     pinMode (PIN_SWITCH_1, OUTPUT);
46     pinMode (PIN_SWITCH_2, OUTPUT);
47     pinMode (PIN_SWITCH_3, OUTPUT);
48     pinMode (PIN_SWITCH_4, OUTPUT);
49     // TODO you setup code
50 }
51
52 void loop()
53 {
54     RemoteXY_Handler ();
55     digitalWrite(PIN_SWITCH_1, (RemoteXY.switch_1==0)?LOW:HIGH);
56     digitalWrite(PIN_SWITCH_2, (RemoteXY.switch_2==0)?LOW:HIGH);
57     digitalWrite(PIN_SWITCH_3, (RemoteXY.switch_3==0)?LOW:HIGH);
58     digitalWrite(PIN_SWITCH_4, (RemoteXY.switch_4==0)?LOW:HIGH);
59     // TODO you loop code
60     // используйте структуру RemoteXY для передачи данных
61     // не используйте функцию delay()
62 }
```

---

# SoftwareSerial

Бібліотека SoftwareSerial дозволяє реалізувати послідовний інтерфейс на будь-яких цифрових портах Arduino за допомогою програмних засобів, які дублюють функціональність UART. Бібліотека дозволяє програмно створювати кілька послідовних портів, які працюють на швидкості до 115200 бод. Для пристроїв, що працюють з інвертованим сигналом, в бібліотеці передбачено відповідний параметр, що включає інвертування

```
1 #include <SoftwareSerial.h>
2 SoftwareSerial mySerial(10, 11); // RX, TX
3
4 void setup() {
5     // Ініціалізація послідовного інтерфейсу
6     Serial.begin(57600);
7     Serial.println("Good");
8     // встановлюємо швидкість передачі даних для SoftwareSerial
9     mySerial.begin(9600);
10    mySerial.println("Hello, world!");
11 }
12
13 void loop() {
14     if (mySerial.available()){
15         Serial.write(mySerial.read());
16         // TODO you setup code
17     }
18 }
```