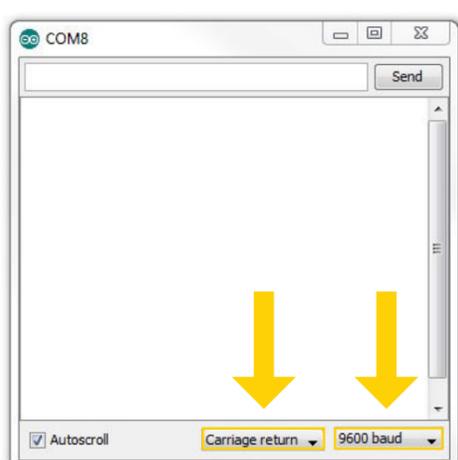
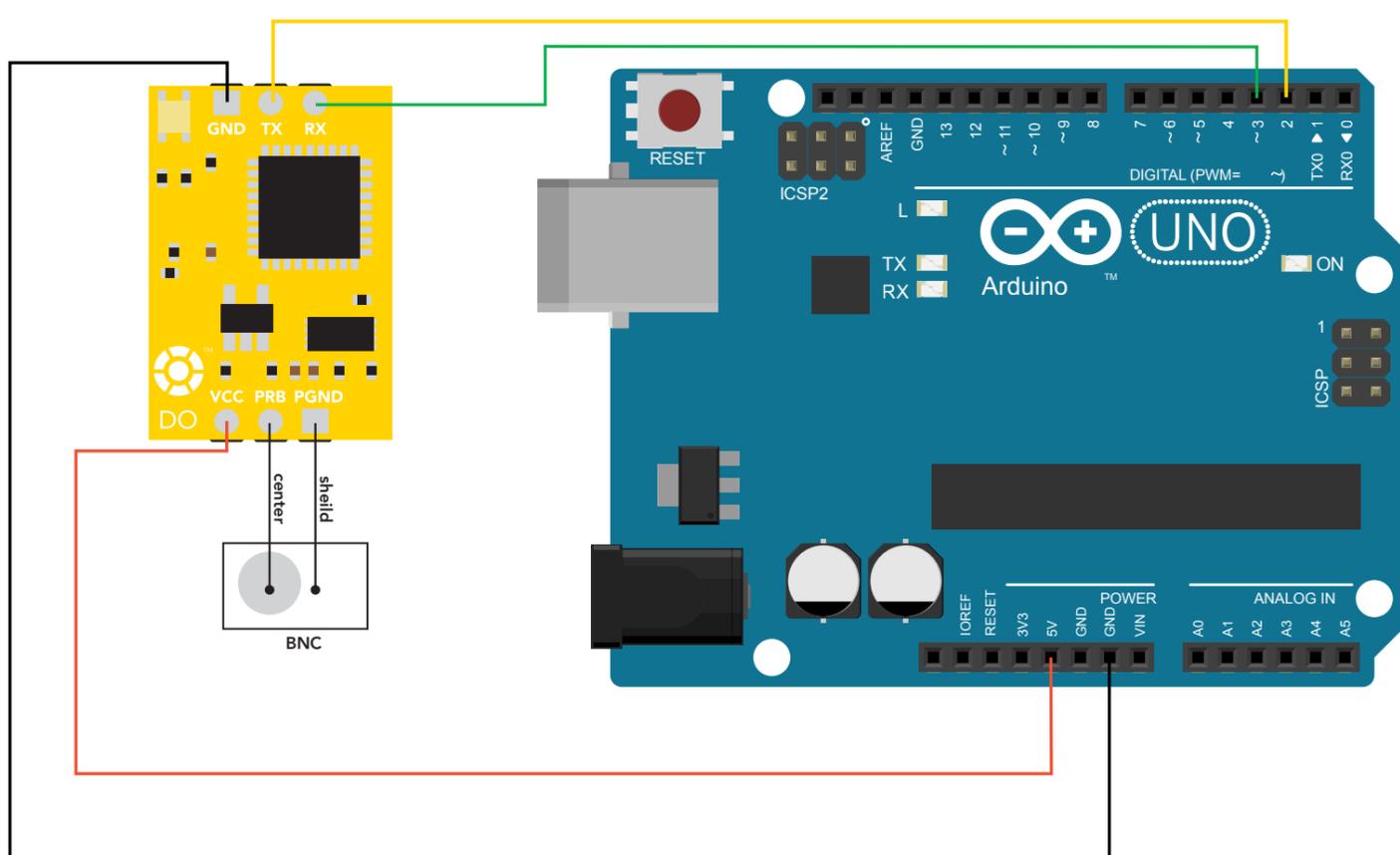


# Arduino Uno Dissolved Oxygen Sample Code



**//This code was written to be easy to understand.  
//Code efficiency was not considered.  
//Modify this code as you see fit.  
//This code will output data to the Arduino serial monitor.  
//Type commands into the Arduino serial monitor to control the DO circuit.  
//This code was written in the Arduino 1.6.5 IDE  
//An Arduino UNO was used to test this code.**



```
#include <SoftwareSerial.h>           //we have to include the SoftwareSerial library, or else we can't use it
#define rx 2                          //define what pin rx is going to be
#define tx 3                          //define what pin tx is going to be

SoftwareSerial myserial(rx, tx);      //define how the soft serial port is going to work

String inputstring = "";              //a string to hold incoming data from the PC
String sensorstring = "";            //a string to hold the data from the Atlas Scientific product
boolean input_string_complete = false; //have we received all the data from the PC
boolean sensor_string_complete = false; //have we received all the data from the Atlas Scientific product
float DO;                             //used to hold a floating point number that is the DO

void setup() {                        //set up the hardware
  Serial.begin(9600);                 //set baud rate for the hardware serial port_0 to 9600
  myserial.begin(9600);               //set baud rate for the software serial port to 9600
  inputstring.reserve(10);            //set aside some bytes for receiving data from the PC
  sensorstring.reserve(30);          //set aside some bytes for receiving data from Atlas Scientific product
}

void serialEvent() {                  //if the hardware serial port_0 receives a char
  inputstring = Serial.readStringUntil(13); //read the string until we see a <CR>
  input_string_complete = true;       //set the flag used to tell if we have received a completed string from the PC
}

void loop() {                         //here we go...

  if (input_string_complete){         //if a string from the PC has been received in its entirety
    myserial.print(inputstring);      //send that string to the Atlas Scientific product
    myserial.print('\r');             //add a <CR> to the end of the string
    inputstring = "";                //clear the string
    input_string_complete = false;    //reset the flag used to tell if we have received a completed string from the PC
  }

  if (myserial.available() > 0) {     //if we see that the Atlas Scientific product has sent a character
    char inchar = (char)myserial.read(); //get the char we just received
    sensorstring += inchar;           //add the char to the var called sensorstring
    if (inchar == '\r') {             //if the incoming character is a <CR>
      sensor_string_complete = true;  //set the flag
    }
  }

  if (sensor_string_complete== true) { //if a string from the Atlas Scientific product has been received in its entirety
    Serial.println(sensorstring);     //send that string to the PC's serial monitor
    if (isdigit(sensorstring[0])) {   //if the first character in the string is a digit
      DO = sensorstring.toFloat();    //convert the string to a floating point number so it can be evaluated by the Arduino
      if (DO >= 6.0) {                //if the DO is greater than or equal to 6.0
        Serial.println("high");      //print "high" this is demonstrating that the Arduino is evaluating the DO as a number
      }                               //and not as a string
      if (DO <= 5.99) {              //if the DO is less than or equal to 5.99
        Serial.println("low");       //print "low" this is demonstrating that the Arduino is evaluating the DO
      }                               //as a number and not as a string
    }
  }
  sensorstring = "";                 //clear the string
  sensor_string_complete = false;    //reset the flag used to tell if we have received a completed string from the
  }
}
```

[Click here to download the \\*.ino file](#)