Introduction

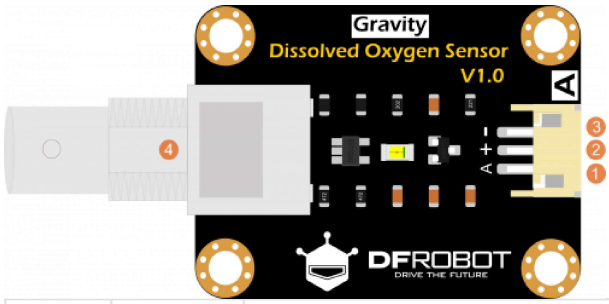
There's an old saying regarding keeping fish, "Good fish deserves good water". Good water quality is very important to aquatic organisms. Dissolved oxygen is one of the important parameters to reflect the water quality. Low dissolved oxygen in water will lead to difficulty in breathing for aquatic organisms, which may threaten the lives of these animals. We launched a new opensource dissolved oxygen sensor kit, which is compatible with Arduino. This product is used to measure the dissolved oxygen in water, which in turn reflects the water quality. It is widely applied in many water quality applications, such as aquaculture, environmental monitoring, natural science and so on. This sensor kit helps you quickly to build your own dissolved oxygen detector. The probe is a galvanic probe, with no need for polarization time and therefore is able to measure at any time. The filling solution and membrane cap is replaceable, leading to the low maintenance cost. The signal converter board is plug and play, and has the good compatibility. It can be easily integrated to any control or detecting system. This product is easy to use and has great compabilility. With open-source code and detailed tutorial provided, this product is very suitable for your water projects for detecting the dissolved oxygen concentration for aquatic organisms.

⚠	**The filling solution is 0.5 mol/L NaOH solution. You need to pour it in the membrane cap before use. Please use caution with this operation because the solution is corrosive. Please wear gloves! If the solution accidentally drops onto the skin, wash your skin with plenty of water immediately.** **The oxygen permeable membrane in the membrane cap is sensitive and fragile. Use caution when handling it. Fingernails and other sharp objects should be avoided.** **The DO sensor will consume a little oxygen during the measurement. Please gently stir the solution and let the oxygen to be distributed evenly in the water.**

Specification

- **Dissolved Oxygen Probe**
  - Type: Galvanic Probe
  - Detection Range: 0~20 mg/L
  - Temperature Range: 0~40 ℃
  - Response Time: Up to 98% full response, within 90 seconds (25℃)
  - Pressure Range: 0~50 PSI
  - Electrode Service Life: 1 year (normal use)
  - Maintenance Period:
    - Membrane Cap Replacement Period:
      - 1~2 months (in muddy water);
      - 4~5 months (in clean water)
    - Filling Solution Replacement Period: Once every month
  - Cable Length: 2 meters
  - Probe Connector: BNC
- **Signal Converter Board**
  - Supply Voltage: 3.3~5.5V
  - Output Signal: 0~3.0V
  - Cable Connector: BNC
  - Signal Connector: Gravity Analog Interface (PH2.0-3P)
  - Dimension: 42mm * 32mm/1.65 * 1.26 inches

Board Overview



No.	Label	Description
1	A	Analog Signal Output (0~3.0V)
2	+	VCC (3.3~5.5V)
3	-	GND
4	BNC	Probe Cable Connector

Tutorial

This tutorial will show how to use this dissolved oxygen sensor kit. The dissolved oxygen probe is a precision electrochemical sensor. Please pay attention to the usage details.

> **Before using the dissolved oxygen probe, 0.5 mol/L NaOH solution should be added into the membrane cap as the filling solution of the probe. As NaOH solution has strong corrosivity, protective gloves should be put on before handling the solution. If the solution accidentally drops onto the skin, wash your skin with plenty of water immediately.**
> **The oxygen permeable membrane in the membrane cap is sensitive and vulnerable. Use caution when handling with it. Fingernail and other sharp objects should be avoided.**
> **During the measuring process, the oxygen probe will consume a little oxygen. You need to gently stir the water and let the oxygen to be distributed evenly in water. On the other hand, do not stirring violently to prevent the oxygen in the air from quickly entering into the water.**

Requirements

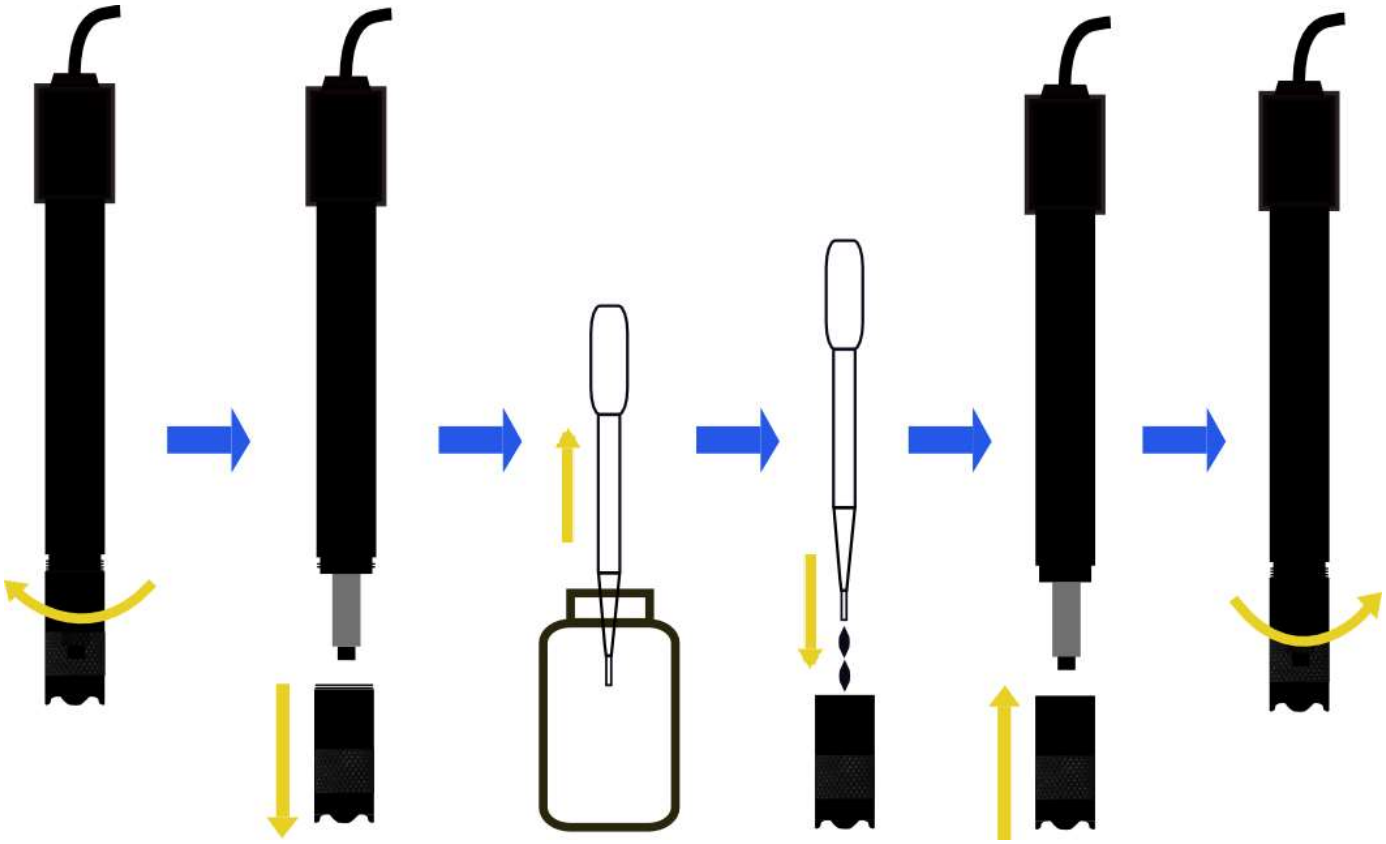
- **Hardware**
  - Dissolved Oxygen Probe (With Membrane Cap) x 1
  - 0.5mol/L NaOH Solution x 1
  - DFRduino UNO R3 (https://www.dfrobot.com/product-838.html) (or similar) x 1
  - Dissolved Oxygen Signal Converter Board x 1
  - Analog Cable (3Pin) x 1
- **Software**
  - Arduino IDE (Version requirements: V1.0.x, V1.6.x or V1.8.x), Click to Download Arduino IDE from Arduino® (https://www.arduino.cc/en/Main/Software)

Prepare the Probe

For a new dissolved oxygen probe, 0.5 mol/L NaOH solution should be added into the membrane cap first as the filling solution. If the probe has been used for some time and the error grows greatly, it is time to change the filling solution. The following tutorial details how to fill the probe with the NaOH solution.

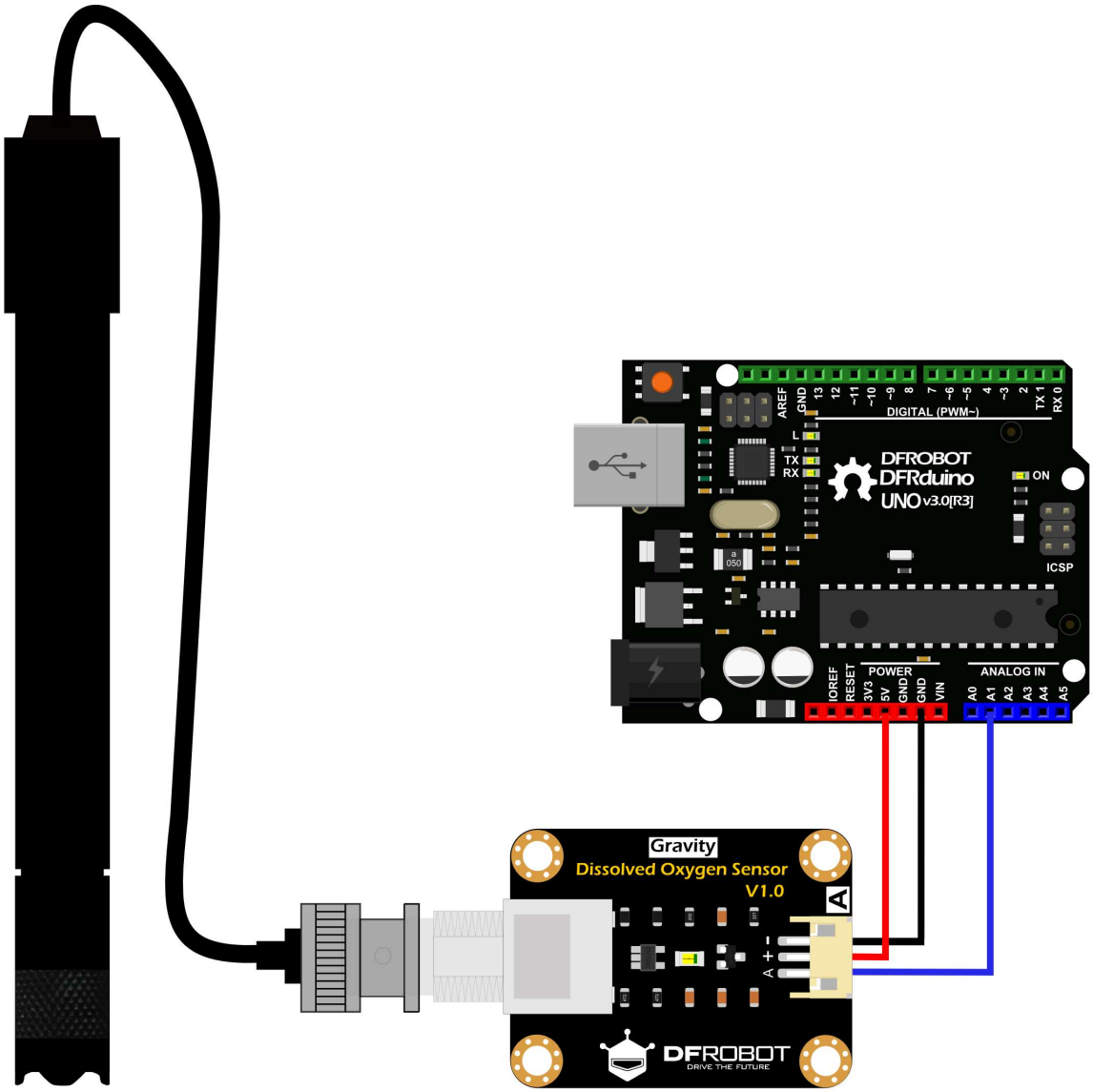
Unscrew the membrane cap from the probe and fill about 2/3 volume of the cap with 0.5 mol/L NaOH solution. Make sure the probe is in vertical position with respect to the horizontal plane. Carefully screw the cap back to the probe. It would be nice if a little bit solution overflows out of the cap to ensure the probe is fully filled with NaOH solution.

- When screwing the cap back to the probe, the probe should be in vertical position with respect to the horizontal plane to avoid creating bubbles in the filling solution.
- If the cap is fully filled with NaOH solution, there will be too much solution overflowing when screwing the cap back to the probe. If the filling solution is too little, bubbles may be created inside the cap. In sum, the best way is to fill about 2/3 volume of the cap. A little bit overflow when screwing the cap back to the probe is okay.
- Clean the overflowed solution with tissue.
- Screw the NaOH solution bottle after every use to prevent the CO2 in the air from affecting the solution.



Connection Diagram

When the probe is filled with NaOH solution, it needs to be calibrated. Before calibration, please connect the probe as shown in the the following diagram. Connect the probe to BNC connector on the signal converter board. Connect the board to the analog input of Arduino main-board.



Probe Calibration

If this is the first time you use the probe or the probe has been used for some time, the probe needs to be calibrated for accuracy. Common calibration methods: single point calibration and double points calibration. The single point calibration calibrates the probe with saturated dissolved oxygen. The double points calibration calibrates the probe with both saturated dissolved oxygen and zero dissolved oxygen. In most cases, the single point calibration is good enough and convenient. The following tutorial details the process of single point calibration. **1. Upload the sample code to Arduino and open the serial monitor. The program will refresh and print the dissolved oxygen content on the screen every second. 2. Dip the probe into the saturated dissolved oxygen water and gently stir the water. Check the dissolved oxygen readings and wait for the readings to be stable.**

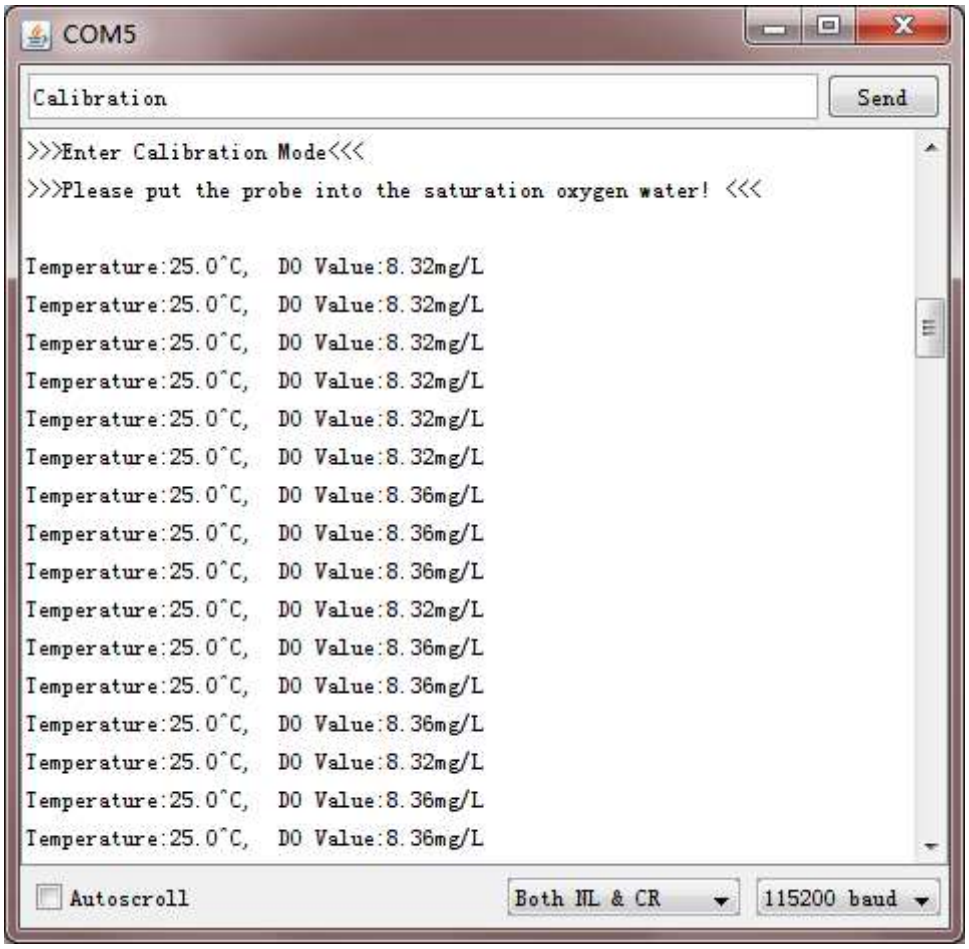
- Usually, it takes over 1 minute for the readings to become stable, because the electrochemical reaction in the probe takes time

electrochemical reaction in the probe takes time.

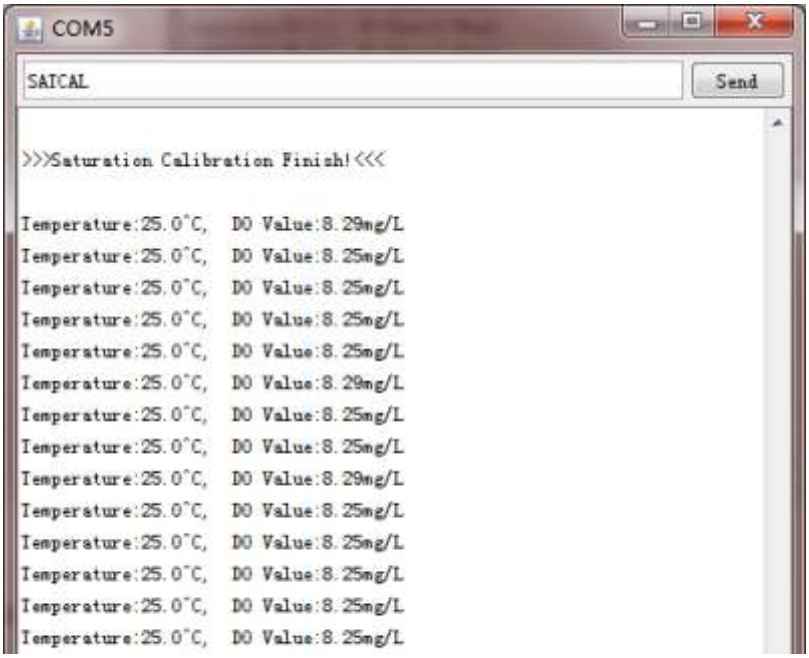
- If saturated dissolved oxygen water is not available, the air will be a replacement. Dip the probe into the water and stir it for several times to wet the permeable membrane of the cap. Expose the probe to the air for over 1 minute.

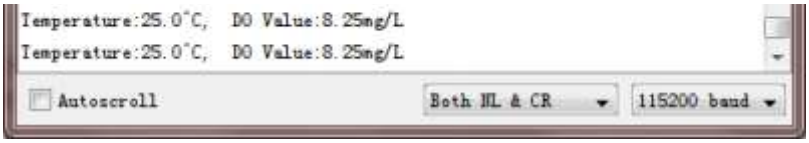
  3. Wait for the dissolved oxygen readings to be stable. Then you can do the calibration. The steps are as followed.

- Enter "Calibration" in the serial monitor to enter into calibration mode.

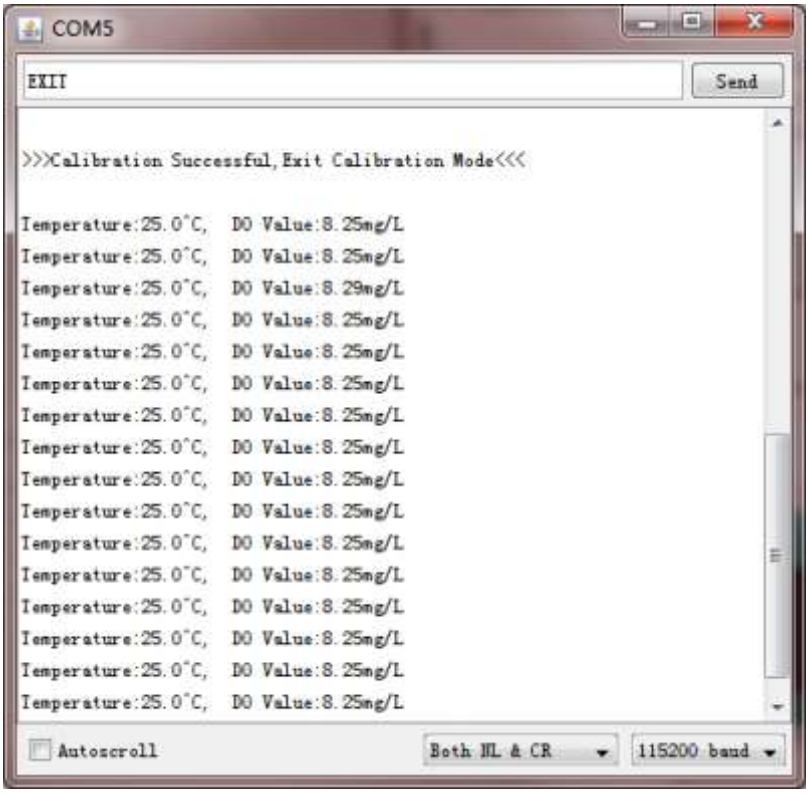


  - Enter "SATCAL" for dissolved oxygen calibration. The program will print whether the calibration is successful or not.





- After the calibration, enter “EXIT” to exit calibration mode.



4. After the process stated above, the saturated dissolved oxygen calibration has completed. The probe is now ready for measurement.

Sample Code

```
/*************************************************
 DFRobot Gravity: Analog Dissolved Oxygen Sensor / Meter Kit for Arduino
 <https://www.dfrobot.com/wiki/index.php/Gravity:_Analog_Dissolved_Oxygen_Sensor_SKU:SEN02

 *************************************************
 This example reads the concentration of dissolved oxygen in water.
 The saturated oxygen calibration is available by UART commends with NL & CR:
 calibration  ----  enter the calibration mode
 satcal       ----  calibrate the parameters with saturated oxygen value
 exit         ----  exit the calibration mode

 Created 2017-5-22
 By Jason <jason.ling@dfrobot.com@dfrobot.com>

 GNU Lesser General Public License.
 See <http://www.gnu.org/licenses/> for details.
 All above must be included in any redistribution
 ************************************************/

 /***********Notice and Trouble shooting***************
 1. This code is tested on Arduino Uno and Leonardo with Arduino IDE 1.0.5 r2 and 1.8.2.
 2. More details, please click this link: <https://www.dfrobot.com/wiki/index.php/Gravity:
 ************************************************/

#include <avr/pgmspace.h>
#include <EEPROM.h>

#define DoSensorPin  A1    //dissolved oxygen sensor analog output pin to arduino mainboar
#define VREF 5000    //for arduino uno, the ADC reference is the AVCC, that is 5000mV(TYP)
float doValue;      //current dissolved oxygen value, unit; mg/L
float temperature = 25;    //default temperature is 25^C, you can use a temperature sensor

#define EEPROM_write(address, p) {int i = 0; byte *pp = (byte*)&(p);for(; i < sizeof(p); i
#define EEPROM_read(address, p)  {int i = 0; byte *pp = (byte*)&(p);for(; i < sizeof(p); i

#define ReceivedBufferLength 20
char receivedBuffer[ReceivedBufferLength+1];    // store the serial command
byte receivedBufferIndex = 0;

#define SCOUNT  30           // sum of sample point
int analogBuffer[SCOUNT];    //store the analog value in the array, readed from ADC
int analogBufferTemp[SCOUNT];
int analogBufferIndex = 0,copyIndex = 0;

#define SaturationDoVoltageAddress 12          //the address of the Saturation Oxygen volt
#define SaturationDoTemperatureAddress 16      //the address of the Saturation Oxygen temp
float SaturationDoVoltage,SaturationDoTemperature;
float averageVoltage;
```

```
const float SaturationValueTab[41] PROGMEM = {      //saturation dissolved oxygen concentr
14.46, 14.22, 13.82, 13.44, 13.09,
12.74, 12.42, 12.11, 11.81, 11.53,
11.26, 11.01, 10.77, 10.53, 10.30,
10.08, 9.86,  9.66,  9.46,  9.27,
9.08,  8.90,  8.73,  8.57,  8.41,
8.25,  8.11,  7.96,  7.82,  7.69,
7.56,  7.43,  7.30,  7.18,  7.07,
6.95,  6.84,  6.73,  6.63,  6.53,
6.41,
};

void setup()
{
    Serial.begin(115200);
    pinMode(DoSensorPin,INPUT);
    readDoCharacteristicValues();      //read Characteristic Values calibrated from the EE
}

void loop()
{
   static unsigned long analogSampleTimepoint = millis();
   if(millis()-analogSampleTimepoint > 30U)     //every 30 milliseconds,read the analog va
   {
     analogSampleTimepoint = millis();
     analogBuffer[analogBufferIndex] = analogRead(DoSensorPin);    //read the analog value
     analogBufferIndex++;
     if(analogBufferIndex == SCOUNT)
         analogBufferIndex = 0;
   }

   static unsigned long tempSampleTimepoint = millis();
   if(millis()-tempSampleTimepoint > 500U)  // every 500 milliseconds, read the temperatur
   {
      tempSampleTimepoint = millis();
      //temperature = readTemperature();  // add your temperature codes here to read the t
   }

   static unsigned long printTimepoint = millis();
   if(millis()-printTimepoint > 1000U)
   {
      printTimepoint = millis();
      for(copyIndex=0;copyIndex<SCOUNT;copyIndex++)
      {
        analogBufferTemp[copyIndex]= analogBuffer[copyIndex];
      }
      averageVoltage = getMedianNum(analogBufferTemp,SCOUNT) * (float)VREF / 1024.0; // re
      Serial.print(F("Temperature:"));
      Serial.print(temperature,1);
      Serial.print(F("^C"));
      doValue = pgm_read_float_near( &SaturationValueTab[0] + (int)(SaturationDoTemperatur
      Serial.print(F(",  DO Value:"));
      Serial.print(doValue,2);
      Serial.println(F("mg/L"));
   }
```

```
    }

    if(serialDataAvailable() > 0)
    {
       byte modeIndex = uartParse();  //parse the uart command received
       doCalibration(modeIndex);    // If the correct calibration command is received, the
    }

}

boolean serialDataAvailable(void)
{
  char receivedChar;
  static unsigned long receivedTimeOut = millis();
  while ( Serial.available() > 0 )
  {
    if (millis() - receivedTimeOut > 500U)
    {
      receivedBufferIndex = 0;
      memset(receivedBuffer,0,(ReceivedBufferLength+1));
    }
    receivedTimeOut = millis();
    receivedChar = Serial.read();
    if (receivedChar == '\n' || receivedBufferIndex == ReceivedBufferLength)
    {
    receivedBufferIndex = 0;
    strupr(receivedBuffer);
    return true;
    }else{
        receivedBuffer[receivedBufferIndex] = receivedChar;
        receivedBufferIndex++;
    }
  }
  return false;
}

byte uartParse()
{
    byte modeIndex = 0;
    if(strstr(receivedBuffer, "CALIBRATION") != NULL)
        modeIndex = 1;
    else if(strstr(receivedBuffer, "EXIT") != NULL)
        modeIndex = 3;
    else if(strstr(receivedBuffer, "SATCAL") != NULL)
        modeIndex = 2;
    return modeIndex;
}

void doCalibration(byte mode)
{
    char *receivedBufferPtr;
    static boolean doCalibrationFinishFlag = 0,enterCalibrationFlag = 0;
    float voltageValueStore;
    switch(mode)
    {
      case 0:
```

```
      if(enterCalibrationFlag)
         Serial.println(F("Command Error"));
      break;

      case 1:
      enterCalibrationFlag = 1;
      doCalibrationFinishFlag = 0;
      Serial.println();
      Serial.println(F(">>>Enter Calibration Mode<<<"));
      Serial.println(F(">>>Please put the probe into the saturation oxygen water! <<<"));
      Serial.println();
      break;

     case 2:
      if(enterCalibrationFlag)
      {
         Serial.println();
         Serial.println(F(">>>Saturation Calibration Finish!<<<"));
         Serial.println();
         EEPROM_write(SaturationDoVoltageAddress, averageVoltage);
         EEPROM_write(SaturationDoTemperatureAddress, temperature);
         SaturationDoVoltage = averageVoltage;
         SaturationDoTemperature = temperature;
         doCalibrationFinishFlag = 1;
      }
      break;

        case 3:
        if(enterCalibrationFlag)
        {
            Serial.println();
            if(doCalibrationFinishFlag)
               Serial.print(F(">>>Calibration Successful"));
            else
              Serial.print(F(">>>Calibration Failed"));
            Serial.println(F(",Exit Calibration Mode<<<"));
            Serial.println();
            doCalibrationFinishFlag = 0;
            enterCalibrationFlag = 0;
        }
        break;
    }
}

int getMedianNum(int bArray[], int iFilterLen)
{
      int bTab[iFilterLen];
      for (byte i = 0; i<iFilterLen; i++)
      {
      bTab[i] = bArray[i];
      }
      int i, j, bTemp;
      for (j = 0; j < iFilterLen - 1; j++)
      {
      for (i = 0; i < iFilterLen - j - 1; i++)
```

```
        {
      if (bTab[i] > bTab[i + 1])
          {
      bTemp = bTab[i];
          bTab[i] = bTab[i + 1];
      bTab[i + 1] = bTemp;
       }
    }
    }
    if ((iFilterLen & 1) > 0)
  bTemp = bTab[(iFilterLen - 1) / 2];
    else
  bTemp = (bTab[iFilterLen / 2] + bTab[iFilterLen / 2 - 1]) / 2;
    return bTemp;
}

void readDoCharacteristicValues(void)
{
    EEPROM_read(SaturationDoVoltageAddress, SaturationDoVoltage);
    EEPROM_read(SaturationDoTemperatureAddress, SaturationDoTemperature);
    if(EEPROM.read(SaturationDoVoltageAddress)==0xFF && EEPROM.read(SaturationDoVoltageAdd
    {
      SaturationDoVoltage = 1127.6;   //default voltage:1127.6mv
      EEPROM_write(SaturationDoVoltageAddress, SaturationDoVoltage);
    }
    if(EEPROM.read(SaturationDoTemperatureAddress)==0xFF && EEPROM.read(SaturationDoTemper
    {
      SaturationDoTemperature = 25.0;   //default temperature is 25^C
      EEPROM_write(SaturationDoTemperatureAddress, SaturationDoTemperature);
    }
}
```

FAQ

Q1.How to configure the saturated dissolved oxygen water by myself ?

A1. You can pump air into water for about 20 minutes to saturate the water with oxygen to obtain 100% dissolved oxygen standard liquid.

Q2.How to make zero dissolved oxygen water ?

A2.Add sodium sulfite(Na2SO3) into water until it is saturated. This can consume all the oxygen in the water to obtain the zero dissolved oxygen liquid.

Q3.How to store the probe ?

Q3.How to store the probe ?

A3.
1. Short time (over night to a week): dip the probe into purified water or deionized water to prevent the filling solution from evaporating. Disconnect the probe from the main instrument every time when it is not in use.
2. Long time: (over a week): unscrew the cap from the probe and wash the electrodes core (cathode: platinum, anode: lead) and wash the cap with purified water or deionized water. Dry all the components with tissue. Screw the cap back to the probe without adding any filling solution to prevent the anode from being consumed. Put all the components back into the package.

Q4.How to make 0.5 mol/L NaOH solution?

A4.You need to purchase the NaOH solution first, and add 1~2 drops glycerinum to the NaOH solution for every 100mL. Only by adding the NaOH solution into the probe can it be ready for use.

Q5.What problems would I usually meet ? How to solve it ?

A5.
1. If the readings of zero dissolved oxygen liquid is not zero or close to zero, you can polish the cathode of the probe.
2. If the readings is not within normal range or the readings drifts, please check the membrane on the cap. If there are cracks, holes or contaminations on the membrane, please change the cap.

For any questions, advice or cool ideas to share, please visit the DFRobot Forum (https://www.dfrobot.com/forum/).

More Documents

- Schematic V1.0 (https://github.com/DFRobot/Gravity_Analog_Dissolved_Oxygen_Sensor/raw/master/Schematic%20V1.0.pdf)
- Layout with Dimension (https://github.com/DFRobot/Gravity_Analog_Dissolved_Oxygen_Sensor/raw/master/Layout%20with%20Dimension%20V1.0.pdf)
- Probe Dimension

(https://github.com/DFRobot/Gravity_Analog_Dissolved_Oxygen_Sensor/raw/master/Probe%20Dimension.jpg)

- SVG files (https://github.com/DFRobot/Gravity_Analog_Dissolved_Oxygen_Sensor/raw/master/SVG%20files.zip)
- TP5551 Datasheet (https://github.com/DFRobot/Gravity_Analog_Dissolved_Oxygen_Sensor/raw/master/TP5551-5552-5554_REV1.1.pdf)
- Github Repository (https://github.com/DFRobot/Gravity_Analog_Dissolved_Oxygen_Sensor)

Get **Gravity: Analog Dissolved Oxygen Sensor** (https://www.dfrobot.com/product-1628.html) from DFRobot Store or **DFRobot Distributor**. (https://www.dfrobot.com/index.php?route=information/distributorslogo)

Turn to the Top