

Научно-исследовательский модуль «Весна»

Лаборатория ЦДТТ «ЮНОСТЬ», г.Сергиев Посад, 2021 г.

Разработчик и изготовитель, автор программного кода — Балаченков Давид Дмитриевич.

Руководитель проекта - преподаватель дополнительного образования Изотов Александр Александрович

Научно-исследовательский модуль «Весна» разработан в лаборатории творческого объединения ЦДТТ «ЮНОСТЬ».

Целью данного проекта является упрощение и ускорение разработки научных программ, выполняемых в микроконтроллере Arduino.

Задачи, которые позволяет решать модуль широки — от применения в образовательной сфере, до выполнение работ в серьёзных научных исследованиях.

Проблема, с которой сталкиваются молодые исследователи часто заключается в выборе программно-аппаратного инструментария, применяемого в экспериментах. Научный исследовательский модуль «Весна» позволяет в значительной мере решить эту проблему.

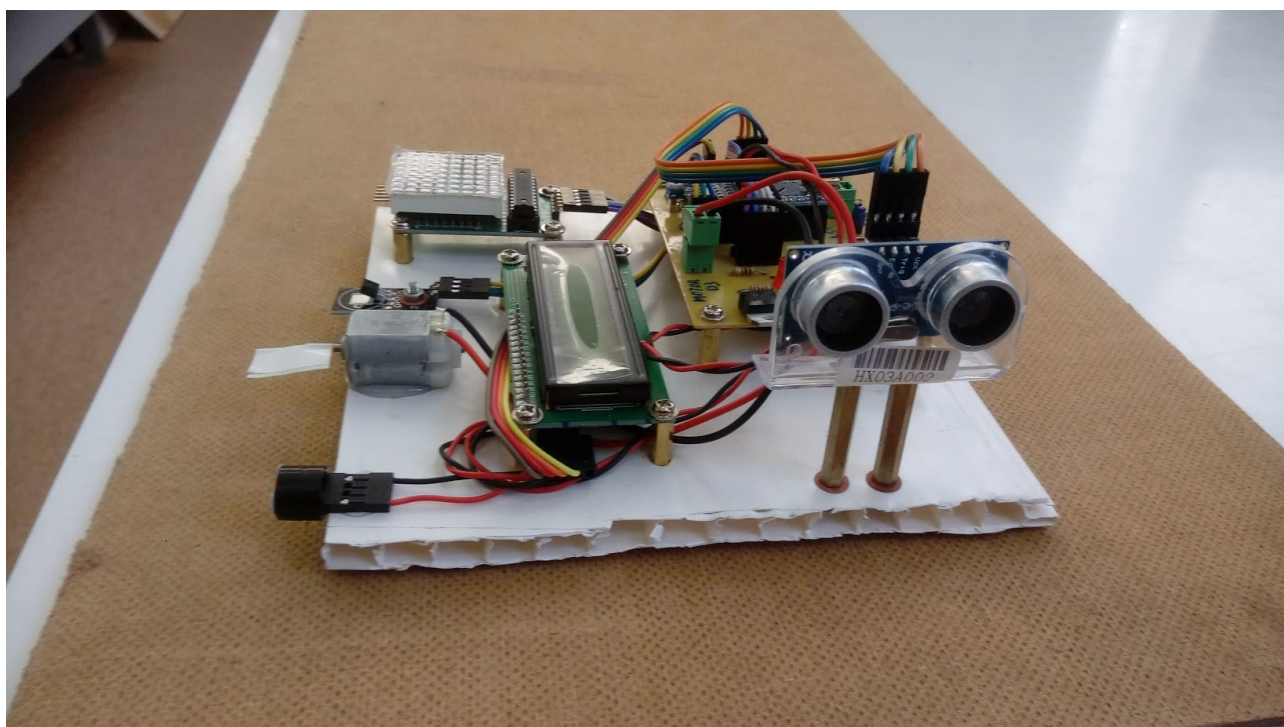
«Весна» - это не просто электронный модуль, это целый программно-аппаратный комплекс, который, несмотря на свои скромные внешние габариты обладает весьма внушительными возможностями.

Потенциальные пользователи модуля «Весна» - школьники, студенты и младшие научные сотрудники лабораторий.

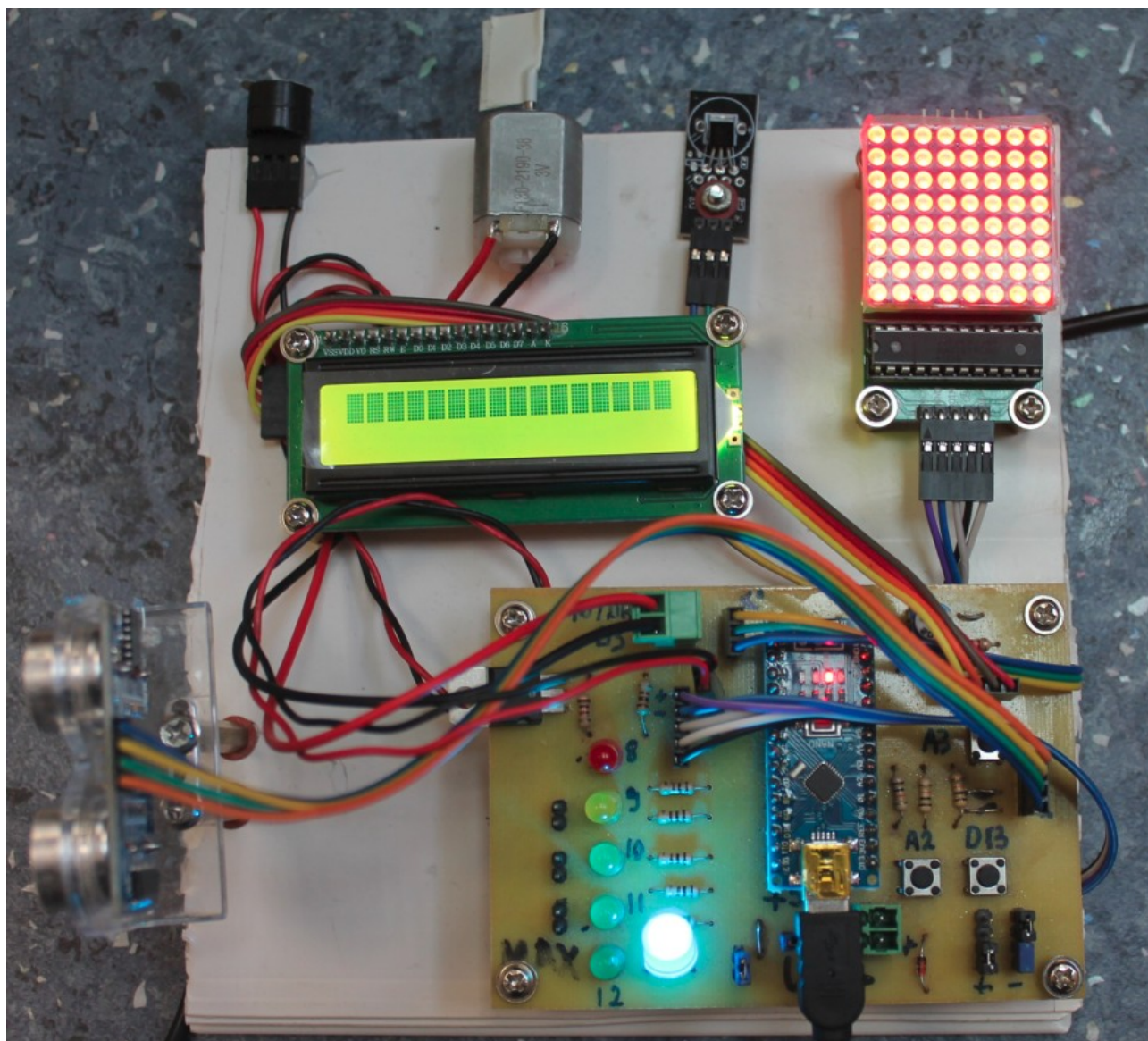
При помощи и с использованием модуля Весна молодые исследователи могут изучать климат планеты, т. к. модуль оснащён встроенным цифровым датчиком температуры. «Весна» также способна пробурить почву встроенным

электромотором. Для взаимодействия с людьми у «Весны» есть несколько возможностей. Во-первых радиосвязь, которая подключается к модулю дискретно. Во-вторых в модуле есть встроенные сигнально-световые устройства, которые позволяют видеть это устройство издалека или с беспилотника. Кроме этого, модуль оснащён встроенным пьезовым динамиком, для сигнализирования о состоянии устройства.

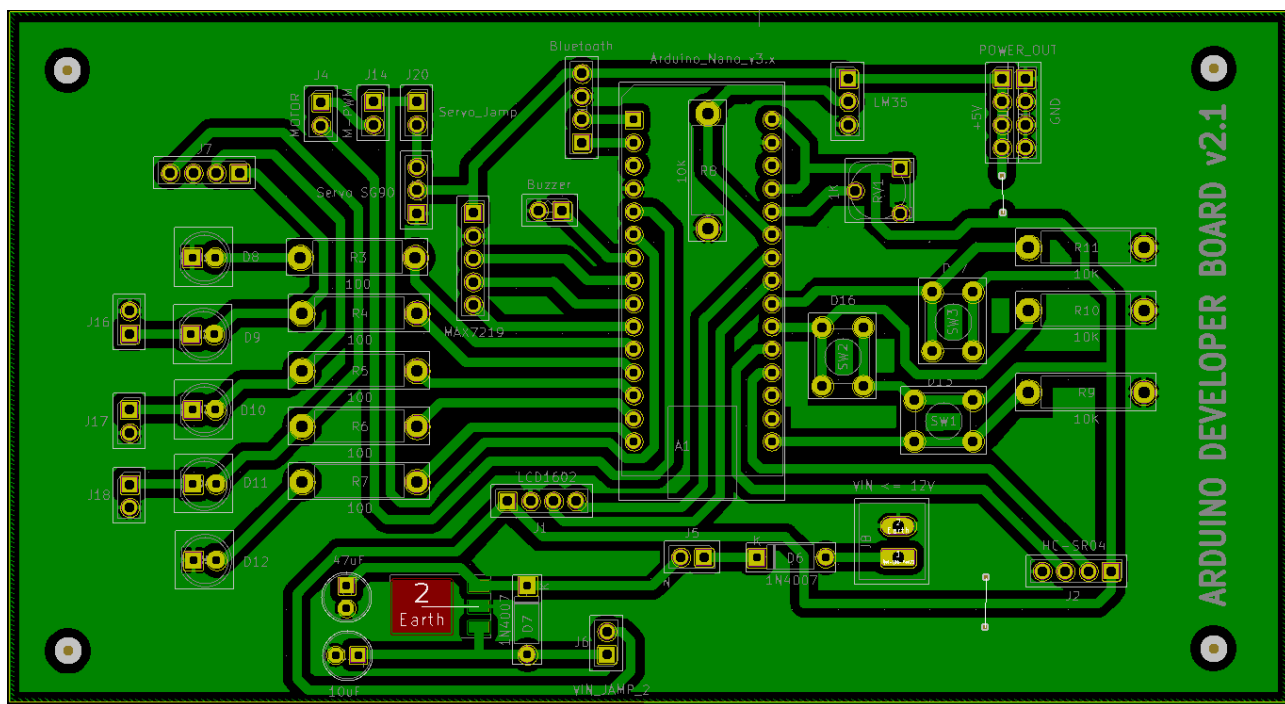
Научно-исследовательский модуль «Весна» спроектирован юными специалистами-робототехниками с использованием свободного программного обеспечения, такими как KiCAD EDA (применяется в Европе, в ЦЕРН при работе с большим адронным коллайдером), Fritzing, Arduino IDE, FlatCam. Основной операционной системой, в которой проходит вся работа над модулем «Весна» является ОС Ubuntu Linux.



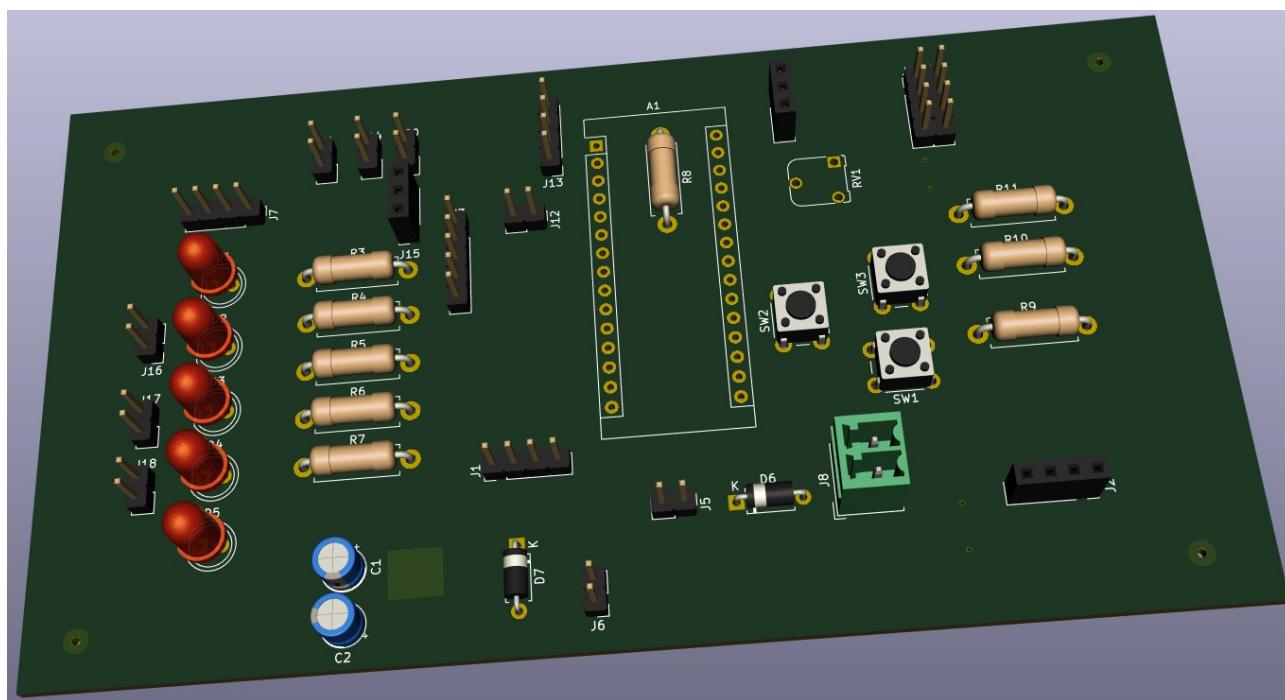
Общий вид модуля «Весна»



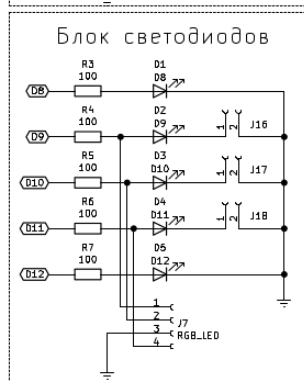
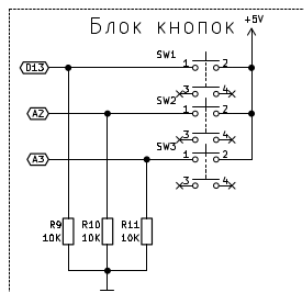
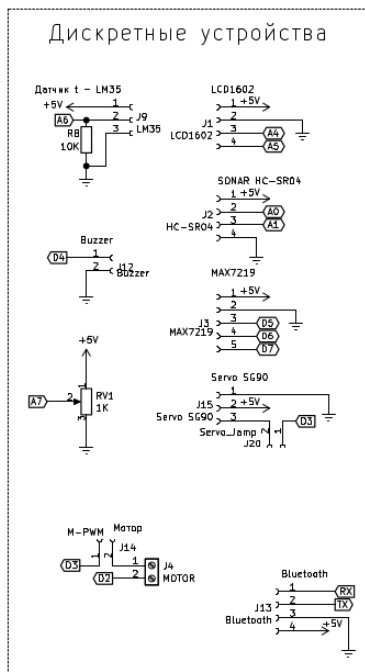
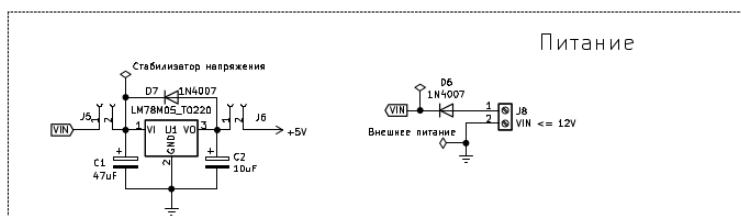
Модуль «Весна» в рабочем состоянии.



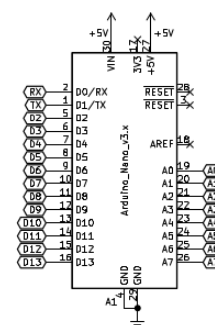
Эскиз печатной платы научно-исследовательского программно-аппаратного модуля «Весна».



3D-модель модуля.



Arduino Dev Board 2.1 модуль "Весна"



Sheet: /
File: ArduinoDevBoard_2_1.sch

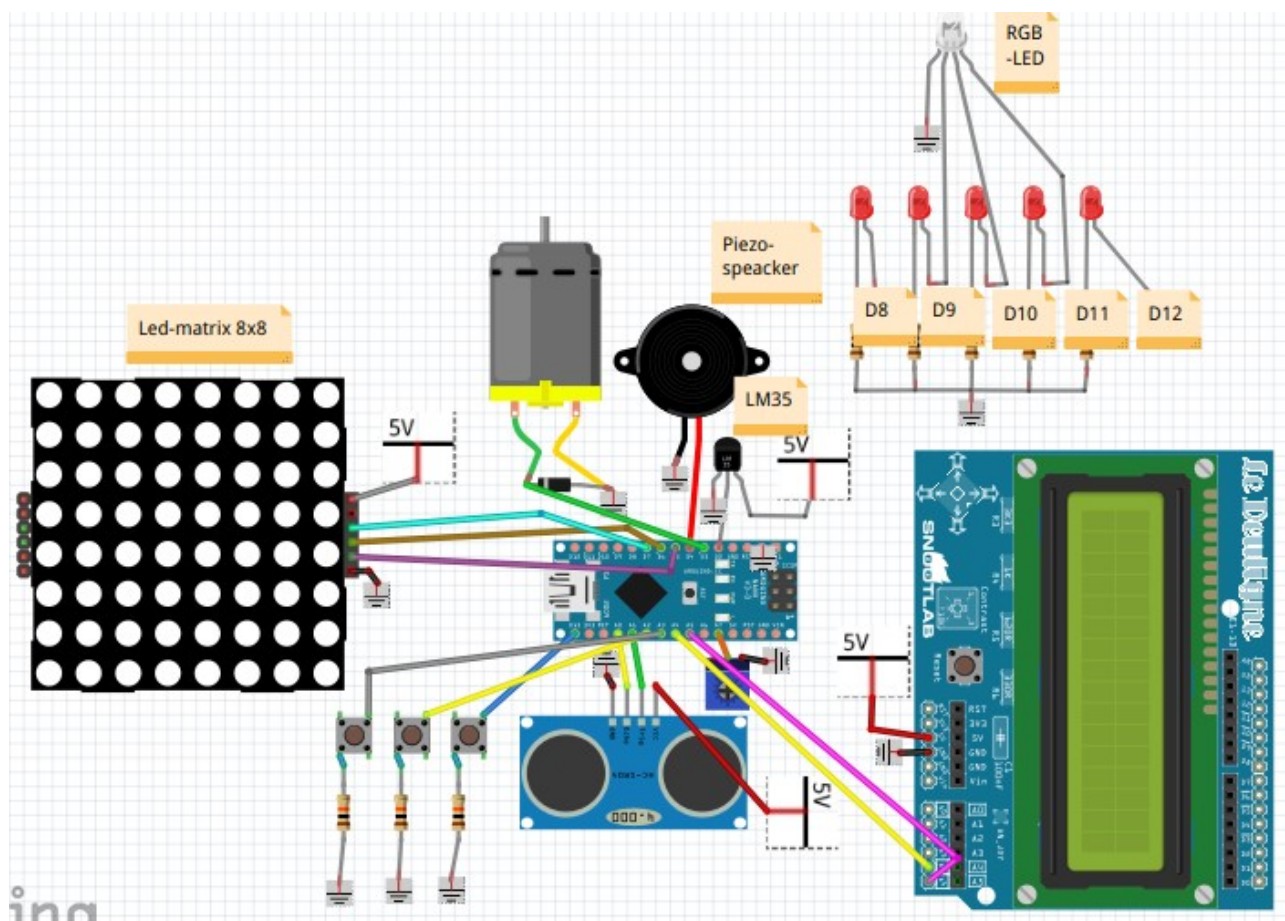
Title:

Size: A4

Date:

Rev:

Принципиальная схема модуля «Весна»



Эскиз модуля

Программный код для тестирования всех компонентов модуля «Весна»:

```
// =====  
// All tests for Arduino Dev Boards V.2  
// V 1.0  
// =====  
  
#include <Arduino.h>  
#include <OneWire.h> // Для работы датчика t dc18b20  
#include <DallasTemperature.h>  
#include <NewPing.h> // Для работы Сонара HC-SR04  
#include "LedControl.h" // Для работы 8x8 светодиодной матрицы  
#include <Wire.h>  
#include <LiquidCrystal_I2C.h> // Библиотека I2C дисплея  
  
// pin 2 для датчика t на Arduino  
const int ONE_WIRE_BUS = 2;  
  
// Шина I2C для работы датчика t  
OneWire oneWire(ONE_WIRE_BUS);  
  
// Передача с шины на датчик t  
DallasTemperature sensors(&oneWire);  
  
LiquidCrystal_I2C lcd(0x27, 16, 2); // Устанавливаем дисплей  
  
/*  
Теперь настраиваем объект LedControl.  
Эти номера контактов, возможно, не будут работать в вашем проекте.  
Если необходимо, поменяйте их.
```

5-ый контакт подключен к DIN.

6-ый контакт подключен к CLK.

7-ый контакт подключен к CS.

У нас есть только один MAX72XX.

*/

```
LedControl lc = LedControl(5, 6, 7, 1); // Объект экрана 1602
```

```
// Пины для подключения Сонара HC-SR04
```

```
const int PIN_ECHO = 14;
```

```
const int PIN_TRIG = 15;
```

```
NewPing sonar(PIN_TRIG, PIN_ECHO, 400); // Объект Сонара HC-SR04
```

```
// Функция Сонара
```

```
// Функция 8x8 светодиодной матрицы
```

```
// Функция 1602 экрана
```

```
// Функция динамика
```

```
// Функция датчика t
```

```
// Функция группы светодиодов
```

```
// Функция RGB-светодиода
```

```
// Функция потенциометра
```

```
// Функция мотора
```

```
// Функция кнопок
```

```
void setup() {
```

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

```
  // Датчик t
```

```
  Serial.println("Dallas Temperature IC Control Library Demo");
```

```
  // Инициализация библиотеки температурного датчика
```



```

sensors.begin();

// -----
// Экран 1602
lcd.init(); // Инициализация LCD
//lcd.backlight();// Включаем подсветку дисплея
lcd.begin(16, 2); // Задаем размерность экрана
//-----
// Светодиодная матрица 8x8
lc.shutdown(0, false);// Выключить энергосбережение, включить матрицу
lc.setIntensity(0, 1);// Установить яркость (0 ~ 15 возможных значений)
lc.clearDisplay(0);// Очистить матрицу
//-----
// Группа встроенных светодиодов
pinMode(4, OUTPUT);
for (int i = 8; i <= 12; ++i) {
    pinMode(i, OUTPUT);
}
//-----
// Сонар HC-SR04
pinMode(PIN_TRIG, OUTPUT);
//-----
}

void loop() {
    // -----
    // Проверка мотора
    /*
    int motorPWM = 3;
    analogWrite (motorPWM, 85);

```

```

*/
// -----
// Датчик температуры
// вызов sensor.requestTemperatures (), чтобы отправить запрос глобальной
// температуры всем устройствам на шине
Serial.print(" Requesting temperatures...");
sensors.requestTemperatures(); // Отправляем команду для получения t
Serial.println("DONE");
Serial.print("Temperature for Device 1 is: ");
Serial.print(sensors.getTempCByIndex(0)); // Why "byIndex"?
// -----
// Проверка экрана 1602
lcd.clear(); // Очищаем экран перед получением нового значения
lcd.setCursor(0, 0); // курсор на 4-й символ 1-й строки
lcd.print("Temperature:"); // Тест на 1-й строке экрана
lcd.setCursor(2, 1); // курсор на 7-й символ 2-й строки
lcd.print(sensors.getTempCByIndex(0)); // Значение t на 2-й строке экрана

// -----
// Проверка 8x8 светодиодной матрицы
//lc.setLed(0, 3, 4, true);
/*
lc.setRow(0, 0, B11111111);
lc.setRow(0, 1, B11111111);
lc.setRow(0, 2, B11100111);
lc.setRow(0, 3, B11011011);
lc.setRow(0, 4, B11100111);
lc.setRow(0, 5, B11111111);
lc.setRow(0, 6, B11111111);

```

```

lc.setRow(0, 7, B11111111);
*/
// Проверка сонара
/*
Serial.print("Sonar: ");
Serial.println(sonar.ping_cm());
*/
// Проверка кнопок
/*
bool button1 = digitalRead(13);
bool button2 = digitalRead(16);
bool button3 = digitalRead(17);
Serial.print("Button 1: ");
Serial.print(button1);
Serial.print(" ");
Serial.print("Button 2: ");
Serial.print(button2);
Serial.print(" ");
Serial.print("Button 3: ");
Serial.println(button3);
delay(20);
*/
// Проверка потенциометра
/*
int pot = A7;
int val = analogRead(pot);
Serial.println(val);
*/
// -----

```

```

// Проверка группы светодиодов
/*
for (int i = 8; i <= 12; ++i) {
    digitalWrite(i, !digitalRead(i));
    delay(150);
}
*/
// -----
// Проверка динамика
/*
digitalWrite(4, HIGH);
delay(500);
digitalWrite(4, LOW);
delay(500);
*/
}
// ----- //
// END FILE
// ----- //

```

Список используемой литературы:

1. Электроника шаг за шагом. Издание 4 дополненное. Сворень Р.А. Москва 2001;
2. Электроника для начинающих. Издание 2. Платт Чарлз. БВХ-Петербург 2017;
3. Электроника для начинающих. Аливерти Паоло. Бомбора, Москва 2019;
4. Изучаем Arduino. Блум Джереми. БВХ-Петербург 2016;
5. Электроника. Мобильные роботы на базе Arduino. 2-е издание. Момот Михаил. БВХ-Петербург 2018.