

Adding Bluetooth (BT) to your robot

Staffan Melin, staffan.melin@oscillator.se, 2015-04-24

This document is some kind of manual for adding BT to the robot described on <http://www.oscillator.se/arduino> (English) and <http://oscillator.se/skola> (Swedish).

With inspiration from <http://blog.miguelgrinberg.com/post/building-an-arduino-robot-part-vi-remote-control>.

Hardware



Get your hands on a "arduino wireless bluetooth transceiver hc-06" on for example Ebay.

The BT module has the following pins:

- STATE: unused
- RXD: receive data; connect to A0 (D14) on the Arduino
- TXD: transmit data; connect to A1 (D15) on the Arduino
- GND: Connect to ground on the Arduino
- VCC: Connect to +5V on the Arduino
- EN: unused

Software

The BT module communicates serially. We would like to use any pins for this, so we need to use the SoftwareSerial library.

Start

At the start of your program you have to add the following lines.

```
#define BT_RX_PIN 15 // A1
#define BT_TX_PIN 14 // A0

#include <SoftwareSerial.h>
SoftwareSerial BTSerial(BT_RX_PIN, BT_TX_PIN);
```

This will create a SoftwareSerial object called BTSerial, which we will use to talk to the BT module.

Setup()

As the module communicates serially, you have to add

```
BTSerial.begin(9600);
```

in the setup() function.

Loop()

We are assuming you have the obstacle-avoiding logic in loop(). Add this code:

```
if (true)
{
    BTControl();
}
// previous code from loop() goes here (ie is inactivated)
// ...
}
```

So, what does BTControl() look like?

```
void BTControl()
{
    if (BTSerial.available() > 0)
    {
        char ch = BTSerial.read();
        switch (ch) {
            case '8': // up
                moveForward();
                break;
            case '2': // down
                moveBackward();
                break;
            case '4': // left
                moveTurnLeft();
                break;
            case '6': // right
                moveTurnRight();
                break;
            case 'D': // release = stop
                moveStop();
                break;
        } // switch
    }
}
```

If you like you can (as I have done) replace the

```
if (true)
```

statement in the loop() function with something that checks the status of a button/switch.

Inspiration for this can be found on <http://www.arduino.cc/en/Tutorial/Switch>.

A BT sender

I have used the **BlueStick** Android app (<https://play.google.com/store/apps/details?id=com.inex.BlueStickControl&hl=en>), which provides a nice layout for sending BT characters. Use the "PAIR" button to search for and connect to the BT module on your robot.

You can also use any BT terminal program for sending the characters that are detected in the BTControl() function.

Good luck!