



# СПбГЭТУ «ЛЭТИ» первый электротехнический



Р.Р. Фаткиева

# Интернет вещей

# Практическая работа. Получение данных с датчика подключенного к Arduino

СПбГЭТУ «ЛЭТИ», 2021 г.





## 1. АППАРАТНОЕ ОБЕСПЕЧЕНИЕ ЮТ. ПОЛУЧЕНИЕ ДАННЫХ С ДАТЧИКА ПОДКЛЮЧЕННОГО К ARDUINO

Цель работы: изучение теоретического материала о взаимодействии датчика и микроконтроллера, формирование практических навыков разработки программы для Arduino IDE, изучение ее простейшей программы.

#### 1.1 . Общие сведения

Arduino IDE - это среда разработки программ для платформы Arduino. Она позволяет взаимодействовать с Arduino как для передачи данных, так и для прошивки кода. Интерфейс программы Arduino IDE 1.6.7 представлен на Рис. 1.



Рис. 1

Интерфейс имеет простую структуру. В верхней части расположена панель навигации, ниже кнопки для сохранения и загрузки программы в Arduino, следом текстовый редактор для написания программ, зеленая





область под текстовым редактором отображает информацию о загрузке программы на Arduino, последняя часть - это консоль для вывода служебной информации. В нижнем правом углу показывается информация о порте, к которому подключена Arduino. Программы, разрабатываемые для Arduino, иногда называют «Скетчами». Скетчи пишутся в текстовом редакторе и сохраняются в файлах с расширением «.ino».

#### 1.2. Порядок выполнения работы

#### 1.2.1 Ознакомление с базовым набором программ

Среда разработки Arduino IDE содержит базовый набор программ (Скетчей), который возможно использовать как для обучения, так и применения их, как шаблонов, для разработки собственных проектов.

Для ознакомления с работой программного обеспечения необходимо загрузить программу «Blink», которая позволяет проявить работоспособность аппаратного обеспечения платы за счет демонстрации «мигать» светодиодом на плате Arduino.

Рассмотрим панель управления, представленная сервисами изображенными на Рис.2



#### Рис. 2

Первая кнопка, в виде «галочки», осуществляет проверку написанного кода, следующая (стрелочка) проверяет работоспособность загруженного программного кода, и сразу после проверки осуществляет его загрузку на микроконтроллер. Следующие три кнопки по порядку отвечают за создания нового документа, открытие существующего и сохранения текущего.

Нажать на кнопку «Открыть», выбрать пункт «01.Basics» и программу «Blink». Проверить открытие нового окна, с уже представленной на листинге 1 программой.



Листинг программы 1



```
void setup() {
    // initialize digital pin LED_BUILTIN as an output.
        pinMode(LED_BUILTIN, OUTPUT);
}
// the loop function runs over and over again forever:
void loop() {
     digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
        delay(1000); // wait for a second
digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW
        delay(800); // wait for a second
}
```

#### 1.2.2 Подключение Arduino к компьютеру

С использованием разъема USB подключить плату Arduino к компьютеру. Обратить внимание, что если всё подключили правильно и питания для платы достаточно, то должен загореться зеленый светодиода напротив надписи «On». Подключение платы Arduino изображено на Рис. 3.



Рис. 3

Определить порт подключения по надписи отображенной в левом нижнем углу программы (определение платформы и порта при подключении Arduino представлено на Рис. 4).









Загрузить программу на Arduino нажав на кнопку «Загрузка». Проверить уведомление о начале компиляции и завершении компиляции, которая отражена в строке сообщений, и окончание загрузки кода на плату Arduino (на Рис. 5).



Рис.5

Осуществить проверку работоспособности кода. Для рассмотреть поведение светодиода. Как только программа была загружена светодиод на плате начинает мигать с задержкой в 1 секунду. Это означает, что плата работает нормально и готова для создания новых проектов. Моргание светодиода на плате Arduino изображено на Рис. 6.







Рис. 6 [1]

#### 1.2.3 Подключение датчика к Arduino

Heart Rate Pulse Sensor - датчик для измерения сердечного ритма. Данный датчик используется для измерения частоты пульса (измеряется в количестве ударов в минуту). Датчик имеет три вывода:

VCC - 5 V;

GND - земля;

SIG - аналоговый выход.

Контакты питания VCC и GND Heart Rate Pulse Sensor подключить к 5V и GND Arduino UNO, a SIG подключить к аналоговому входу A3 (схема соединения продемонстрирована на Рис. 7).



Рис. 7





Перейти во вкладку CODE, где открыть окно с уже готовой программой, код которой представлен на листинге 2. Выполните код.

Листинг программы 2

```
// Include Libraries
#include "Arduino.h"
#include "pulse-sensor-arduino.h"
// Pin Definitions
#define HEARTPULSE_PIN_SIG
                                Α3
// Global variables and defines
// object initialization
PulseSensor heartpulse;
// define vars for testing menu
const int timeout = 10000;
                                 //define timeout of 10 sec
char menuOption = 0;
long time0;
void setup()
{
    // Setup Serial which is useful for debugging
    // Use the Serial Monitor to view printed messages
    Serial.begin(9600);
    while (!Serial); // wait for serial port to connect. Needed for native USB
    Serial.println("start");
    heartpulse.begin(HEARTPULSE_PIN_SIG);
    menuOption = menu();
}
void loop()
{
    if(menuOption == '1') {
    // Heart Rate Pulse Sensor - Test Code
    //Measure Heart Rate
    int heartpulseBPM = heartpulse.BPM;
    Serial.println(heartpulseBPM);
    if (heartpulse.QS == true) {
    Serial.println("PULSE");
    heartpulse.QS = false;
    }
    }
    if (millis() - time0 > timeout)
    {
        menuOption = menu();
    }
}
// Menu function for selecting the components to be tested
// Follow serial monitor for instructions
char menu()
{
    Serial.println(F("\nWhich component would you like to test?"));
    Serial.println(F("(1) Heart Rate Pulse Sensor"));
    Serial.println(F("(menu) send anything else or press on board reset button\n"));
    while (!Serial.available());
   // Read data from serial monitor if received
```





```
while (Serial.available())
{
    char c = Serial.read();
    if (isAlphaNumeric(c))
    {
        if(c == '1')
               Serial.println(F("Now Testing Heart Rate Pulse Sensor"));
        else
        {
            Serial.println(F("illegal input!"));
            return 0;
        }
        time0 = millis();
        return c;
    }
}
```

}

Откройте окно с консолью с автоматическим определением Arduino и активным портом, передающим данные считайте данные с датчика в реальном времени.





## КОНТРОЛЬНЫЕ ВОПРОСЫ

- 1. В чем необходимо обязательно убедиться перед загрузкой программы в контроллер Arduino?
- 2. Где находится встроенный светодиод? Как он обозначен? К какому цифровому выходу он подключён?
- 3. Где находятся светодиоды, которые могут служить индикаторами загрузки программы? Как они обозначены?
- 4. Какая функция отвечает за включение светодиода?
- 5. Что означает ошибка при загрузке скетча «programmer is not responding»?
- 6. В отличие микропроцессоров от микроконтроллеров?
- 7. Сколько цифровых контактов (входов / выходов) есть на платформе? Где они расположены?





## СПИСОК ЛИТЕРАТУРЫ

- 1. Муромцев Д.И., Шматков В.Н. «Интернет Вещей: Введение в программирование на arduino» СПб: Университет ИТМО, 2018. 36 с.
- 3. «Петин В. А., Биняковский А. А. П29 Практическая энциклопедия Arduino. - М.: ДМК Пресс, 2017. - 152 с https://www.rulit.me/data/programs/resources/pdf/Petin\_Praktich eskaya-enciklopediya-Arduino\_RuLit\_Me\_603742.pdf

