

Московский Физико-Технический Институт
(Государственный университет)

Курсовой проект по курсу "Микроконтроллеры"

МЕТЕОСТАНЦИЯ

Карцев Вадим, Б01-904
Тяжкороб Ульяна, Б01-909
Янович Максим, Б01-909
Безянова Елизавета, С01-919

Долгопрудный
2021

1 Описание проекта

1.1 Возможности

Собранное устройство позволяет в реальном времени отслеживать параметры окружающей среды, такие, как температура, влажность и давление. Данные представляются в наиболее популярных величинах измерения.

Кроме того, на базе атмосферного давления устройство умеет определять примерную высоту над уровнем моря.

Также устройство отображает уведомления о слишком низкой или высокой влажности и уведомления о неисправности сенсора.

1.2 Управление

Управление устройством осуществляется с помощью одной кнопки, которая переключает экраны, отображающие различные категории значений: Температура (Temp), Давление (Pressure), Высота (Altitude) и Влажность (Humidity).

2 Документация

2.1 Используемые модули

- Arduino Nano (ATMega 328p, 16MHz)
- Комбинированный датчик условий окружающей среды BME280
- Экран монохромный HX1230
- Сенсорная кнопка (цифровая)

2.2 Используемые библиотеки

- U8g2lib.h
- Adafruit_Sensor.h
- Adafruit_BME280.h

3 Подключение модулей

Display

CLK → *D4*

DIN → *D5*

CE → *D7*

RST → *D8*

BME280 Sensor

SCL → *D13*

SDA → *D11*

SCB → *D10*

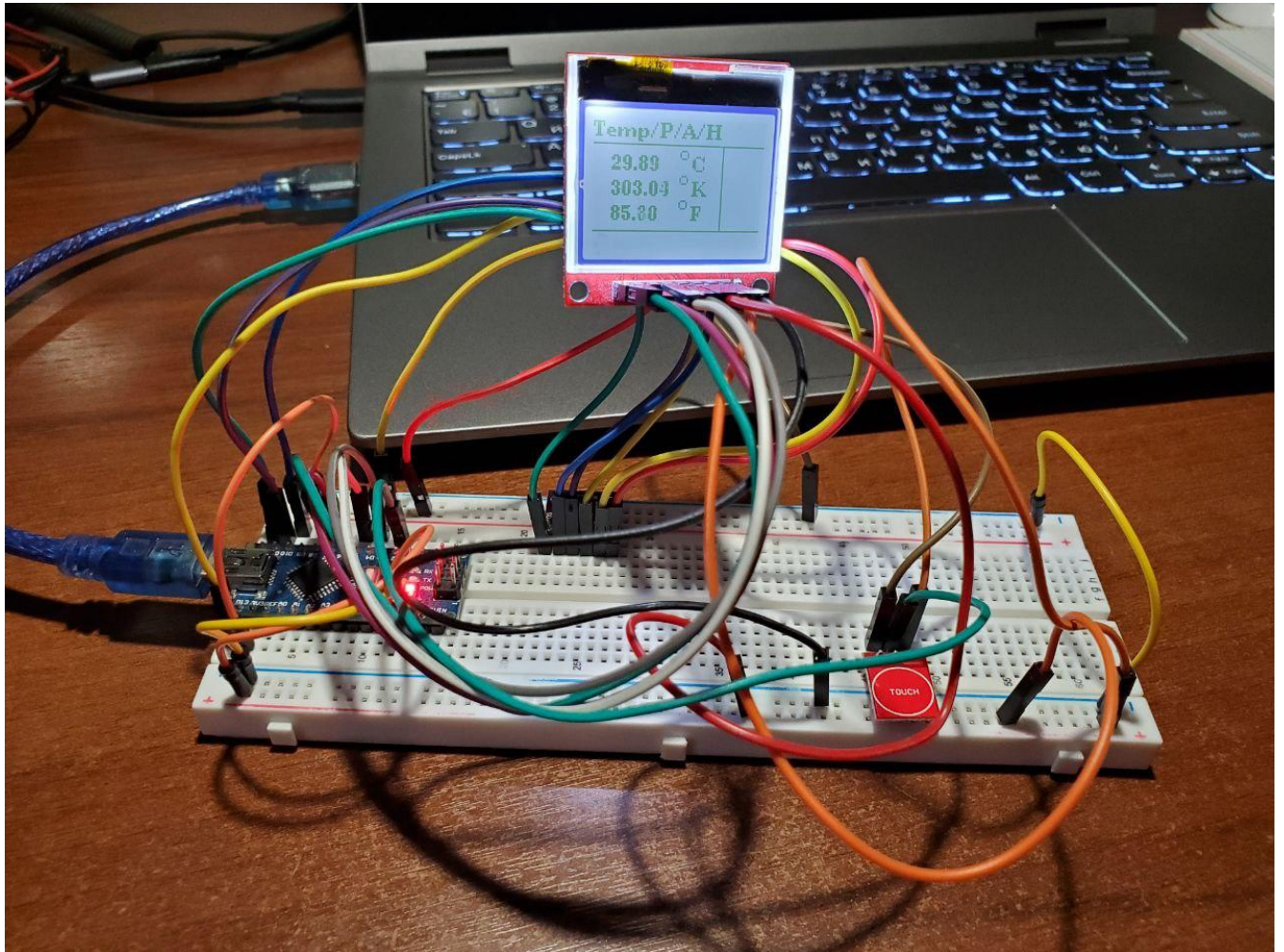
SDO → *D12*

Touch Button

I/O → *D2*

4 Фотографии проекта

4.1 Устройство



4.2 Интерфейс

Temp/P/A/H	
30.56 °C	
303.71 °K	
87.01 °F	
too dry	

Температура

T/Pressure/A/H	
97.71 kPa	
732.87 mm hg	
too dry	

Давление

T/P/Altitude/H	
305.29 m	
Approximate	
Error ~20m	
too dry	

Высота

T/P/A/Humidity	
38.61 %	
Normal: 40-60%	
too dry	

Влажность

5 Исходный код проекта

```
1  #include <U8g2lib.h>
2  #include <Adafruit_Sensor.h>
3  #include <Adafruit_BME280.h>
4
5  // define pins for sensor (VCC -> 3V3, GND -> GND)
6  #define BME_SCK 13 // SCL pin on sensor
7  #define BME_MISO 12 // SDO pin on sensor
8  #define BME_MOSI 11 // SDA pin on sensor
9  #define BME_CS 10 // SCB pin on sensor
10
11 // define pins for display (VCC -> 5V, BL -> 3V3, GND -> GND)
12 #define DISPLAY_CLOCK 4 // CLK pin on display
13 #define DISPLAY_DATA 5 // DIN pin on display
14 #define DISPLAY_CS 7 // CE pin on display
15 #define DISPLAY_RESET 8 // RST pin on display
16
17 // define pins for key (5V -> 5V, GND -> GND)
18 #define TOUCH_KEY 2
19
20 #define SEALEVELPRESSURE_HPA (1013.25)
21
22 Adafruit_BME280 sensor(BME_CS, BME_MOSI, BME_MISO, BME_SCK);
23 U8G2_HX1230_96X68_F_3W_SW_SPI display(U8G2_R0, DISPLAY_CLOCK, DISPLAY_DATA, DISPLAY_CS, DISPLAY_RESET);
24
25
26 void sensorError();
27 void drawFrame(int frame);
28 int checkKey();
29 int next(int frame);
30
31
32 int FrameNumber = 0;
33 int Time_from_last_update = 0;
34 int SensorError = 0;
35
36
37 void setup(void)
38 {
39     display.begin();
40
41     bool status;
42     status = sensor.begin();
43     if (!status) SensorError = 1;
44     delay(100); //Sensor bootup
45     drawFrame(FrameNumber);
46 }
47
48 void loop(void)
49 {
50     int frame = FrameNumber;
51     int command = checkKey();
52
53     if (command == 1) FrameNumber = next(FrameNumber);
54 }
```

```

55     if ((frame != FrameNumber) or (Time_from_last_update > 1000))
56     {
57         Time_from_last_update = 0;
58         drawFrame(FrameNumber);
59     }
60
61     delay(1);
62     Time_from_last_update++;
63 }
64
65
66 void drawFrame(int frame)
67 {
68     display.clearBuffer();
69     display.setFont(u8g2_font_ncenB08_tr);
70
71     display.drawLine(0,13,96,13);
72     display.drawLine(0,58,96,58);
73
74     drawBar();
75
76     switch(frame)
77     {
78         case(0):
79         {
80             display.drawStr(0, 10, "Temp/P/A/H");
81             float temp_celsius = sensor.readTemperature();
82             display.setCursor(10, 27);
83             display.print(temp_celsius, 2); // Temperature in Celsius
84             display.drawCircle(50, 18, 2);
85             display.drawStr(52, 27, " C");
86             display.setCursor(10, 40);
87             display.print(temp_celsius + 273.15, 2); // Temperature in Kelvins
88             display.drawCircle(50, 31, 2);
89             display.drawStr(52, 40, " K");
90             display.setCursor(10, 53);
91             display.print(temp_celsius * 1.8 + 32, 2); // Temperature in Farengeith
92             display.drawCircle(50, 44, 2);
93             display.drawStr(52, 53, " F");
94             display.drawLine(72,13,72,58);
95             break;
96         }
97         case(1):
98         {
99             display.drawStr(0, 10, "T/Pressure/A/H");
100             float pressure_pa = sensor.readPressure();
101             display.setCursor(10, 27);
102             display.print(pressure_pa / 1000.0F, 2);
103             display.drawStr(50, 27, "kPa");
104             display.setCursor(10, 40);
105             display.print(pressure_pa / 133.332, 2);
106             display.drawStr(50, 40, "mm hg");
107             break;
108         }
109         case(2):
110         {

```

```

111         display.drawStr(0, 10, "T/P/Altitude/H");
112         display.setCursor(10, 27);
113         display.print(sensor.readAltitude(SEALEVELPRESSURE_HPA), 2);
114         display.drawStr(50, 27, "m");
115         display.drawStr(10, 40, "Approximate");
116         display.drawStr(10, 53, "Error ~20m");
117         break;
118     }
119     case(3):
120     {
121         display.drawStr(0, 10, "T/P/A/Humidity");
122         display.setCursor(10, 27);
123         display.print(sensor.readHumidity(), 2);
124         display.drawStr(40, 27, "%");
125         display.drawStr(10, 40, "Normal: 40-60%");
126         break;
127     }
128 }
129
130 display.sendBuffer();
131 }
132
133
134 int checkKey()
135 {
136     int command = 0;
137
138     if(digitalRead(TOUCH_KEY))
139     {
140         command = 1;
141         while(digitalRead(TOUCH_KEY));
142     }
143
144     return command;
145 }
146
147
148 int next(int frame)
149 {
150     int NewFrame = frame + 1;
151     if(NewFrame > 3) NewFrame = 0;
152     return NewFrame;
153 }
154
155
156 void drawBar()
157 {
158     if(SensorError) display.drawStr(0, 67, "err");
159     if(sensor.readHumidity() < 40) display.drawStr(55, 67, "too dry");
160     else if (sensor.readHumidity() > 60) display.drawStr(55, 67, "too wet");
161 }
162
163
164 //end;

```