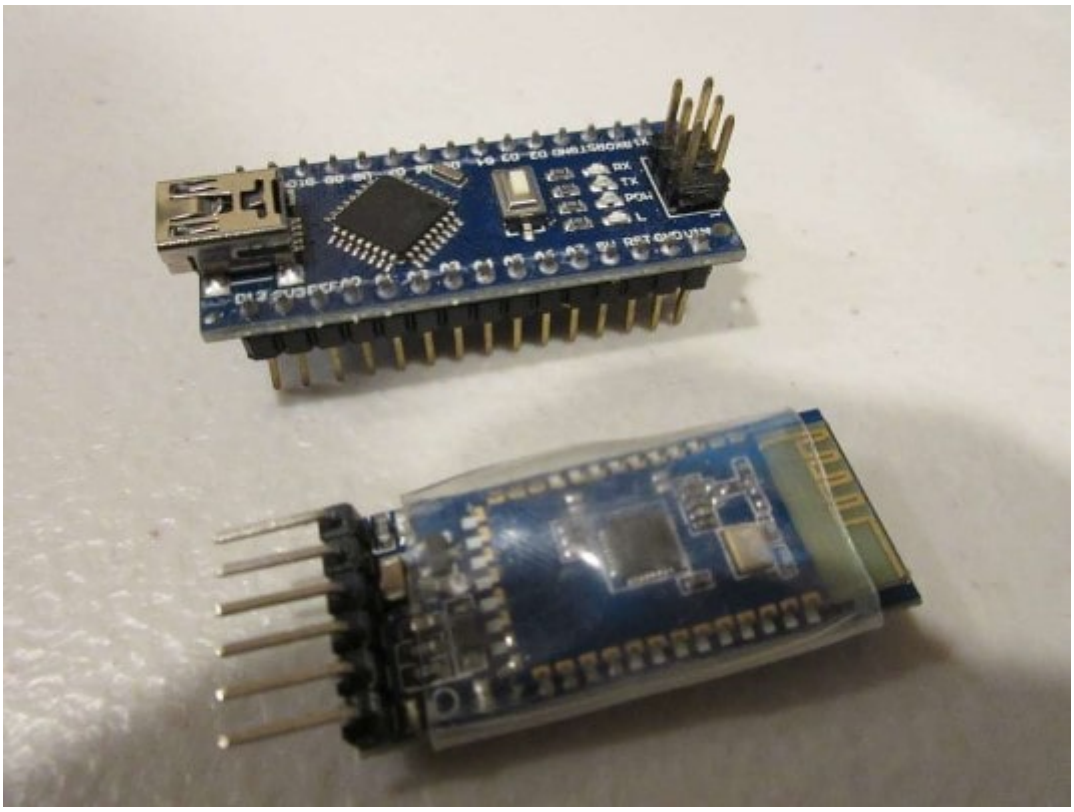
HC-05 Module with Arduino

The HC-05 Bluetooth Module

The HC-05 module allows an Arduino to communicate over Bluetooth using the Serial Port Profile (SPP) protocol. This enables it to wirelessly communicate with a Bluetooth Classic-enabled device, such as a computer. Notably, the Arduino code used to communicate through the HC-05 closely resembles the code used to communicate over the standard USB serial port, which will be demonstrated below.

Below is an image of the HC-05 module, with an Arduino Nano for scale:



The HC-05 module has a serial interface, with RX and TX pins. It is a 3.3V logic device, so you need a voltage divider for it to work with a 5V Arduino. (See Using an Ultrasonic Sensor with the Raspberry Pi for more information.)

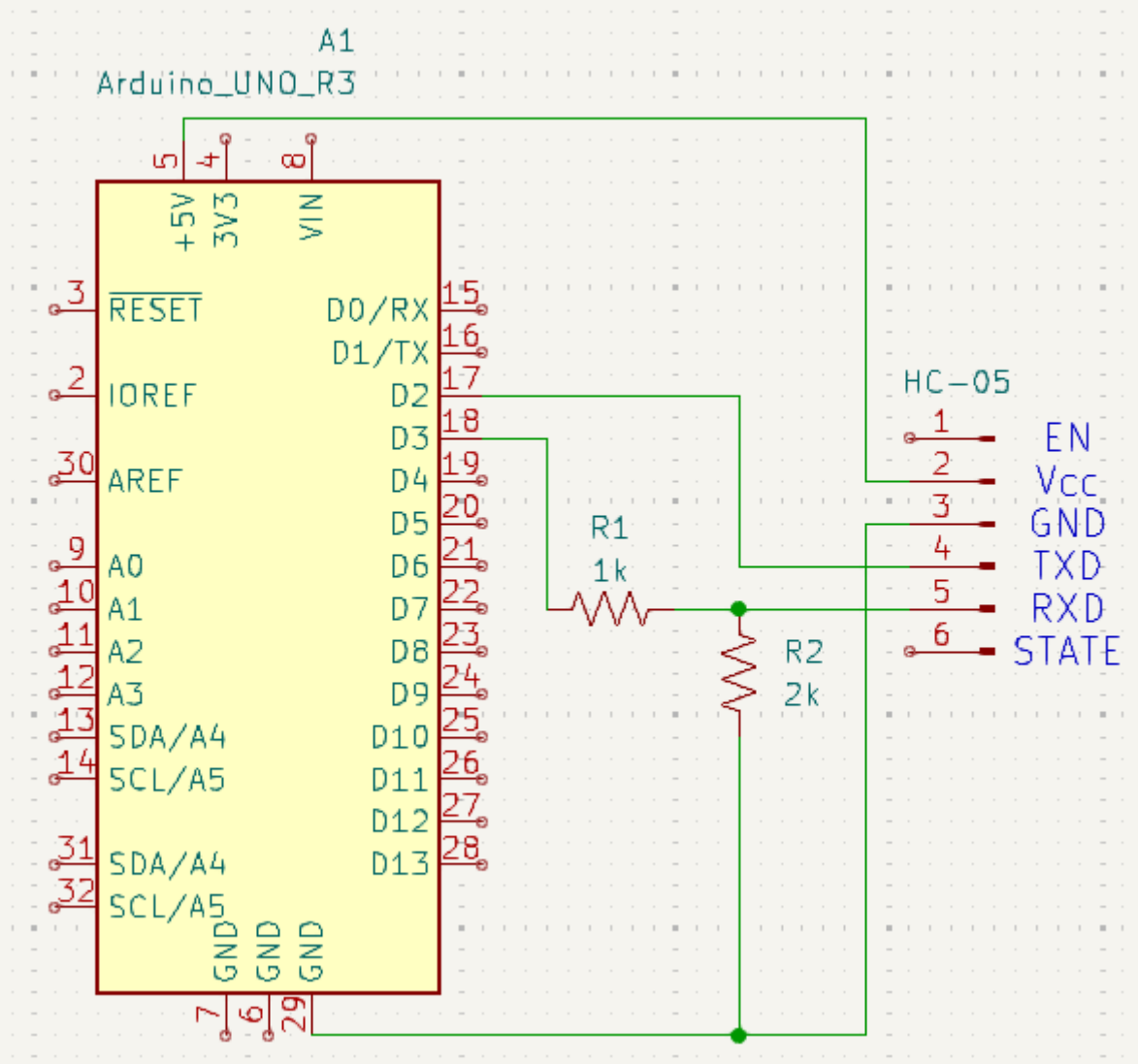
HC-05 Module Pinout

Pin	Function
STATE	Tells if the HC-05 is connected wirelessly to a device
RXD	Data received by the module
TXD	Data transmitted by the module
GND	Ground connection

Pin	Function
V_{CC}	Power connection
EN	Toggle between Data Mode and AT Command Mode

- EN pin HIGH: Set to AT Command Mode[1]
- EN Pin Low: Set to Data Mode (Default)

Schematic



Code

```
#include <SoftwareSerial.h> // Include library

SoftwareSerial bt(2, 3); // RX, TX

void setup() {
  bt.begin(9600);
```

```
  pinMode(LED_BUILTIN, OUTPUT);   // Set built-in LED pin as output. LED_BUILTIN is
equal to 13 on most boards,
  // it changes depending on the board you select, since not all boards have the LED
on 13.
  digitalWrite(LED_BUILTIN, LOW); // LED off on start

  bt.println("Enter '0' to turn off LED, enter '1' to turn on"); // Send instructions
}

void loop() {
  if (bt.available()) {
    char ch = bt.read(); // Read incoming command
    if (ch == '0') {
      // Turn off LED
      bt.println("LED Off");
      digitalWrite(LED_BUILTIN, LOW);
    }
    else if (ch == '1') {
      // Turn on LED
      bt.println("LED On");
      digitalWrite(LED_BUILTIN, HIGH);
    }
  }
}
```

This code uses the `SoftwareSerial` library, which allows serial communication with the HC-05 to be on pins other than 0 and 1. This prevents Bluetooth communication from interfering from regular USB communication with the Arduino, such as uploading code and using the serial monitor.

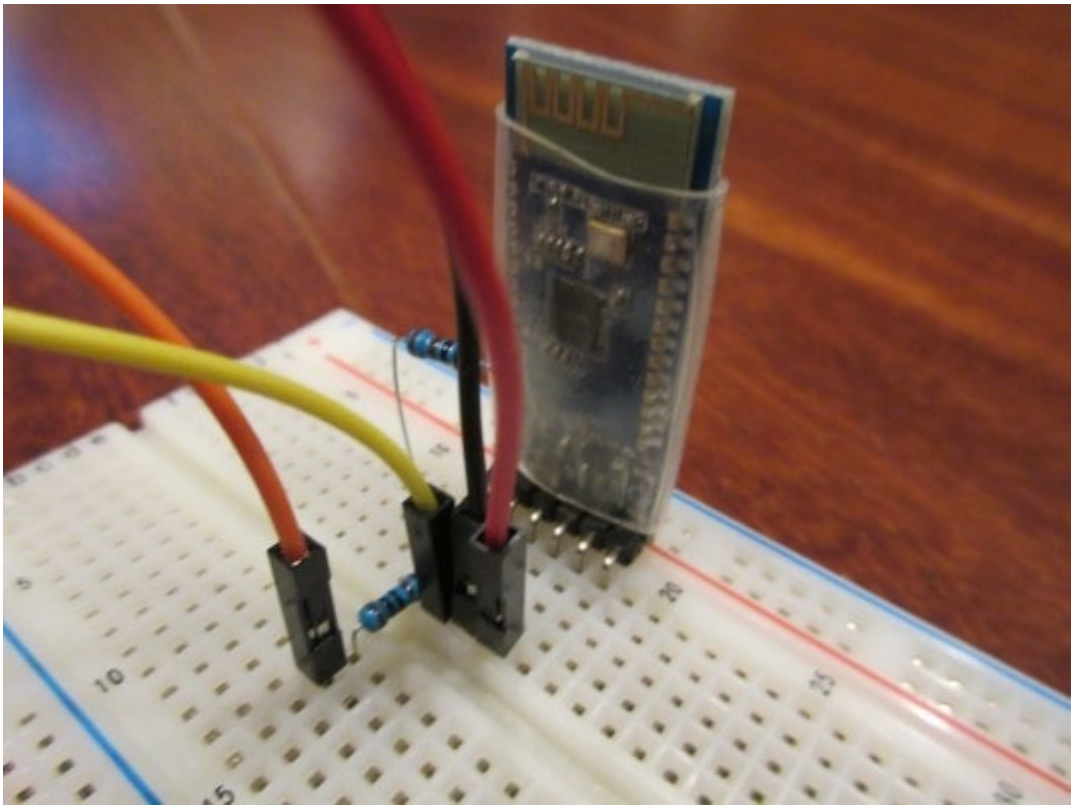
`SoftwareSerial` has a similar interface to `Serial`: the `available` and `read` functions allow the Arduino to receive data from Bluetooth, while the `println` function allows it to send data over Bluetooth with a newline at the end.

Controlling Your Arduino

To control your Arduino from your computer, you can use WhaleConnect. For more information on Bluetooth communication in WhaleConnect, refer to its usage documentation.

You will need to pair the HC-05 with your computer before communicating with it. It will usually have a name such as **JDY-30, JDY-31, HC-05, or HC-06**. If it requires a passkey, it is usually **0000 or 1234**.

After connecting, you are ready to control your Arduino. Enter 0 to turn off the D13 L LED and enter 1 to turn on the LED.



[1] AT Command Mode allows you to send commands and configuration settings to the module. It is not used in this guide.