**ЛАБОРАТОРНАЯ РАБОТА №3**

**РАЗРАБОТКА АВТОНОМНЫХ УСТРОЙСТВ ИЗМЕРЕНИЯ**  
**ХАРАКТЕРИСТИК ОКРУЖАЮЩЕЙ СРЕДЫ**

**Цель работы:**

Изучить основные принципы приема данных с различных сенсоров и  
датчиков, их обработки и индикации. Изучить и проверить работу датчика  
температуры и влажности dht11 , датчика температуры DALLAS 18b20,  
газометра GazSensorV1.3.

**3.1 Краткие теоретические сведения**

В технике часто нужно разрабатывать устройства, которые не должны  
быть сопряжены с компьютером. Чаще всего требуется производить  
считывания с датчиков каких-либо значений, выполнять их обработку и  
выводить информацию не через последовательный порт в компьютер, а,  
например, на ЖК-дисплей. Также возможна ситуация, когда необходимо  
получить некоторую реакцию на внешний раздражитель. Ярким примером  
может послужить система подогрева, которая автоматически  
включается/выключается при определенных значениях температуры. Также  
таким примером может быть обыкновенная сигнализация.  
Arduino Uno может получать питание не только через подключение  
USB, но и от внешнего источника питания. Внешнее питание может  
подаваться через преобразователь напряжения AC/DC (блок питания) или  
непосредственно от аккумуляторной батареи. Преобразователь напряжения  
подключается посредством разъема 2.1 мм с центральным положительным  
полюсом. Провода от батареи подключаются к выводам Gnd и Vin разъема  
питания.

Датчик Dallas18b20 позволяет измерять температуру с большей

точностью. Он работает по шине One-Wire. Для устройств, работающих по

этой шине, в среде Arduino используется библиотека OneWire.

Датчик может находиться в металлическом корпусе, который

заизолирован термоусадочной трубкой. Поэтому его можно погружать в

воду. На металле трубка прилегает очень плотно, на кабеле слабее, поэтому

есть вероятность того, что внутрь может просочиться вода. С целью

избегания данной ситуации желательно не погружать датчик в воду целиком.

Датчик содержит уникальный 64-битный ROM код, состоящий из 8

битов, определяющих код серии, 48 бит уникального номера и 8 бит

помехоустойчивого CRC кода.

Информация об измеренной температуре хранится в оперативной

памяти датчика, которая состоит из 9 байт:

1 и 2 байты хранят информацию о температуре.

3 и 4 байты хранят соответственно верхний и нижний пределы

температуры.

5 и 6 байты зарезервированы.

7 и 8 байты используются для сверхточного измерения температуры.

9 байт хранит помехоустойчивый CRC код предыдущих 8 байт.

Подключение датчика к плате имеет свои особенности. Он использует  
так называемое «паразитное питание». Паразитное питание – это питание  
электронного устройства напряжением каких-либо сигналов без  
использования специально выделенной шины питания.

Интерфейс 1 -Wire использует паразитное питание.  
 1 -Wire – двунаправленная шина связи для устройств с низкоскоростной  
передачей данных (обычно 15,4 Кбит/с, максимум 125 Кбит/с), в которой  
данные передаются по цепи питания (то есть всего используются два провода  
– один для заземления, а второй для питания и данных; в некоторых случаях  
используют и отдельный провод питания).  
 Обычно интерфейс 1 -Wire используется для того, чтобы связываться с  
недорогими простыми устройствами, такими как, например, цифровые  
термометры и измерители параметров внешней среды.  
 Собственно, схема подключения датчика изображена на рисунке 3.1.

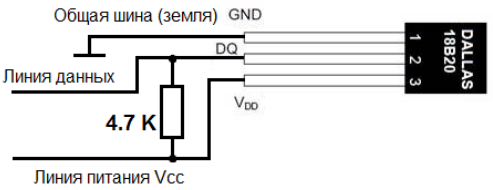


Рисунок 3.1 – Схема подключения датчика

**3.2 Порядок выполнения работы**

1. Ознакомиться с теоретическими сведениями.

2. Ознакомится со своим вариантом задания на лабораторную работу.

3. Импортировать и подключить к проекту нужные для работы  
библиотеки.

4. Разработать и спроектировать схему устройства.

5. Написать программу, которая выполняет поставленню задачу.

6. Произвести компиляцию проектов.

7. Собрать разработанную схему и загрузить программу в контролер.  
Проверить правильность работы устройства.

8. Сделать отчет по выполненной работе.

**3.3 Постановка задачи**

Разработать устройство, которое зажигает один светодиод, когда  
температура воздуха падает ниже определенного значения и  
зажигает второй светодиод, когда температура превышает  
некоторое значение. Значения выбирать самостоятельно.

**3.4 Текст программы**

#include <OneWire.h>

#include <DallasTemperature.h>

#define ONE\_WIRE\_BUS 2

#define ANALOG\_LED\_1 9

#define ANALOG\_LED\_2 10

#define THRESHOLD\_TEMP 27

OneWire oneWire(ONE\_WIRE\_BUS);

DallasTemperature sensors(&oneWire);

void setup(void)

{

sensors.begin();

}

void loop(void)

{

sensors.requestTemperatures();

float t = sensors.getTempCByIndex(0);

if (t > THRESHOLD\_TEMP ) {

analogWrite(ANALOG\_LED\_1, 0x80);

analogWrite(ANALOG\_LED\_2, 0x00);

} else {

analogWrite(ANALOG\_LED\_1, 0);

analogWrite(ANALOG\_LED\_2, 0x80);

}

delay(10);

}

**1.5 Особенности IDE Arduino выявленные в ходе выполнения лабораторной работы**

В ходе лабораторной работы было выявлено, что функции setup и loop вызываются из функции main, исходный текст которой расположен в hardware/arduino/cores/main.c, также в данной директории находится большинство стандартных функций.

**Выводы**

В ходе лабораторной работы были изучены основные принципы приема данных с различных сенсоров и датчиков, их обработки и индикацию. Было разработано работоспособное приложение, которое зажигает определенный светодиод в зависимости от полученных данных с датчика температуры DALLAS 18b20.