

# Набор СкарТ

## УМНЫЙ ДОМ

Представляем Вашему вниманию второй продукт серии СкарТ- ЛАБ "Умный дом"

Набор СкарТ ЛАБ УМНЫЙ ДОМ предназначен для тех, кто уже прошел первый этап ознакомления с аппаратными и программными модулями Ардуино. Для тех, кто уже позажигал светодиоды, попищал пищалкой и хочет понять для себя — ЗАЧЕМ ? Какая от этого польза ?

Мы , разрабатывая данный комплект , стремились дать Вам в руки инструмент, который позволит применить на практике полученные знания. Соберите комплект. Установите на нем датчики и настройте их работу.

На игровом поле нарисованы разные помещения. Жилые и служебные. Вы сами должны решить, какие датчики и куда нужно установить.

**Все датчики , которые идут в комплекте , Вы можете снять и разместить в Вашей реальной квартире.**

Обратите внимание, что для включения Элементов мы положили в набор блоки реле, которые могут коммутировать нагрузку с напряжением до 220В. (ВНИМАНИЕ !!! Работу с напряжением более 24 вольт необходимо проводить под руководством взрослых, имеющими необходимый уровень допуска по электробезопасности !) Это сделано для того, что бы Вы смогли снять датчики с Игрового поля и расположить их в Вашей комнате, доме или квартире.

### Состав набора СкарТ-ЛАБ "Умный дом"

Игровое поле

Электронная плата , совместимая с Ардуино УНО

Блок реле x4

Блок реле x1

Блок питания 12В (x2 блок для 4-х батарей АА)

Соединительный провода П+М

Соединительные провода

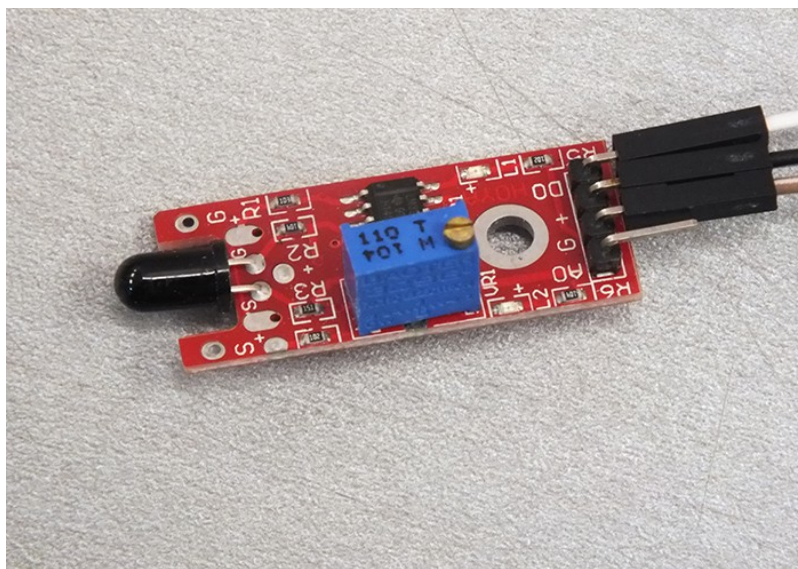
Светодиодная RGB-лента — 20 см.

Светодиодная лента белого свечения - 50 см

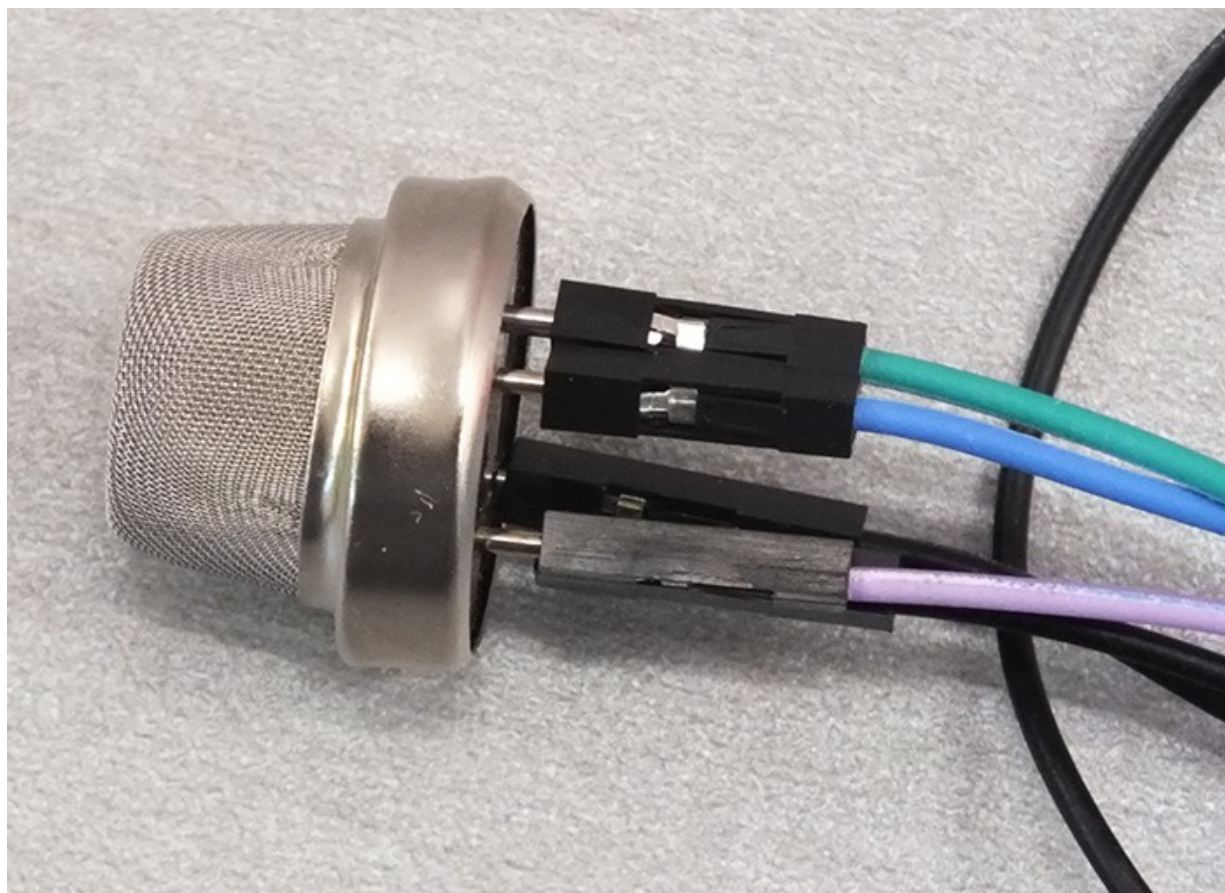
**ВНИМАНИЕ !!! ДЛЯ СБОРКИ КОМПЛЕКТА ВАМ ПОНАДОБИТСЯ ПАЯЛЬНИК И ПАЯЛЬНЫЕ ПРИНАДЛЕЖНОСТИ, ДАННЫЕ ИНСТРУМЕНТЫ В КОМПЛЕКТ ПОСТАВКИ НЕ ВХОДЯТ!**

### Набор датчиков

Датчик огня - сигнализирует в случае обнаружения открытого пламени.

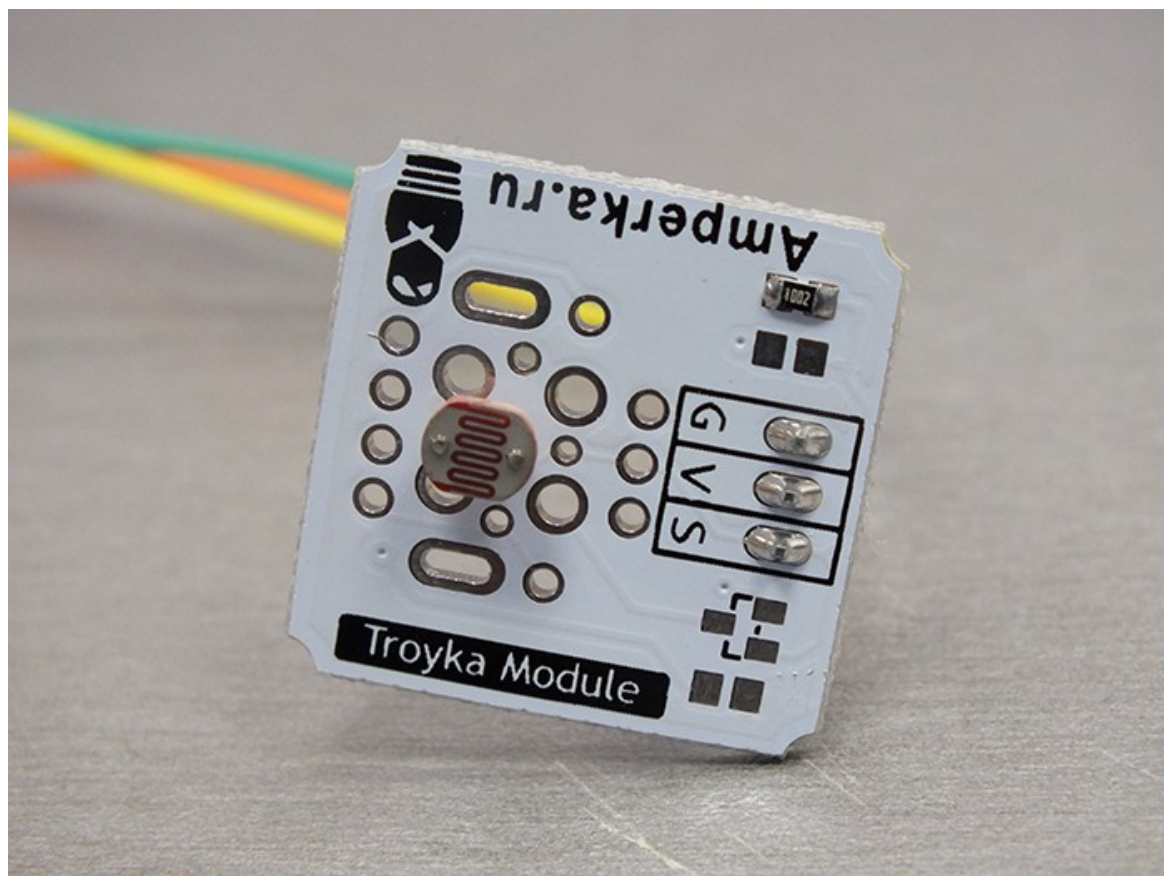


Датчик дыма - сигнализирует о задымлении помещения..

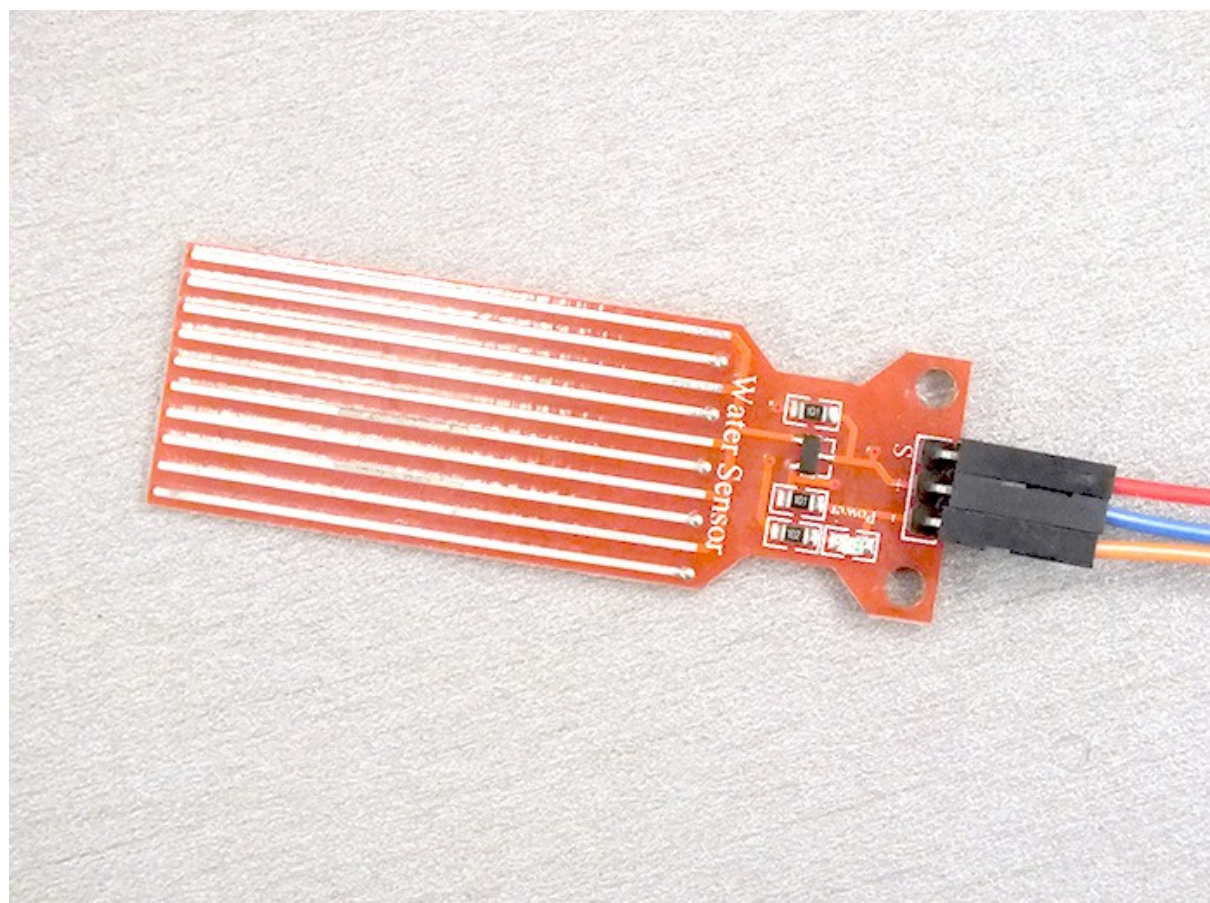




Датчик освещенности - показывает степень освещенности в помещении

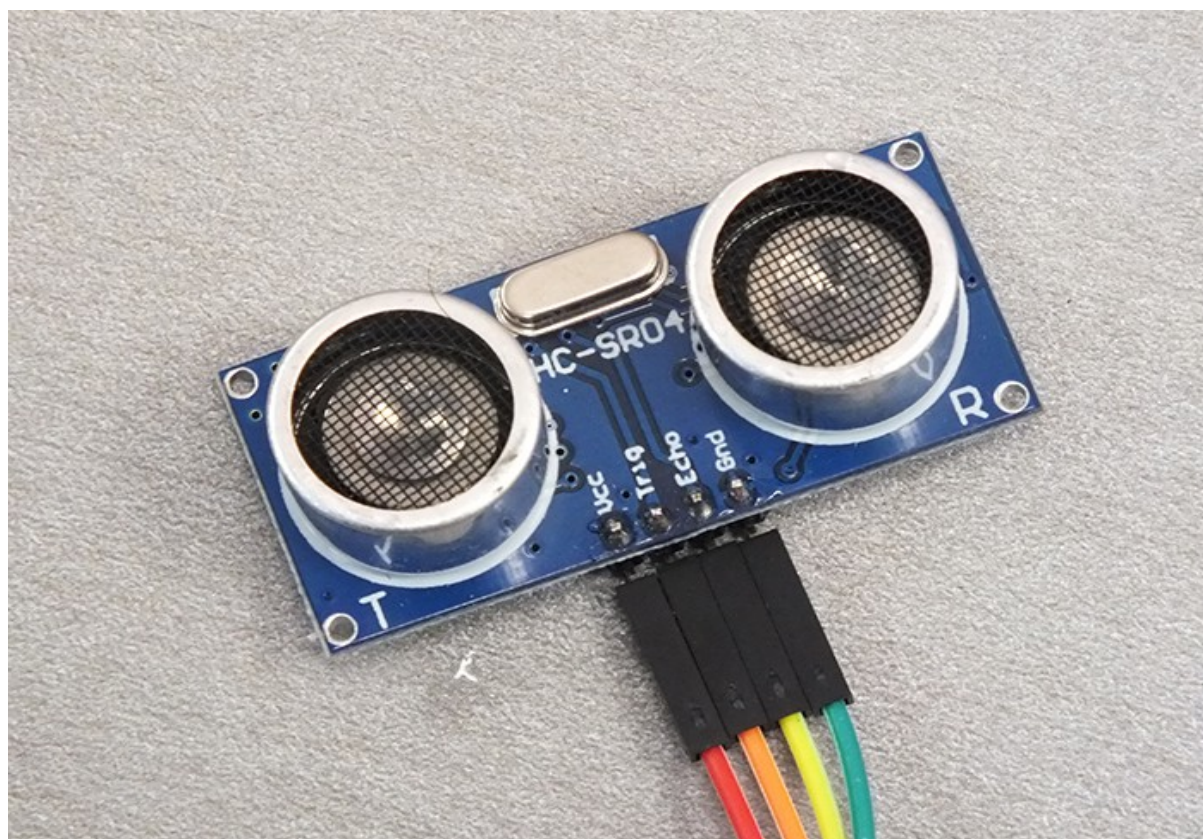


Датчик протечек - сигнализирует о появлении жидкости на полу.

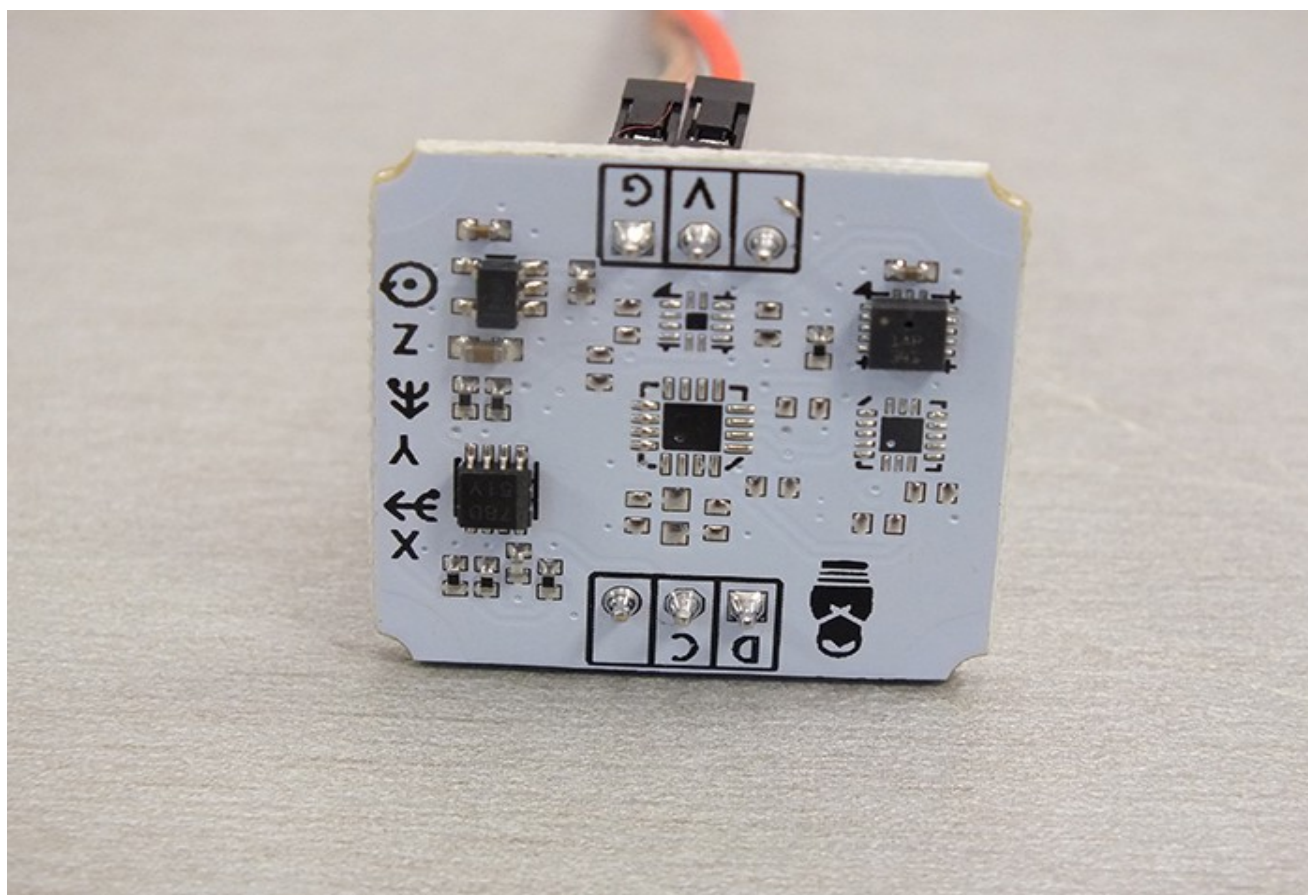




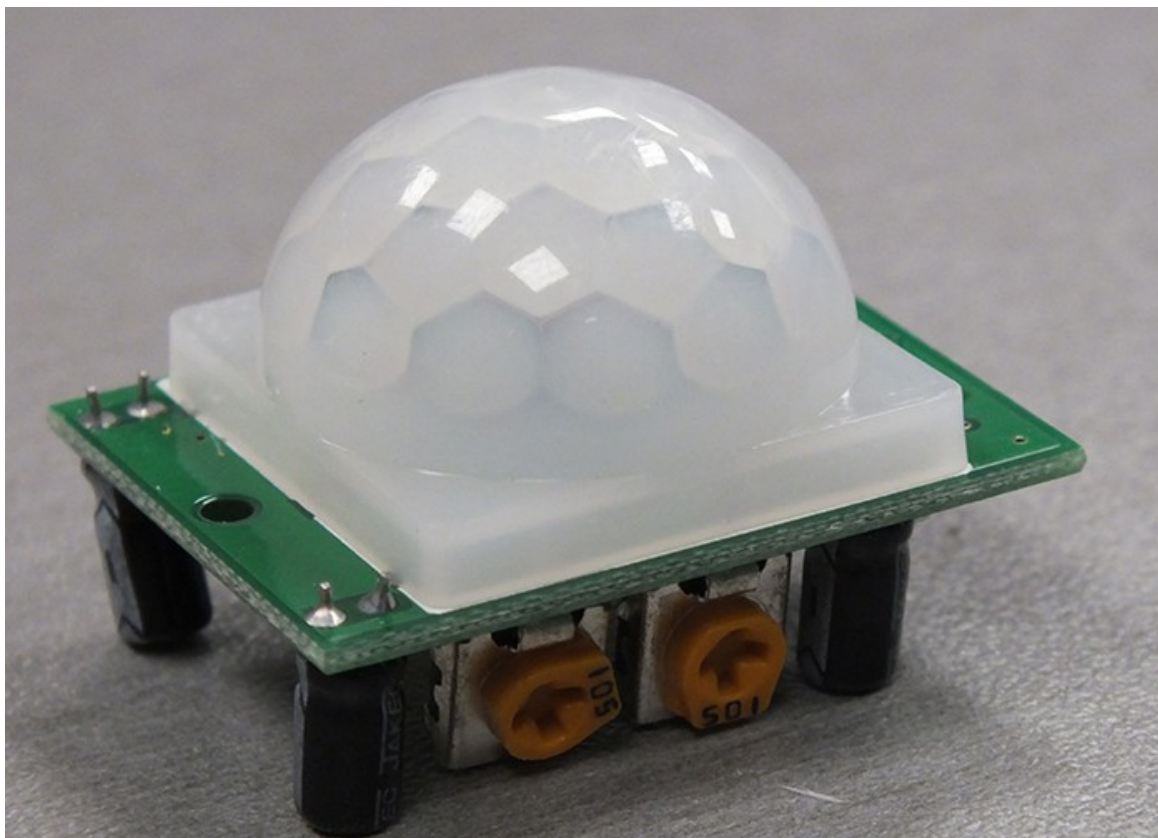
Датчик расстояния ультразвуковой (сонар) - показывает расстояние до препятствий



Датчик давления — показывает атмосферное давление, температуру и высоту/глубину над уровнем моря.



Датчик движения — реагирует на движение теплых предметов (человека, животных)



### Что у Вас должно получиться

Полностью настроенная система должна активно реагировать на изменения в окружающей среде, а именно:

- При появлении в зоне действия датчика огня открытого пламени должен включиться Тревожный свет.
- При задымлении датчика дыма должен включиться Тревожный свет.
- Датчик расстояния (сонар) дает команду на включение прожекторов и включает Тревожный свет.
- Датчик температуры — программируется на два режима — комфорт и противопожарная безопасность. В режиме комфорт Вы выставляете температуру в помещении, которую желаете получить. При понижении этой температуры должен включиться Обогреватель, В случае, если температура в помещении увеличится например до 45 градусов, то должен включиться Тревожный свет.
- При срабатывании датчика протечек, должен включиться Тревожный свет. Для срабатывания датчика нужно намочить рабочие контакты датчика.
- При срабатывании датчика движения должен включиться свет

Что Вам понадобится для сборки ?

1. Паяльник и паяльные принадлежности



2. Компьютер с ОС Windows или Linux
3. Комплект тонких проводов суммарной длиной около 10 м.
4. Нож, кусачки, пассатижи

**Подготовим игровое поле.** Для придания жесткости нужна основа. В качестве основы можно взять лист ватмана формата A1, кусок старых обоев или картон листовый или от упаковочной коробки. Закрепите банер на листе ватмана или другого материала. Придайте ему форму цилиндра. Закрепите края. Если используется упаковочный картон, то подготовьте лист размером 75x55 см. Закрепите банер на нем.

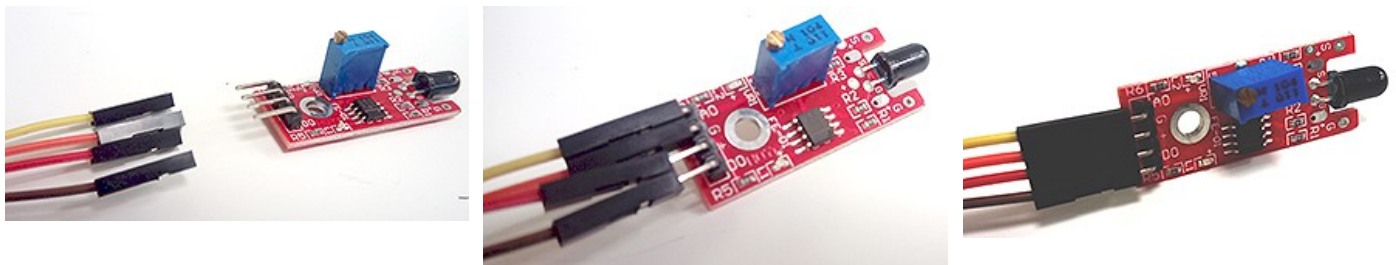
Основа готова.

## Электроника

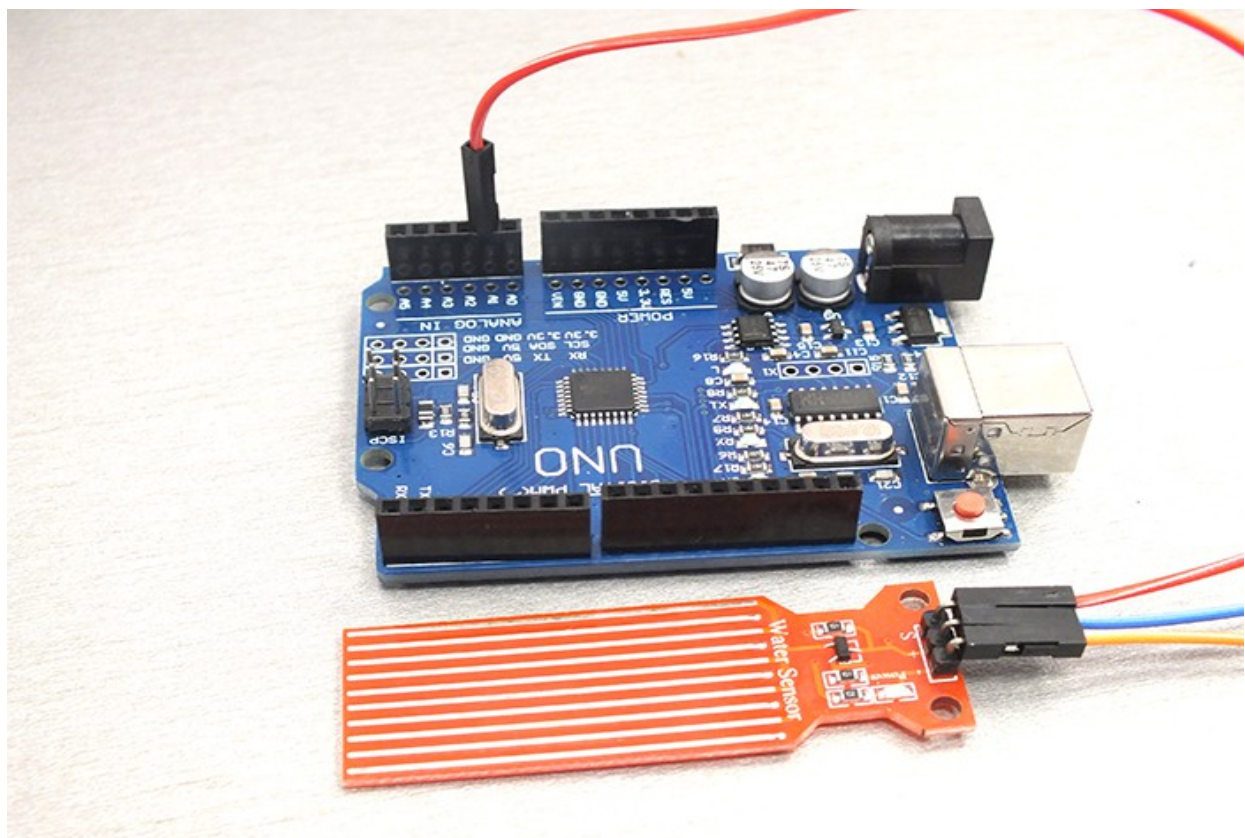
Для начала давайте соберем наладочную схему и проверим её работоспособность.

Для этого совершим следующие действия

1. Возьмем Электронную плату, датчики, провода, входящие в комплект.
2. Оденьте провода наконечниками на выходы датчиков

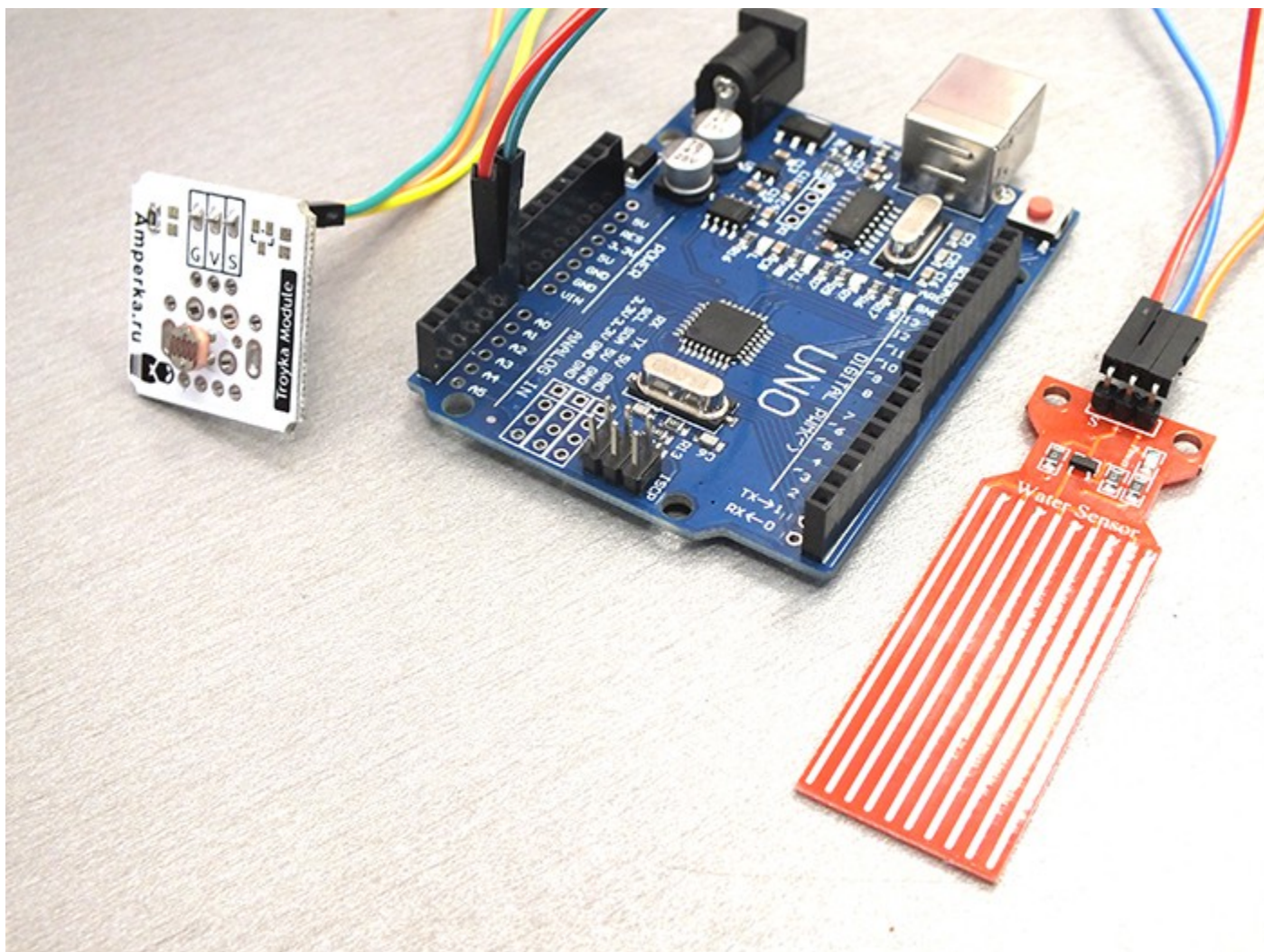


3. Подсоедините датчик воды к Электронной плате воткнув сигнальный провод в разъем A1

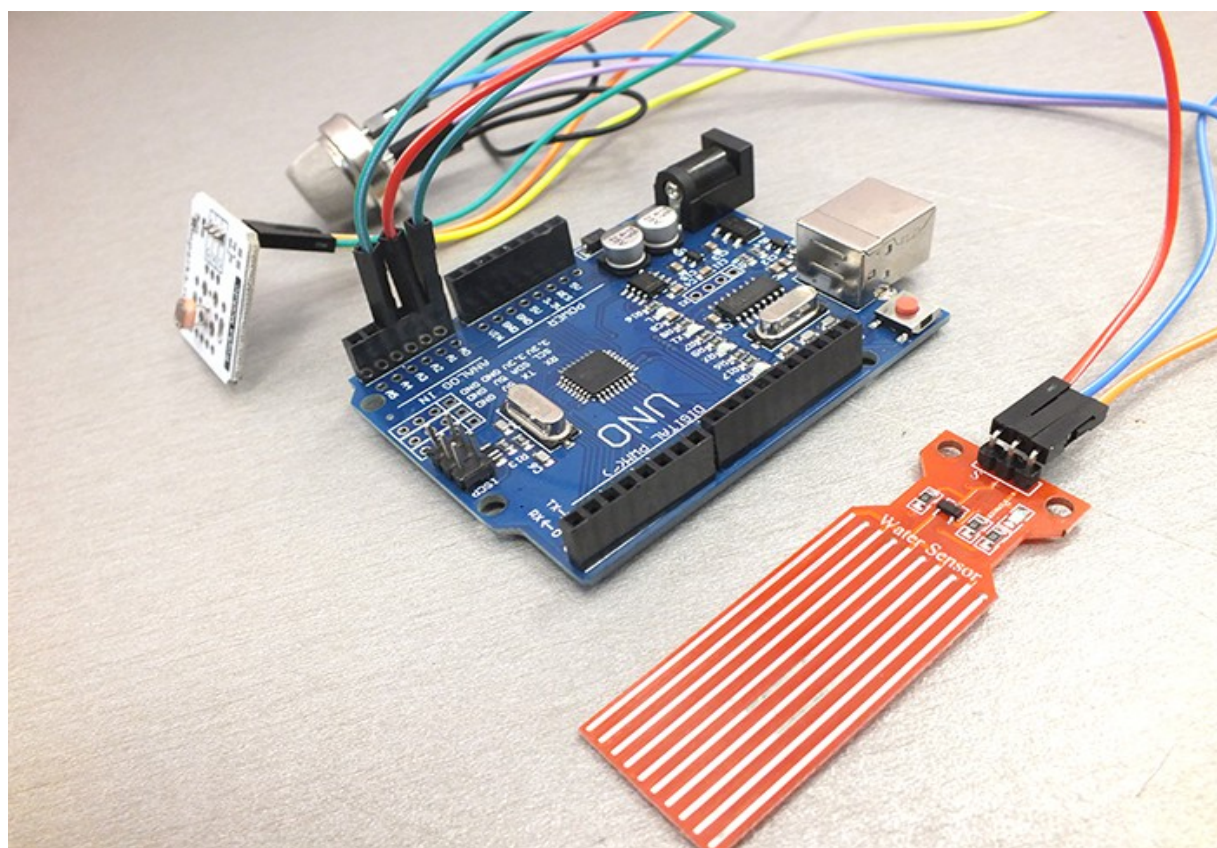




4. Подсоедините датчик освещенности к Электронной плате воткнув сигнальный провод в разъем A0

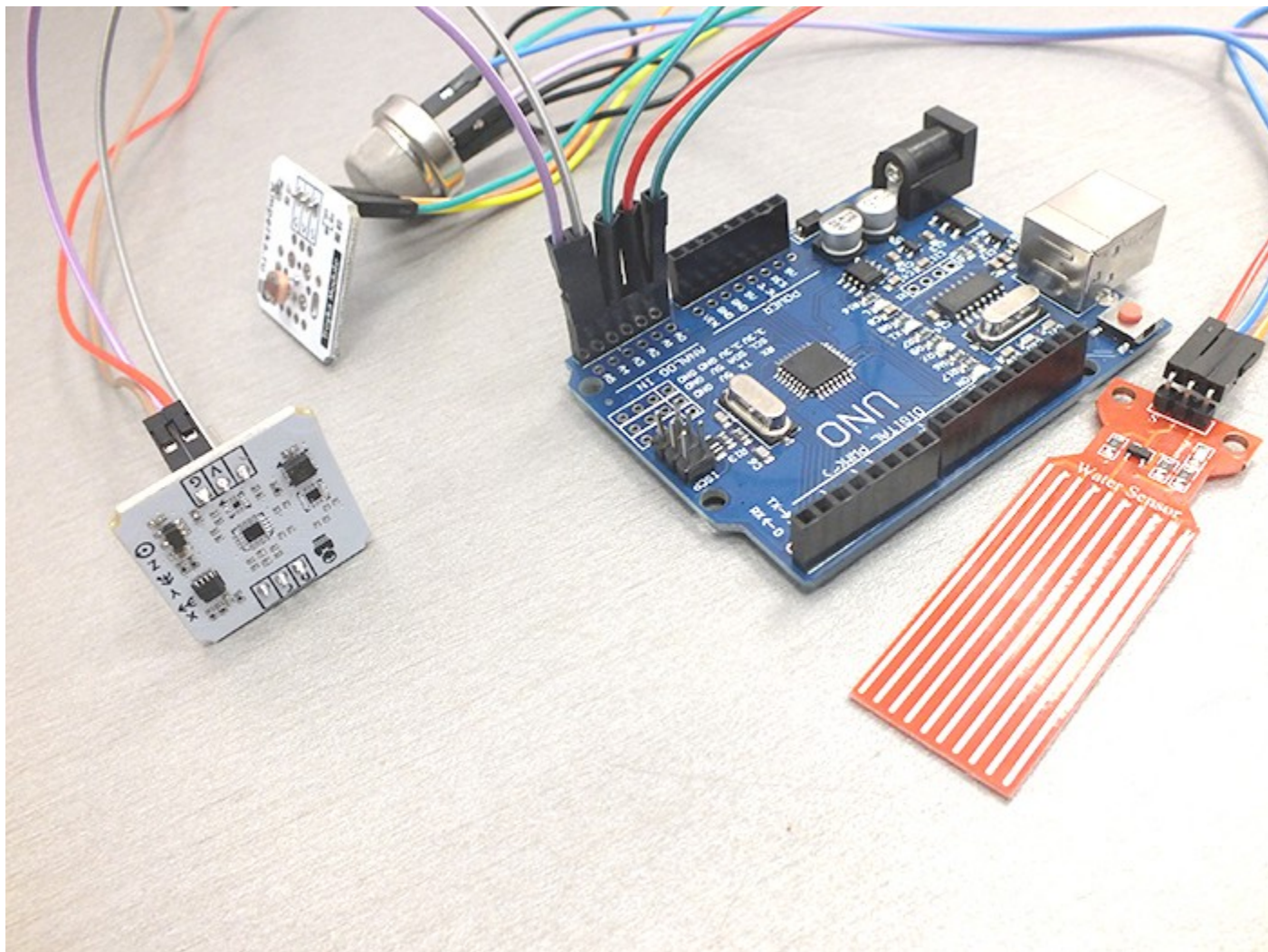


5. Подсоедините датчик газа к Электронной плате воткнув сигнальный провод в разъем A3

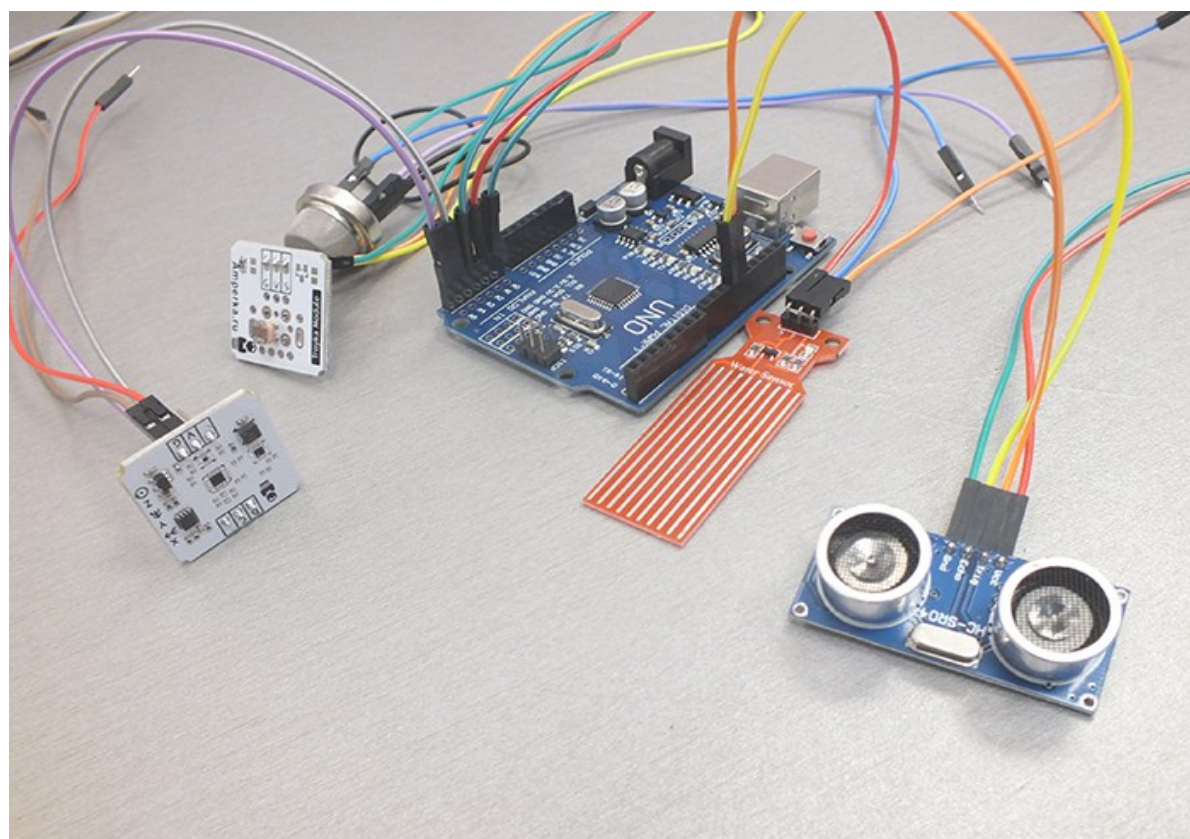




6. Подсоедините Барометр к Электронной плате воткнув провод D в разъем A4, а C в разъем A5

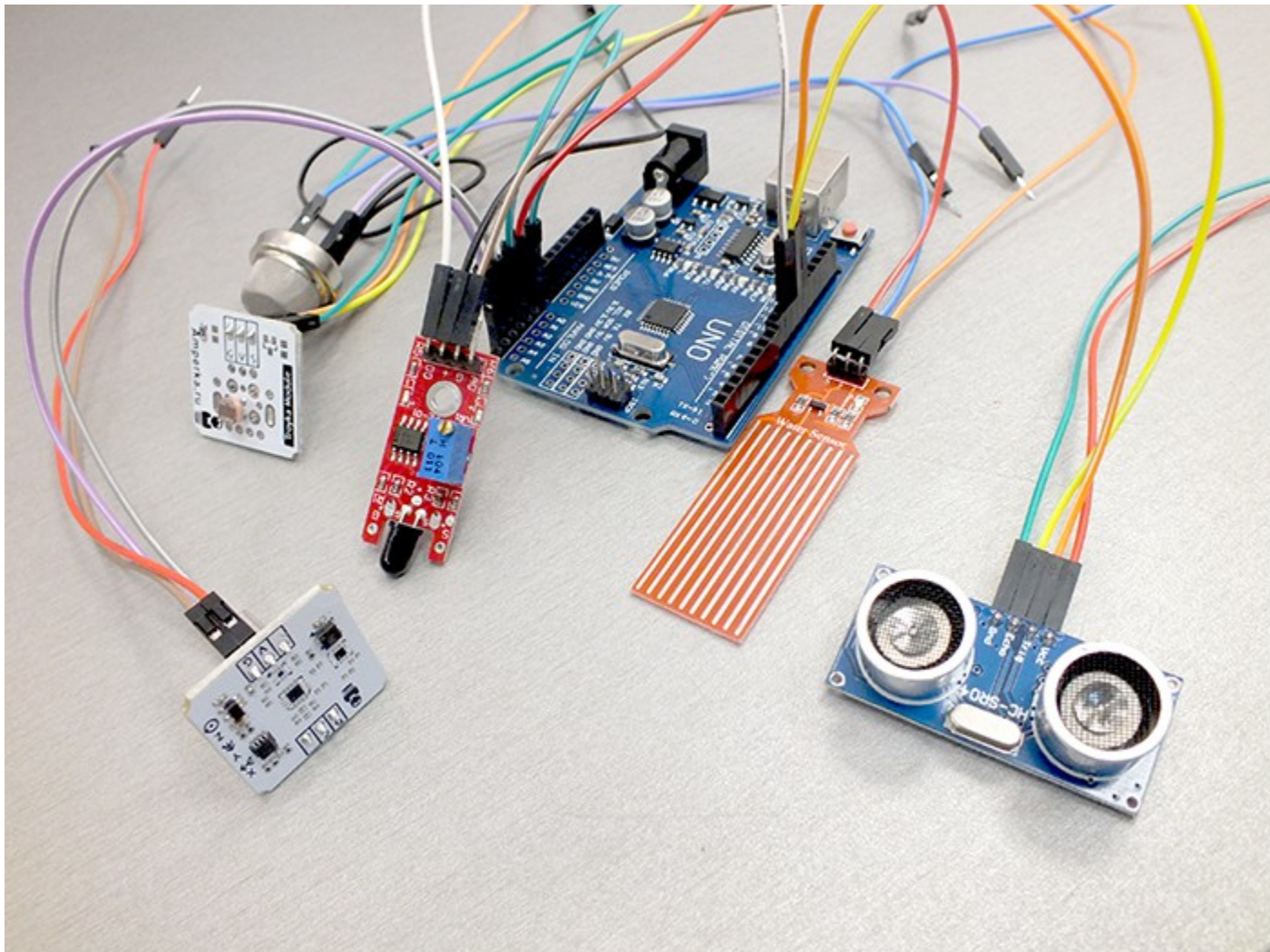


7. Подсоедините Сонар к Электронной плате воткнув провод Trig в разъем D12, а Echo в разъем D11

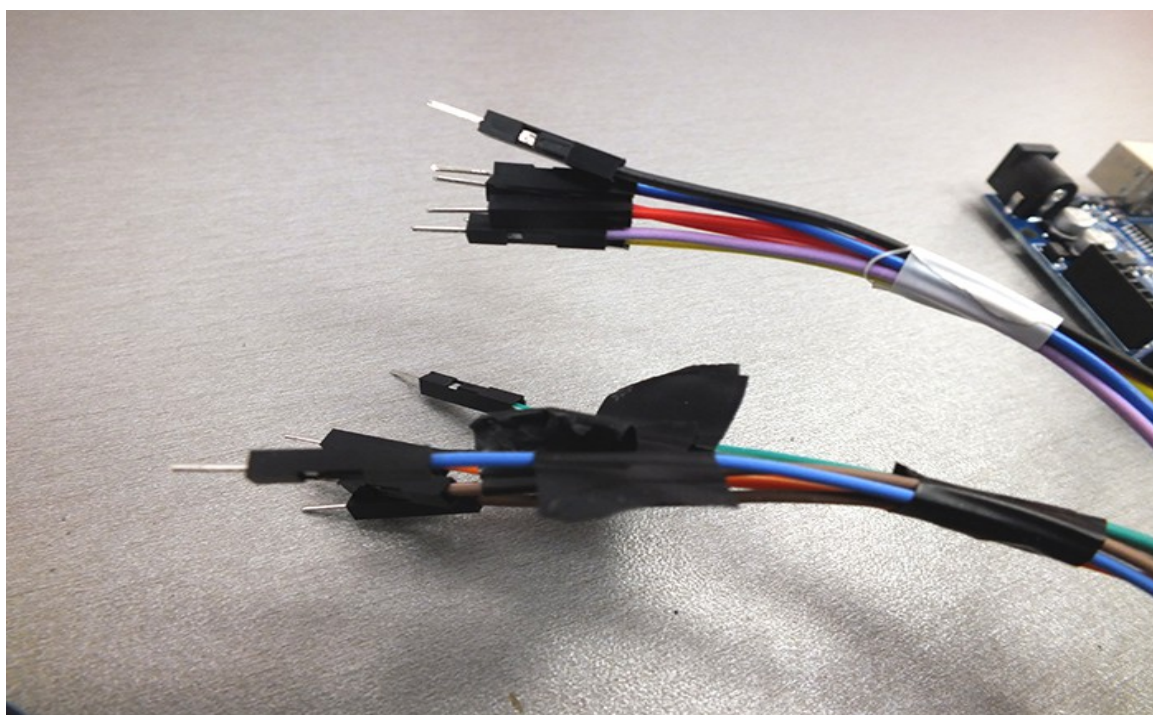




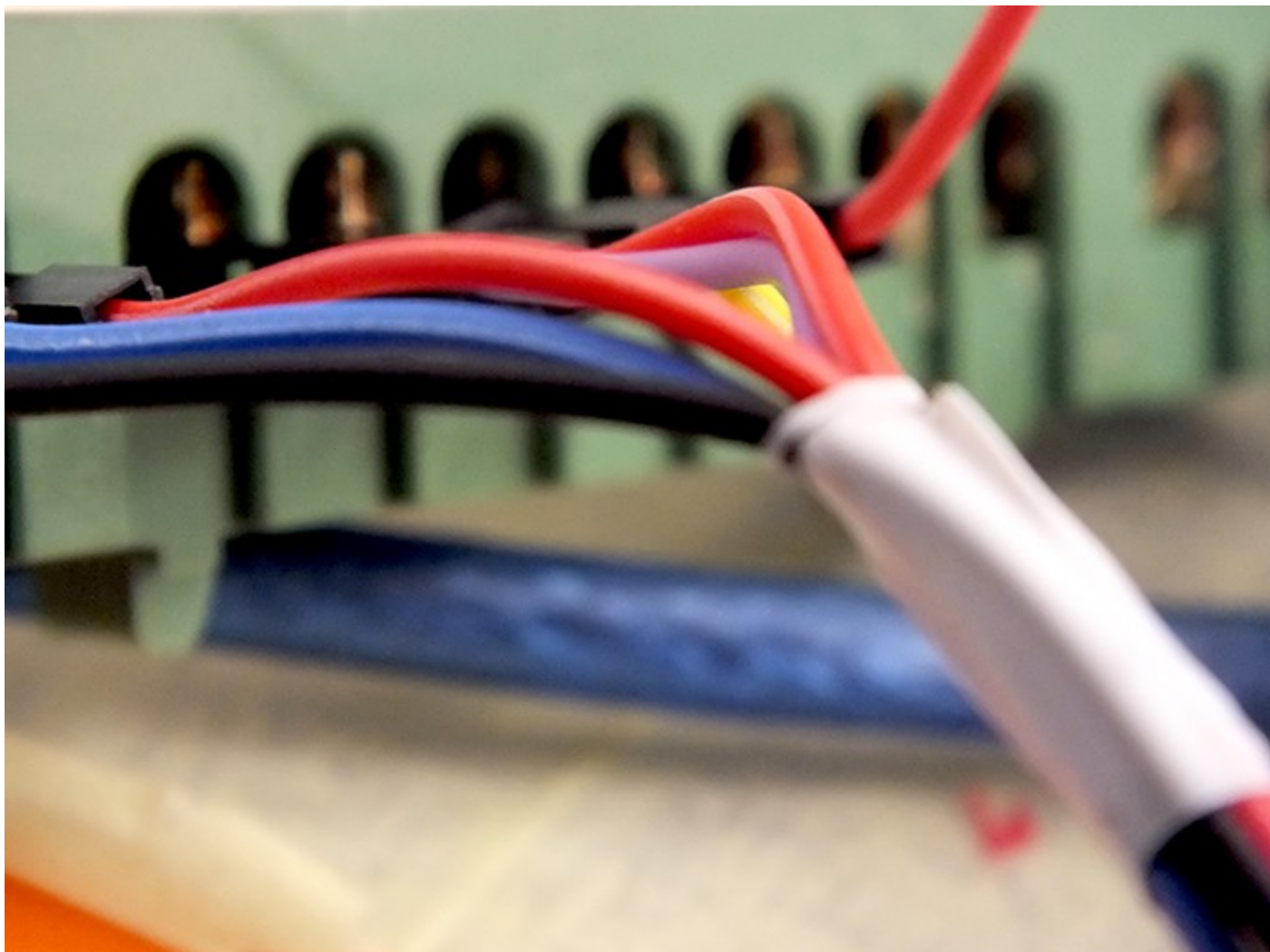
8. Подсоедините датчик Огня к Электронной плате воткнув сигнальный провод в разъем 10D



9. Мы подсоединили сигнальные провода к нашей плате. Теперь нам нужно подать питание на датчики. Отделим все провода GND и +. У нас получатся два пучка проводов.



Заведем провода GND на общую шину. В качестве шины можно использовать: монтажную колодку, планку от металлического конструктора, скрутку, пайку. Шину подсоединим проводом к разъему GND на электронной плате. Аналогично поступаем с проводами +. Подсоединяем провод + к разъему 5V на электронной плате.



Проверочная схема собрана.

Теперь перейдем к проверке работоспособности наших датчиков.

Для этого :

1. Запустите Ваш компьютер и войдите в Интернет
2. Загрузите Arduino IDE по ссылке <https://www.arduino.cc/en/Main/Software>
3. Установите программную среду, следуя указаниям инсталлятора. Во время установки подтверждаем установку драйверов.
4. Скачайте и установите библиотеки для датчиков . Для правильной установки следуйте рекомендациям на сайте.
5. Скачиваем программу Putty – это программа для чтения данных с портов. Плюс её в том, что она поддерживает вывод пояснений на русском языке.
6. Запускаем программу.
7. Подключите плату управления к USB - разъему компьютера.



## Картинка

8. В меню Инструменты программы выбираем Плата. Наша плата Arduino UNO Там же выбираем COM port.
9. Для проверки работоспособности напишем маленькие скетчи (программы) для каждого датчика.

### Датчик Протечек (Уровня воды)

```
int waterPin = 0;

void setup()

{

  Serial.begin(9600);

}

void loop()

{

  waterPin = analogRead(A1);

  Serial.println(waterPin);

  Serial.print("Наличие воды  ");

  delay(1000);

}
```

Нажимаем галочку в Arduino IDE и проверяем нашу программу. Если все сделали правильно, нажимаем стрелочку рядом и записываем нашу программу в память Электронной платы. Запускаем Putty. Выбираем Serial. Вводим номер нашего COM – порта. Нажимаем Open. Мы должны увидеть данные с датчика. 0, если датчик сухой и число больше нуля , если датчик намочить. Закроем Putty.

### Датчик освещенности

```
#define LDR_PIN A0

int lightstate = 0;

void setup()

{

  Serial.begin (9600);

}

void loop()

{

  lightstate = analogRead(A0);

  Serial.println("Освещенность ");

  Serial.print(lightstate);

  delay(10);

}
```

```
{  
  
}  
  
}
```

Проверяем, как предыдущий датчик

#### Датчик Газа

```
int sensorPin = 3;  
  
int gas = 0;  
  
void setup() {  
  
  Serial.begin(9600);  
  
}  
  
void loop() {  
  
  gas = analogRead(sensorPin);  
  
  Serial.println(gas);  
  
  delay(1000);  
  
}
```

Проверяем, как предыдущий датчик

#### Барометр

```
#include <Wire.h>  
  
#include <troyka-imu.h>  
  
Barometer barometer;  
  
void setup()  
  
{  
  
  Serial.begin(9600);  
  
  Serial.println("Begin init...");  
  
  barometer.begin();  
  
  Serial.println("Init completed");  
  
}  
  
void loop()  
  
{  
  
  float pressure = barometer.readPressureMillibars();
```



```

float altitude = barometer.pressureToAltitudeMeters(pressure);

float temperature = barometer.readTemperatureC();

Serial.print("p: ");

Serial.print(pressure);

Serial.print(" mbar \t");

Serial.print("h: ");

Serial.print(altitude);

Serial.print(" m \t");

Serial.print("t: ");

Serial.print(temperature);

Serial.println(" C");

delay(1000);

}

```

Проверяем, как предыдущий датчик

#### Датчик расстояния (Сонар)

```

int echoPin = 11;

int trigPin = 12;

void setup() {

  Serial.begin (9600);

  pinMode(trigPin, OUTPUT);

  pinMode(echoPin, INPUT);

}

void loop() {

  int duration, cm;

  digitalWrite(trigPin, LOW);

  delayMicroseconds(2);

  digitalWrite(trigPin, HIGH);

  delayMicroseconds(10);

  digitalWrite(trigPin, LOW);

  duration = pulseIn(echoPin, HIGH);

  cm = duration / 58;

```

```
Serial.print(cm);
```

```
Serial.println(" cm");
```

```
delay(100);
```

```
}
```

Проверяем, как предыдущий датчик. Для проверки работоспособности помещаем перед датчиком руку или твердый предмет. Показания датчика должны изменяться.

#### Датчик огня

```
int flamePin = 10;
```

```
int LED_Pin = 13;
```

```
int waterPin = 0;
```

```
void setup() {
```

```
    pinMode( flamePin, INPUT );
```

```
    pinMode( LED_Pin, OUTPUT );
```

```
}
```

```
void loop()
```

```
{
```

```
{
```

```
    Serial.begin(9600);
```

```
}
```

```
    if ( !digitalRead( flamePin ) )
```

```
        digitalWrite( LED_Pin, LOW );
```

```
    else
```

```
    {
```

```
        digitalWrite( LED_Pin, HIGH );
```

```
        Serial.println ( "ОБНАРУЖЕНО ВОЗГОРАНИЕ !!!!");
```

```
        delay(1000);
```

```
    }
```

```
}
```





## ВНИМАНИЕ !

Опыты с открытым пламенем можно проводить только в присутствии взрослых и в помещении , обеспечивающем своевременную ликвидацию возможного очага возгорания .

10. Подсоединим датчик движения по аналогичной схеме

11. Подсоединяем реле. 1,2,3,4 контакты реле соединяем с 4,5,6,7 входами на Электронной плате.

Подсоединяем провода питания 5V от реле к Электронной плате для проверки. Наше реле рассчитано для работы 12V, поэтому данное подключение делаем временно. Только для проверки.

12. Составляем программу , в которой объединяем все наши датчики и добавляем к ним функции управления исполнительными устройствами. Ваша программа будет иметь примерно такой вид.

```
#include <Ultrasonic.h>
```

```
#include <Wire.h>
```

```
#include <troyka-imu.h>
```

```
Barometer barometer;
```

```
#define LDR_PIN 0
```

```
int lightstate = A0;
```

```
int echoPin = 11;
```

```
int trigPin = 12;
```

```
int flamePin = 10;
```

```
int LED_Pin = 13;
```

```
const int WaterPin = A1;
```

```
int waterPin = 0;
```

```
// const int LDR_PIN = A0;
```

```
int sensorPin = 3; // Пин, на котором у нас висит сенсор газа
```

```
int gas = 0; // Переменная для хранения значения газового сенсора
```

```
int ProjectorPIN = 7;
```

```
int ObogrevPIN = 6;
```

```
int AlarmPIN = 5;
```

```
int PompaPIN = 4;
```

```
void setup() {
```

```
  Serial.begin(9600);
```

```
  pinMode(flamePin, INPUT );
```

```
  pinMode(LED_Pin, OUTPUT );
```

```
  pinMode(trigPin, OUTPUT);
```

```
  pinMode(echoPin, INPUT);
```

```
  pinMode(ProjectorPIN, OUTPUT );
```

```
  pinMode(ObogrevPIN, OUTPUT );
```

```
  pinMode(AlarmPIN, OUTPUT);
```

```
  pinMode(PompaPIN, OUTPUT);
```

```
  barometer.begin();
```

```
}
```

```
void loop()
```

```
{
```

```
  // ДАТЧИК ОГНЯ-----
```

```
  if( !digitalRead( flamePin ) )
```

```
  {
```

```
    digitalWrite(LED_Pin, LOW );
```

```
    digitalWrite(AlarmPIN, LOW );
```

```
  }
```

```
  else
```

```
  {
```

```
    digitalWrite(LED_Pin, HIGH );
```

```
    digitalWrite(AlarmPIN, HIGH );
```

```
    Serial.println ("ОБНАРУЖЕНО ВОЗГОРАНИЕ !");
```

```
    delay(0);
```

```
  }
```

```
}
```

```
// ДАТЧИК ПРОТЕЧЕК-----
```

```
int water = analogRead(WaterPin);
```

```
if (water < 100 )
```

```
{
```

```
digitalWrite(LED_Pin, LOW );
```

```
digitalWrite(AlarmPIN, LOW );
```

```
digitalWrite(PompaPIN, LOW );
```

```
}
```

```
else
```

```
{
```

```
digitalWrite( LED_Pin, HIGH );
```

```
digitalWrite(AlarmPIN, HIGH );
```

```
digitalWrite(PompaPIN, HIGH );
```

```
Serial.println ("ОБНАРУЖЕНО ЗАТОПЛЕНИЕ!");
```

```
delay(0);
```

```
}
```

```
// ДАТЧИК РАССТОЯНИЯ (СОНАР)-----
```

```
int duration, cm;
```

```
digitalWrite(trigPin, LOW);
```

```
delayMicroseconds(2);
```

```
digitalWrite(trigPin, HIGH);
```

```
delayMicroseconds(10);
```

```
digitalWrite(trigPin, LOW);
```

```
duration = pulseIn(echoPin, HIGH);
```

```
cm = duration / 58;
```

```
if (cm > 15)
```

```
{
```



```
digitalWrite(LED_Pin, LOW );  
  
digitalWrite(AlarmPIN, LOW );  
  
digitalWrite(ProjectorPIN, LOW );  
  
}  
  
else  
{  
  
    digitalWrite(LED_Pin, HIGH );  
  
    digitalWrite(AlarmPIN, HIGH );  
  
    digitalWrite(ProjectorPIN, HIGH );  
  
    Serial.println ("ОБНАРУЖЕНО ПРЕПЯТСТВИЕ !");  
  
}
```

```
// БАРОМЕТР-----
```

```
  
// создаём переменную и присваиваем ей значения абсолютного давления  
float pressure = barometer.readPressureMillibars();  
  
// создаём переменную и присваиваем ей значения высоты над уровнем море  
float altitude = barometer.pressureToAltitudeMeters(pressure);  
  
// создаём переменную и присваиваем ей температуру окружающей среды  
float temperature = barometer.readTemperatureC();  
  
  
if (temperature <= 20 )  
{  
  
    digitalWrite(LED_Pin, HIGH );  
  
    digitalWrite(ObogrevPIN, HIGH );  
  
    Serial.println ("ВКЛЮЧЕН ОБОГРЕВ");  
  
}  
  
  
else if (temperature > 45 )  
{  
  
    digitalWrite(LED_Pin, HIGH );
```

```
digitalWrite(AlarmPIN, HIGH );

Serial.println ("ОПАСНОЕ УВЕЛИЧЕНИЕ ТЕМПЕРАТУРЫ! ВОЗМОЖНО ВОЗГАРАНИЕ !!!");

}

else

{

digitalWrite(LED_Pin, LOW );

digitalWrite(AlarmPIN, LOW );

Serial.println ("ОБОГРЕВ ВЫКЛЮЧЕН");

}

}

}

// Вывод данных в Serial порт

Serial.print("Текущее давление: ");

Serial.print(pressure);

Serial.print(" mbar \t");

Serial.print("Высота над уровнем моря: ");

Serial.print(altitude);

Serial.print(" m \t");

Serial.print("Температура: ");

Serial.print(temperature);

Serial.println(" C");

delay(10);

// ДАТЧИК ОСВЕЩЕННОСТИ-----

lightstate = analogRead(A0);

if (lightstate < 500)

{

digitalWrite(LED_Pin, LOW );

digitalWrite(ProjectorPIN, LOW );

Serial.println ("ВЫКЛЮЧЕНО ОСВЕЩЕНИЕ");

}
```

```

else
{
    digitalWrite(LED_Pin, HIGH );
    delay(0);
    digitalWrite(ProjectorPIN, HIGH );
    delay(0);
    Serial.println ("ВКЛЮЧЕНО ОСВЕЩЕНИЕ");
}

// ДАТЧИК ДЫМА-----
{
    gas = analogRead(sensorPin); // Получаем значения из датчика

    Serial.println(gas); // Пишем в серийный порт

    delay(10);
}
}

```

Теперь перейдем к подключению Элементов

12. Соберите из двух батарейных блоков простую схему. Подсоедините красный провод одного блока к черному проводу другого. Оставшиеся свободными провода нарастите для подключения реле и Элементов. Вставьте батарейки в блоки.

13. Запараллельте провода, парных светодиодных лент, что бы наши ленты загорались одновременно.

14. Подсоедините контакты блока питания к блокам светодиодной ленты , для проверки её работоспособности.

**15. Припаяйте провода к контактам на разъема питания платы Ардуино УНО — Вы будете подавать туда 12В от батарей.**

16. Теперь снимите питание 5В с управления Блока реле и подайте туда 12В.

**ВНИМАНИЕ !!! ДЛЯ РАБОТОСПОСОБНОСТИ СХЕМЫ СОЕДИНИТЕ GND 5V и 12V. В противном случае схема работать не будет. GND должен быть общим.**

17. Проверьте работоспособность всей схемы.

На этом отладочный этап можно считать законченным.



## **Переходим к оформлению Игрового поля.**

Для крепления датчиков обычно достаточно прорезать небольшие отверстия размером примерно 2x8 мм. В них хорошо вставляются наконечники проводов, одетых на контакты датчиков.

Разместите датчики так, как считаете нужным для имитации контроля максимального количества параметров. В данном конструкторе нет жесткой привязки к местам расположения датчиков. Длины имеющихся отладочных проводов Вам не хватит. Воспользуйтесь дополнительными проводами и удлините или перепаяйте нужные.

Проверьте работоспособность собранной схемы.