

Всероссийская научно-техническая
конференция студентов, магистрантов,
аспирантов и молодых ученых
«Молодая мысль»

ИЗМЕРЕНИЕ ТЕМПЕРАТУРЫ И ВЛАЖНОСТИ
ВОЗДУХА НА БАЗЕ ПЛАТЫ ARDUINO UNO

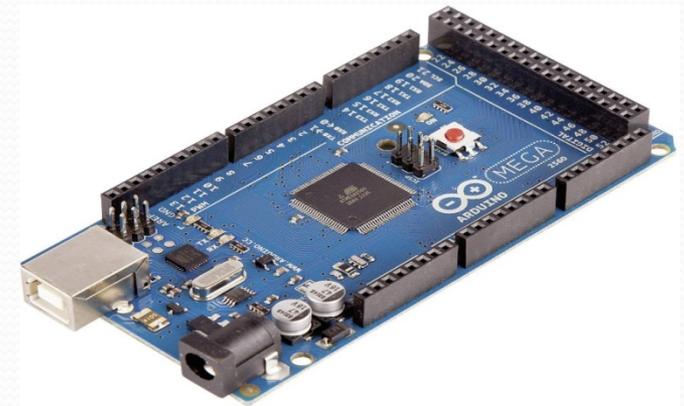
Выполнили: Мареев А.С^а., Метлина Т.А.
^аe-mail: s.nea@yandex.ru

Братск 2020

Arduino – аппаратная вычислительная платформа быстрой разработки электронных устройств.

Достоинства Arduino:

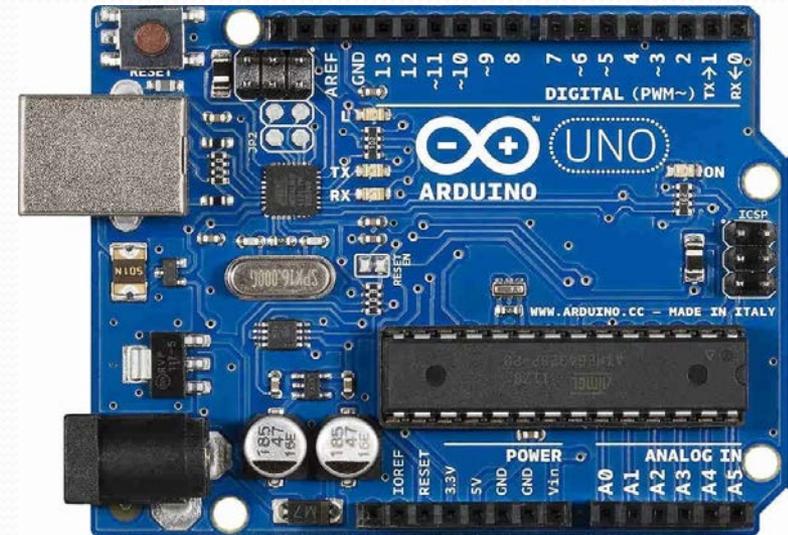
- гибкость параметров конечных устройств
- большое число различных модулей
- вариативность комбинаций модулей
- открытость архитектуры



Для постоянного отслеживания температуры и влажности воздуха в помещении на базе платы Arduino, нужно изучить необходимые компоненты, собрать их в одну схему и запрограммировать микроконтроллер.

Контроллер Arduino Uno

Контроллер Arduino Uno построен на базе микроконтроллера ATmega328. Платформа имеет 14 цифровых вход/выходов, 6 аналоговых входов, кварцевый генератор 16 МГц и разъем USB. Имеет 32 Кб флэш-памяти, из которой доступно 31.5 Кб. Для работы необходимо подключить платформу к компьютеру посредством кабеля USB, либо подать питание при помощи адаптера. Рекомендуемый диапазон питания платы в пределах 7-12В. Также плата имеет разъем на 5В для питания различных сторонних датчиков и дисплеев.



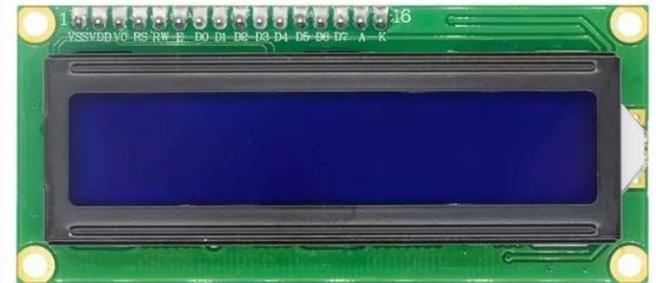
Датчик температуры и влажности DHT11

Для измерения температуры и влажности воздуха используется датчик DHT11. Он состоит из двух частей – термистора и гигрометра. Первый используется для измерения температуры, второй – влажности воздуха. Датчик содержит в себе АЦП для преобразования аналоговых значений. Измерение влажности возможно в диапазоне от 20% до 80% с погрешностью 5%. Измерение температуры происходит в интервале от 0 до 50 °С с погрешностью 2%. Питание – от 3 до 5 Вольт. Датчик смонтирован на собственной плате и уже имеет резистор на 10 кОм.



Дисплей LCD 1602

Для отображения информации используется жидкокристаллический дисплей LCD 1602 с адаптером I2C. Экран имеет 16 контактов и выполнен по технологии жидких кристаллов. Дисплей имеет формат отображения 16x2 (2 строки по 16 знаков). Каждый символ построен в виде матрицы 5x7 пикселей.



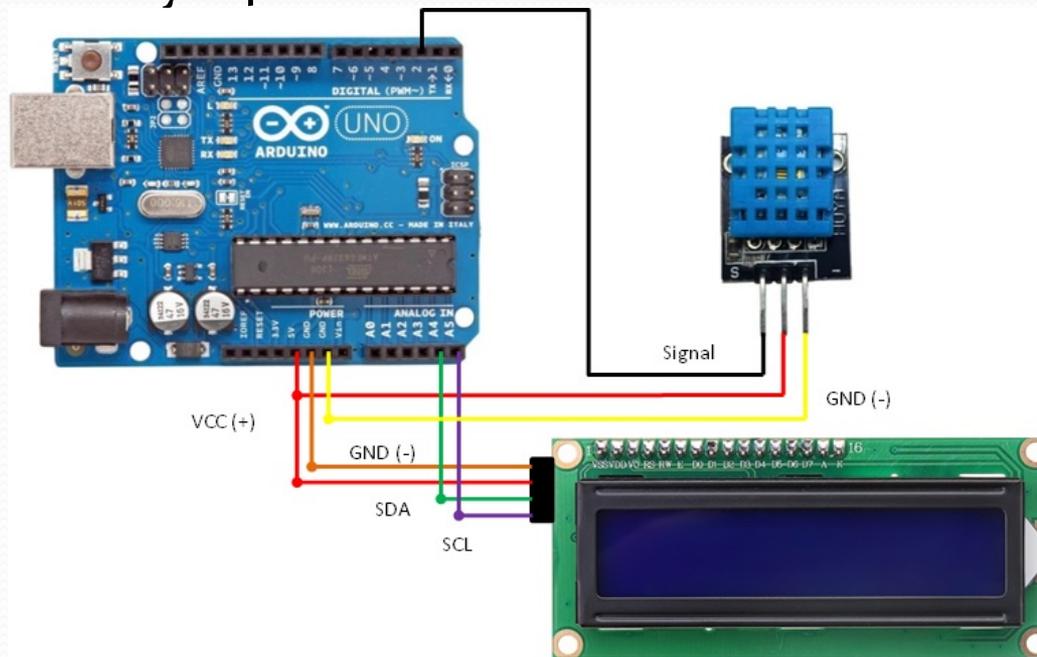
Адаптер I2C

Для подключения дисплея к плате используется I2C/IIC адаптер, который подключает стандартный экран 1602 к плате UNO при помощи 4 пинов, питание (VIN), земля (GND), линия тактирования (SCL) и линия данных (SDA). Адаптер имеет встроенный потенциометр для регулирования контрастности дисплея вручную. Дисплей подключается к адаптеру колодкой, которая соединяет идентичные пины.

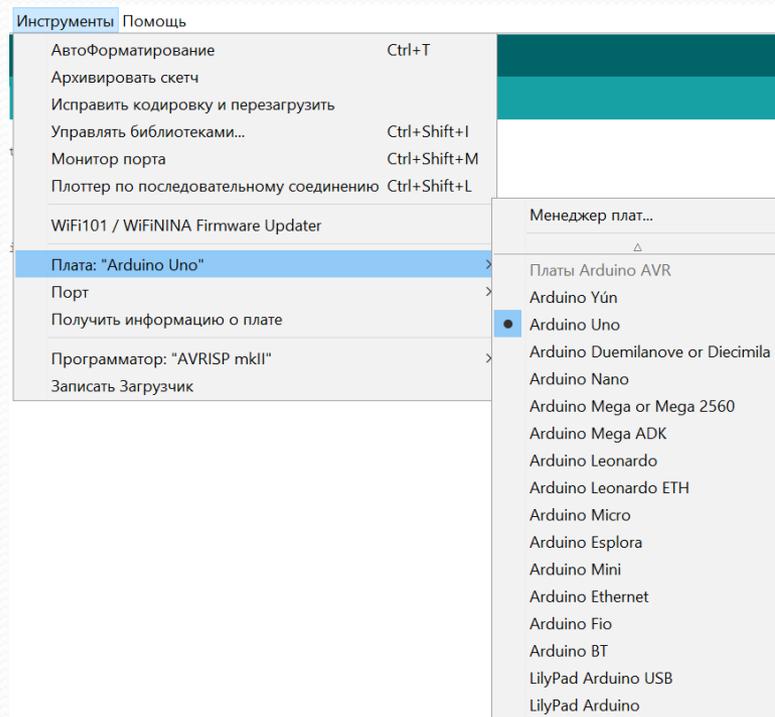
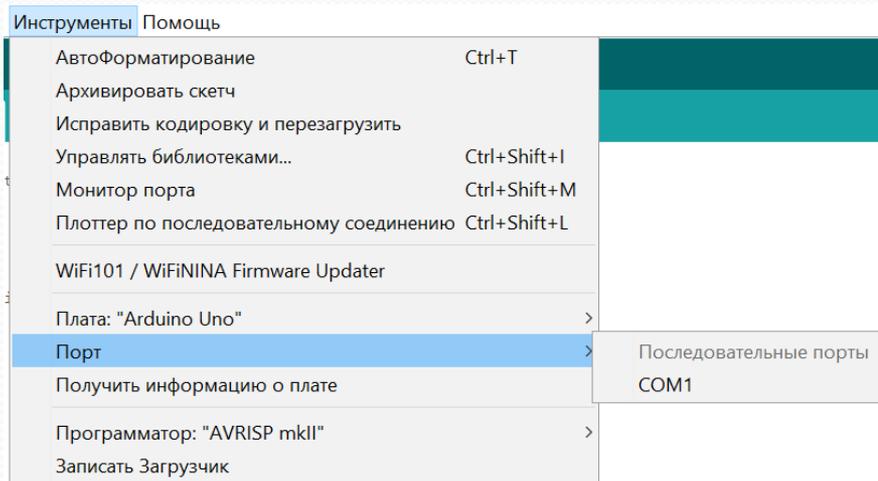


Подключение датчика и дисплея

Подключение происходит следующим образом: левый контакт датчика S (Signal) подключается к свободному пину 2, который будет использоваться в скетче (программа для работы микроконтроллера) для считывания информации с датчика, средний плюсовой контакт (Vcc) – к 5V на плате, правый минусовой контакт (Ground) – к пину GND. Жидкокристаллический монитор с поддержкой I2C подключается при помощи четырех проводов: вывод GND подключается к GND на плате, вывод VCC – на 5V, SCL подключается к пину A5, SDA подключается к пину A4.



Для работы схемы необходимо запрограммировать микроконтроллер. Для написания кода используется программное обеспечение Arduino IDE. Необходимо в настройках программы указать номер порта, через который подключен микроконтроллер к компьютеру и выбрать модель используемой платы.



Программирование микроконтроллера

```
#include <Wire.h> // Библиотека для связи микроконтроллера с дисплеем
#include <LiquidCrystal_I2C.h> // Библиотека для работы с дисплеем
LiquidCrystal_I2C lcd(0x27, 16, 2); // Адрес и размерность дисплея(16 знаков, 2 ряда)
#include <dht11.h> // Библиотека для работы с датчиком
dht11 sensor; // Тип используемого датчика
#define DHT11PIN 2 // Подключение датчика к пину 2
byte gradus[8] = { // Кодирование символа градуса
  В00111,
  В00101,
  В00111,
  В00000,
  В00000,
  В00000,
  В00000,};
void setup(){
  lcd.begin(); // Инициализация LCD дисплея
  lcd.backlight(); // Включение подсветки дисплея
  lcd.createChar(1, gradus); // Создание символа под номером 1
void loop(){
  lcd.clear(); // Очистка экрана
  lcd.setCursor(0, 0); // Установка курсора в начало 1 строки
  lcd.print("Hum: %"); // Вывод текста
  lcd.setCursor(11, 0); // Установка курсора на 11 знак 1 строки
  lcd.print(sensor.humidity); // Вывод параметра влажности с датчика
  lcd.setCursor(0, 1); // Установка курсора в начало 2 строки
  lcd.print("temp: 1С"); // Вывод текста, вместо 1 будет значок градуса
  lcd.setCursor(11, 1); // Установка курсора на 11 знак 2 строки
  lcd.print(sensor.temperature); // Вывод параметра температуры с датчика
  delay(2000); // Считывание данных каждые 2 секунды
```

Для написания кода использовались следующие директивы и команды:

Include – директива, позволяющая подключать в проект дополнительные файлы с кодом. Для правильной работы скетча используются сторонние библиотеки `dht11` для работы с датчиком и `LiquidCrystal_I2C` для работы с дисплеем, которые не входят в стандартный пакет Arduino IDE.

Библиотека `Wire` предназначена для работы микроконтроллера с дисплеем через интерфейс `I2C`.

Define – директива, дающая команду процессору заменить указанное название на указанное значение.

Byte – переменная, элементарная ячейка для хранения данных, имеющая вес 1 байт. Так как символ градуса отсутствует в памяти дисплея, его необходимо запрограммировать вручную в переменную, где 0 – пустые, а 1 – закрашенные пиксели.

`Void setup` – функция, содержимое которой выполняется один раз при запуске микроконтроллера.

`Lcd...` – адрес дисплея, на который отправляются команды.

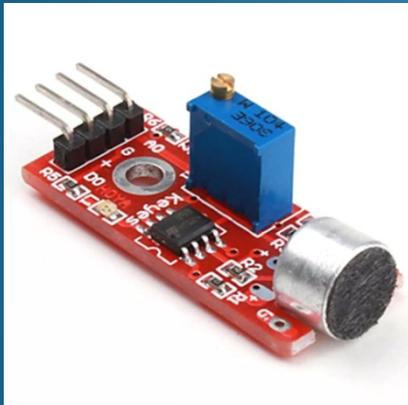
`Void loop` – функция, содержимое которой выполняется в цикле на протяжении всего времени работы микроконтроллера. `Delay` останавливает выполнение кода на 2 секунды для более точного обновления информации.

После подключения платы к компьютеру через USB и загрузки скетча в микроконтроллер плата полностью готова к работе. Необходимо убедиться, что компиляция прошла успешно и скетч загружен полностью, об этом сообщит сама программа. При подключении питания от блока на 12В на дисплее каждые 2 секунды будет обновляться информация о температуре и влажности воздуха.

Примечание: плата имеет возможность подключения большого количества каналов, а также запрограммированный на языке C++ интерфейс и возможность задания диапазона поддержки температуры средствами нагревания и охлаждения автоматически.

Итог

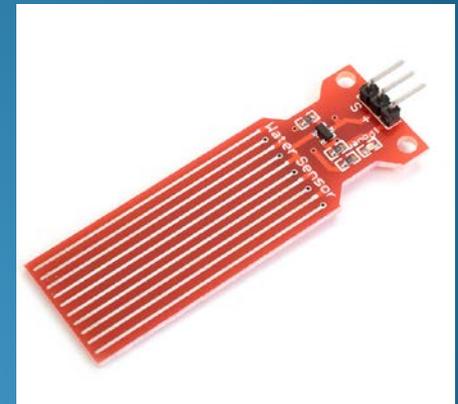
Таким образом, используя достаточно простые, дешевые и общедоступные компоненты, можно организовать систему сбора данных, точность измерений которой будет определяться точностью датчиков, подключаемых к микроконтроллеру.



Датчик звука



Датчик удара
и вибрации



Датчик уровня
воды