**Урок 3. Обзор доступных средств индикации (светодиоды, светодиодные матрицы, дисплеи)**

Задание 1. Необходимо подключить светодиод и потенциометр, для того, чтобы управлять яркостью нашего светодиода. Светодиод можно выбрать любой.  
Задание 2.\* Необходимо любыми доступными средствами индикации отобразить сигнал с любого датчика. Например: взять датчик температуры и отображать на светодиодной матрице грустный смайлик, если погода плохая, и веселый - если хорошая.

Задание 1:

int sensorPin = A2; // select the input pin for the potentiometer

int ledPin = 2; // select the pin for the LED

int sensorValue = 0; // variable to store the value coming from the sensor

void setup() {

// declare the ledPin as an OUTPUT:

pinMode(ledPin, OUTPUT);

}

void loop() {

// read the value from the sensor:

sensorValue = analogRead(sensorPin)/4;

// turn the ledPin on

analogWrite(ledPin, sensorValue);

delay(10);

Результат работы кода показан на видео: PWM\_led.mp4

Задание 2:

#include <OneWire.h>

/\*

\* Описание взаимодействия с цифровым датчиком ds18b20

\* Подключение ds18b20 к ардуино через пин 8

\*/

OneWire ds(8); // Создаем объект OneWire для шины 1-Wire, с помощью которого будет осуществляться работа с датчиком

int i;

void setup(){

Serial.begin(9600);

int initPins[6]={3,4,5,6,7,9};

for (i=0; i<=5; i++)

{ pinMode(initPins[i], OUTPUT);

}

}

void loop(){

// Определяем температуру от датчика DS18b20

byte data[2]; // Место для значения температуры

ds.reset(); // Начинаем взаимодействие со сброса всех предыдущих команд и параметров

ds.write(0xCC); // Даем датчику DS18b20 команду пропустить поиск по адресу. В нашем случае только одно устрйоство

ds.write(0x44); // Даем датчику DS18b20 команду измерить температуру. Само значение температуры мы еще не получаем - датчик его положит во внутреннюю память

delay(1000); // Микросхема измеряет температуру, а мы ждем.

ds.reset(); // Теперь готовимся получить значение измеренной температуры

ds.write(0xCC);

ds.write(0xBE); // Просим передать нам значение регистров со значением температуры

// Получаем и считываем ответ

data[0] = ds.read(); // Читаем младший байт значения температуры

data[1] = ds.read(); // А теперь старший

// Формируем итоговое значение:

// - сперва "склеиваем" значение,

// - затем умножаем его на коэффициент, соответсвующий разрешающей способности (для 12 бит по умолчанию - это 0,0625)

float temperature = ((data[1] << 8) | data[0]) \* 0.0625;

// Выводим полученное значение температуры в монитор порта

Serial.println(temperature);

indicator(temperature);

}

int indicator(int level)

{ int numberPins[12];

if (level>26)

{int numberPins[12]={3,1,4,0,5,1,6,1,7,1,9,1};

for (i=0; i<=11; i+=2)

{ digitalWrite(numberPins[i], numberPins[i+1]);

}

}

else

{int numberPins[12]={3,1,4,1,5,0,6,0,7,1,9,0};

for (i=0; i<=11; i+=2)

{ digitalWrite(numberPins[i], numberPins[i+1]);

}

}

}

Работа кода показана на видео LH\_temp.mp4