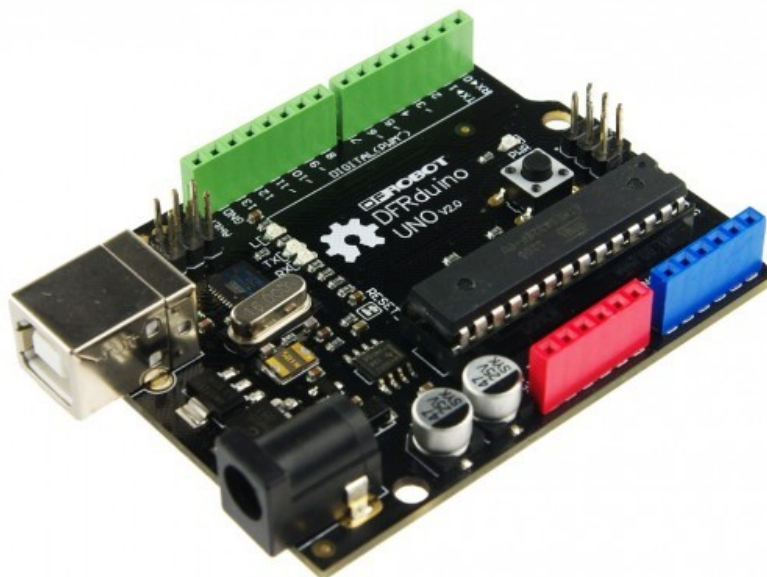


DFROBOT

DFRduino UNO V2.0



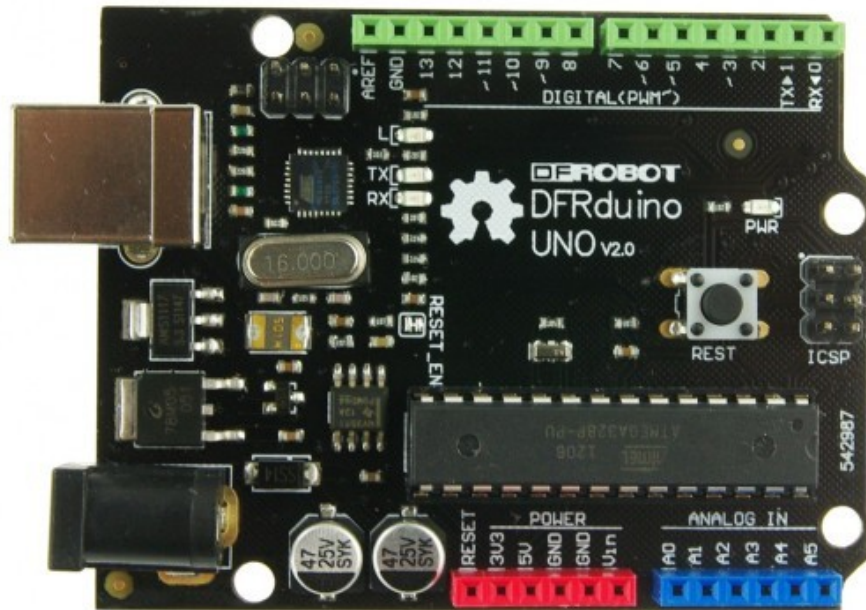
Введение

Плата DFRduino Uno V2.0 от DFRobot является полностью совместимой с Arduino UNO R2. На ней расположена микросхема ATmega8U2, запрограммированная как USB-to-serial конвертер. Так же удобно, что при изготовлении DFRduino используются DIP (штырьковые) корпуса. Это означает, что вы можете, например, заменить микроконтроллер на плате или вынуть его, что бы использовать в каком-то проекте, требующим более компактные решения.

Разъемы на плате DFRduino раскрашены разными цветами, для того, что бы было удобно различать разные типы портов: красный – секция питания, синий – аналоговые входы/выходы, зеленый – цифровые входы/выходы. Эти цвета используются в сенсорных кабелях от DFRobot, поэтому, вам будет легко разобраться в их подключении.

Arduino является полностью открытой платформой, состоящей из платы и среды разработки, в которой реализована переработанная версия языка Processing/Wiring. Возможно использование для разработки автономных интерактивных объектов, либо взаимодействовать с вашим компьютером (с помощью Flash, Processing, MaxMSP).

Arduino Uno - плата с микроконтроллером ATmega328, который имеет 14 цифровых входов/выходов (6 из которых можно использовать как ШИМ - выходы), 6 аналоговых входов, кварцевый резонатор с частотой 16МГц, разъем USB, разъем питания, ICSP разъем, кнопку сброса. Содержится все необходимое для эксплуатации микроконтроллера. Просто подключите плату к компьютеру с помощью USB кабеля, либо запитайте через AC-DC адаптер (блок питания) или от батарей.



В настоящее время Arduino IDE доступно для Windows, Mac OS X и Linux 32bit. Среду можно скачать по ссылке: <http://arduino.cc/en/Main/Software>

Данная программа распространяется в виде архива и не нуждается в установке.

Перед началом программирования необходимо выбрать номер последовательного порта (виртуальный COM – порт) и тип платы. Сделать это можно во вкладке **Service**.

В меню **File -> Examples** находятся базовые примеры работы с портами Arduino и его библиотеками.

Структура программы

В любой, даже самой простой программе для Arduino должны присутствовать две основные функции, без которых она не будет корректно компилироваться.

Функция `setup()` необходима для настроек устройства во время запуска. С нее начинается выполнение программы, поэтому она очень удобна для хранения в ней первоначальных настроек микроконтроллера: прерывания, порты, которые будут в режиме выходов и т.д. По умолчанию, все порты находятся в режиме входов, что снижает вероятность случайного короткого замыкания.

Функция **`loop()`** является основой программы. Ее содержимое будет постоянно повторяться, пока устройство не будет выключено или перезагружено.

Информация о языке программирования может быть найдена тут:

<http://arduino.cc/en/Reference/HomePage> (на английском языке)

<http://arduino.ru/Reference>

Примеры

Начнем с самого простого примера, с которого обычно начинают обучение программированию плат Arduino.

*/**

Пример1. Blink

Включает светодиод на секунду, затем выключает его на секунду.

**/*

// На большинстве плат Arduino к пину 13 подсоединен светодиод.

//Для удобства, дадим ему название:

`int led = 13;`

//функция setup() запускается один раз:

`void setup()`

`{`

//инициализация цифрового порта 13 как выхода.

`pinMode(led, OUTPUT);`

`}`

//тело функции loop() повторяется в бесконечном цикле:

`void loop()`

`{`

`digitalWrite(led, HIGH); // включить светодиод (HIGH - уровень напряжения)`

```
delay(1000);           // подождать секунду
digitalWrite(led, LOW); // выключить светодиод
delay(1000);           // подождать секунду
}
```

/*

Пример2. ReadAnalogVoltage

Считывает значение на ножке D0, преобразует его в напряжение и выводит результат в монитор последовательного порта. Присоедините среднюю ножку потенциометра к порту A0, а боковые ножки к +5 и общему проводу.

*/

```
void setup()
{
    // инициализация последовательного порта микроконтроллера на скорости 9600 бит в
    //секунду:
    Serial.begin(9600);
}

//тело функции loop() повторяется в бесконечном цикле:
void loop()
{
    // считать значение аналогового входа 0:
    int sensorValue = analogRead(A0);
    // Преобразовать значение АЦП (которое лежит в диапазоне 0 – 1023) в напряжение (0 //-
    5V):
    float voltage = sensorValue * (5.0 / 1023.0);
    // вывести считанное значение:
    Serial.println(voltage);
}
```

Характеристики платы

- Микроконтроллер ATmega328 (DIP корпус)

- Рабочее напряжение 5В
- Входное напряжение (рекомендуется) 7-12В
- Входное напряжение (Предел) 6-20В
- Цифровые входы/выходы 14 (из которых 6 позволяют генерировать ШИМ)
- Аналоговые входы 6
- Ток на один вход/выход 40 мА
- Ток стабилизатора напряжения на 3.3V 50 мА
- Память программ 32 Кб, из которой 2 кб заняты загрузчиком
- Тактовая частота 16 МГц
- Размеры: 75x54x15 мм
- Вес: 45 г.