

```

/*
  Скetch к проекту "Домашняя метеостанция на датчике CO2
  MH-Z19B"

  Страница проекта (схемы, описания): https://lavent.ru/domashnyaya-meteostanciya-na-datchike-co2-mh-z19b/
  Автор: Андрей Л., ООО "ЛАВЕНТ"
  http://www.lavent.ru
*/

#include <Adafruit_BME280.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SPITFT.h>
#include <Adafruit_SPITFT_Macros.h>
#include <gfxfont.h>
#include <avr/pgmspace.h>
#include <BME280I2C.h>
#include <Adafruit_SSD1306.h>
#include <SoftwareSerial.h>
#include <Wire.h>

SoftwareSerial mySerial(A0, A1); // RX, TX сенсора
byte cmd[9] = {0xFF, 0x01, 0x86, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x79};
char response[9];

#define SERIAL_BAUD 9600
#define OLED_RESET 4
#define SCREEN_WIDTH 128 // OLED display width, in pixels
#define SCREEN_HEIGHT 64 // OLED display height, in pixels

Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, -1);
BME280I2C bme = BME280I2C();

const uint8_t Tempa[] PROGMEM = {
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x7c, 0x00,
  0x00, 0xfe, 0x00, 0x01, 0xc7, 0x00, 0x01,

```

```
    0x83, 0x00, 0x01, 0x83, 0x00, 0x07, 0x01, 0x80, 0x07,
0x01, 0xf0, 0x07, 0x11, 0xf8, 0x07, 0x39,
    0xf0, 0x07, 0x39, 0x80, 0x07, 0x39, 0x80, 0x07, 0x39,
0xf0, 0x07, 0x39, 0xf8, 0x07, 0x39, 0xf0,
    0x07, 0x39, 0x80, 0x07, 0x39, 0x80, 0x07, 0x39, 0x80,
0x07, 0x39, 0xf0, 0x07, 0x39, 0xf8, 0x07,
    0x39, 0xf0, 0x07, 0x39, 0x80, 0x07, 0x39, 0x80, 0x07,
0x39, 0x80, 0x0f, 0x39, 0xe0, 0x1e, 0x38,
    0xf0, 0x38, 0x7c, 0x78, 0x31, 0xff, 0x18, 0x31, 0xff,
0x18, 0x77, 0xbf, 0x8c, 0x67, 0x3f, 0x8c,
    0x6e, 0x7f, 0xec, 0x6e, 0xff, 0xec, 0x6f, 0xff, 0xec,
0x6f, 0xff, 0xec, 0x67, 0xff, 0x8c, 0x67,
    0xff, 0x9c, 0x71, 0xff, 0x18, 0x38, 0x7e, 0x70, 0x1e,
0x00, 0xf0, 0x1e, 0x00, 0xf0, 0x0f, 0xc7,
    0xe0, 0x07, 0xff, 0x00, 0x00, 0xfc, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,
};
```

```
const uint8_t Air[] PROGMEM = {
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0xc0, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x7f,
```

0xff, 0xff, 0xff, 0x07, 0xe0, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x7f,
0xff, 0xff, 0xfc, 0x3c, 0x60, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x60,
0x00, 0x00, 0x00, 0xe0, 0x60, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x60,
0x00, 0x00, 0x07, 0x06, 0x60, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x60,
0x00, 0x00, 0x38, 0x1e, 0x60, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x60,
0x00, 0x00, 0x21, 0x9e, 0x60, 0x00, 0x00,
0x00, 0x00, 0x00, 0x1f, 0xf8, 0x00, 0x18, 0x00, 0x60,
0x00, 0x00, 0x27, 0x9e, 0x60, 0x00, 0x00,
0x00, 0x00, 0x00, 0x3f, 0xfc, 0x00, 0x78, 0x00, 0x60,
0x00, 0x00, 0x27, 0x9e, 0x60, 0x00, 0x00,
0x00, 0x00, 0x00, 0x78, 0x3e, 0x00, 0xe0, 0x00, 0x60,
0x00, 0x00, 0x27, 0x9e, 0x60, 0x00, 0x00,
0x00, 0x00, 0x00, 0xe0, 0x0f, 0x03, 0xc0, 0x00, 0x60,
0x00, 0x00, 0x27, 0x9e, 0x60, 0x00, 0x00,
0x00, 0x00, 0x03, 0xc0, 0x07, 0xff, 0x80, 0x00, 0x60,
0x00, 0x00, 0x27, 0x9e, 0x60, 0x00, 0x00,
0x00, 0x00, 0x07, 0x00, 0x03, 0xff, 0x00, 0x00, 0x60,
0x00, 0x00, 0x27, 0x9e, 0x60, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0xfc, 0x00, 0x00, 0x60,
0x00, 0x00, 0x27, 0x9e, 0x60, 0x00, 0x00,
0x00, 0x00, 0x00, 0x1f, 0xf8, 0x00, 0x18, 0x00, 0x60,
0x00, 0x00, 0x27, 0x9e, 0x60, 0x00, 0x00,
0x00, 0x00, 0x00, 0x3f, 0xfc, 0x00, 0x78, 0x00, 0x60,
0x00, 0x00, 0x27, 0x9e, 0x60, 0x00, 0x00,
0x00, 0x00, 0x00, 0x78, 0x3e, 0x00, 0xe0, 0x00, 0x60,
0x00, 0x00, 0x27, 0x9e, 0x60, 0x00, 0x00,
0x00, 0x00, 0x00, 0xe0, 0x0f, 0x03, 0xc0, 0x00, 0x60,
0x00, 0x00, 0x27, 0x9e, 0x60, 0x00, 0x00,
0x00, 0x00, 0x03, 0xc0, 0x07, 0xff, 0x80, 0x00, 0x60,
0x00, 0x00, 0x27, 0x9e, 0x60, 0x00, 0x00,
0x00, 0x00, 0x07, 0x00, 0x03, 0xff, 0x00, 0x00, 0x60,
0x00, 0x00, 0x27, 0x9e, 0x60, 0x00, 0x00,
0x00, 0x00, 0x00, 0x03, 0xc0, 0xfc, 0x00, 0x00, 0x60,

0x00, 0x00, 0x20, 0x00, 0x60, 0x00, 0x00,
0x00, 0x00, 0x00, 0x1f, 0xf8, 0x00, 0x18, 0x00, 0x60,
0x00, 0x00, 0x20, 0x00, 0x60, 0x00, 0x00,
0x00, 0x00, 0x00, 0x3f, 0xfc, 0x00, 0x78, 0x00, 0x60,
0x00, 0x00, 0x27, 0x9e, 0x60, 0x00, 0x00,
0x00, 0x00, 0x00, 0x78, 0x3e, 0x00, 0xe0, 0x00, 0x60,
0x00, 0x00, 0x27, 0x9e, 0x60, 0x00, 0x00,
0x00, 0x00, 0x00, 0xe0, 0x0f, 0x03, 0xc0, 0x00, 0x60,
0x00, 0x00, 0x27, 0x9e, 0x60, 0x00, 0x00,
0x00, 0x00, 0x03, 0xc0, 0x07, 0xff, 0x80, 0x00, 0x60,
0x00, 0x00, 0x27, 0x9e, 0x60, 0x00, 0x00,
0x00, 0x00, 0x07, 0x00, 0x03, 0xff, 0x00, 0x00, 0x60,
0x00, 0x00, 0x27, 0x9e, 0x60, 0x00, 0x00,
0x00, 0x00, 0x00, 0x03, 0xc0, 0xfc, 0x00, 0x00, 0x60,
0x00, 0x00, 0x27, 0x9e, 0x60, 0x00, 0x00,
0x00, 0x00, 0x00, 0x1f, 0xf8, 0x00, 0x18, 0x00, 0x60,
0x00, 0x00, 0x27, 0x9e, 0x60, 0x00, 0x00,
0x00, 0x00, 0x00, 0x3f, 0xfc, 0x00, 0x78, 0x00, 0x60,
0x00, 0x00, 0x27, 0x9e, 0x60, 0x00, 0x00,
0x00, 0x00, 0x00, 0x78, 0x3e, 0x00, 0xe0, 0x00, 0x60,
0x00, 0x00, 0x27, 0x9e, 0x60, 0x00, 0x00,
0x00, 0x00, 0x00, 0xe0, 0x0f, 0x83, 0xc0, 0x00, 0x60,
0x00, 0x00, 0x27, 0x9e, 0x60, 0x00, 0x00,
0x00, 0x00, 0x03, 0xc0, 0x07, 0xff, 0x80, 0x00, 0x60,
0x00, 0x00, 0x27, 0x9e, 0x60, 0x00, 0x00,
0x00, 0x00, 0x07, 0x00, 0x03, 0xff, 0x00, 0x00, 0x60,
0x00, 0x00, 0x27, 0x9e, 0x60, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0xfc, 0x00, 0x00, 0x60,
0x00, 0x00, 0x21, 0x9e, 0x60, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x60,
0x00, 0x00, 0x38, 0x1e, 0x60, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x60,
0x00, 0x00, 0x0f, 0x06, 0x60, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x60,
0x00, 0x00, 0x00, 0xe0, 0x60, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x7f,
0xff, 0xff, 0xfc, 0x3c, 0x60, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x7f,


```
    0x07, 0xff, 0xe0, 0x00, 0x00, 0x0f, 0xff, 0xf0, 0x00,
0x00, 0x1f, 0x81, 0xf8, 0x00, 0x00, 0x3e,
    0x00, 0x7c, 0x00, 0x00, 0x7c, 0x00, 0x1e, 0x00, 0x00,
0xf8, 0x00, 0x0f, 0x00, 0x00, 0xf0, 0x00,
    0x0f, 0x00, 0x01, 0xe0, 0x00, 0x07, 0x80, 0x01, 0xe0,
0x00, 0x07, 0x80, 0x03, 0xc0, 0x00, 0x03,
    0xc0, 0x07, 0xc0, 0x00, 0x03, 0xe0, 0x0f, 0xc0, 0x00,
0x03, 0xf0, 0x1f, 0x80, 0x00, 0x01, 0xf8,
    0x3e, 0x00, 0x00, 0x00, 0x7c, 0x78, 0x0e, 0x0e, 0x00,
0x1e, 0x78, 0x1f, 0x1f, 0x00, 0x1e, 0xf0,
    0x31, 0x31, 0x80, 0x0f, 0xf0, 0x20, 0x20, 0x80, 0x07,
0xe0, 0x20, 0x20, 0x98, 0x07, 0xe0, 0x20,
    0x20, 0xbc, 0x07, 0xe0, 0x20, 0x20, 0xa4, 0x07, 0xe0,
0x31, 0x31, 0x84, 0x07, 0xe0, 0x1f, 0x1f,
    0x08, 0x07, 0xf0, 0x0e, 0x0e, 0x10, 0x0f, 0x70, 0x00,
0x00, 0x3c, 0x0f, 0x78, 0x00, 0x00, 0x00,
    0x1e, 0x7c, 0x00, 0x00, 0x00, 0x3e, 0x3e, 0x00, 0x00,
0x00, 0x7c, 0x1f, 0xff, 0xff, 0xff, 0xf8,
    0x0f, 0xff, 0xff, 0xff, 0xf0, 0x03, 0xff, 0xff, 0xff,
0xc0, 0x00, 0x7f, 0xff, 0xfe, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,
};
```

```
const uint8_t Gas[] PROGMEM = {
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x7f, 0xf0,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0xff, 0xfe,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
```


0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0xc0, 0x3f, 0xe0,
0x1e, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0xf0, 0xff, 0xf8,
0x7e, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0xff, 0xff, 0xff,
0xfe, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0xff, 0xff, 0xff,
0xfc, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0xff, 0xff, 0xff,
0xfc, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xff, 0xff, 0xff,
0xf8, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xff, 0xff, 0xff,
0xf8, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x7f, 0xff, 0xff,
0xf0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x7f, 0xff, 0xff,
0xf0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x7f, 0xff, 0xff,
0xf0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3f, 0xf0, 0x7f,
0xf0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3f, 0x80, 0x0f,
0xe0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3f, 0x00, 0x07,
0xe0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x78, 0x3c, 0x3f, 0xe3,
0xe0, 0xf0, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x01, 0xfc, 0x3c, 0xff, 0xf9,
0xe1, 0xfc, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x07, 0xfc, 0x79, 0xff, 0xfc,
0xf1, 0xff, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x0f, 0xff, 0xf3, 0xc8, 0x9e,
0x7f, 0xff, 0x80, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x1f, 0xff, 0xe3, 0xc8, 0x9e,
0x7f, 0xff, 0xc0, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x1f, 0xff, 0xe7, 0xff, 0xff,
0x3f, 0xff, 0xc0, 0x00, 0x00, 0x00, 0x00,


```
};
```

```
const uint8_t Humi[] PROGMEM = {  
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x06,  
    0x00, 0x00, 0x00, 0x00, 0x0f, 0x00, 0x00,  
    0x00, 0x00, 0x1f, 0x80, 0x00, 0x00, 0x00, 0x1f, 0x80,  
    0x00, 0x00, 0x00, 0x1f, 0x80, 0x00, 0x00,  
    0x00, 0x39, 0xc0, 0x00, 0x00, 0x00, 0xf9, 0xf0, 0x00,  
    0x00, 0x00, 0xf1, 0xf0, 0x00, 0x00, 0x71,  
    0xf0, 0xf8, 0x00, 0x00, 0x71, 0xe6, 0x78, 0x00, 0x00,  
    0xfb, 0xe3, 0x78, 0x00, 0x03, 0xff, 0xc1,  
    0x3c, 0x00, 0x03, 0xff, 0xc1, 0x3c, 0x00, 0x07, 0xdf,  
    0x00, 0x0c, 0x03, 0x0f, 0x8f, 0x00, 0x0e,  
    0x03, 0x0f, 0x8f, 0x00, 0x0e, 0x03, 0x0f, 0x0f, 0x00,  
    0xce, 0x0f, 0x9f, 0x07, 0xc0, 0x06, 0x0f,  
    0x9c, 0x07, 0xc0, 0x06, 0x1f, 0xfc, 0x03, 0xe0, 0x06,  
    0x1c, 0xf8, 0x01, 0xe0, 0x06, 0x3c, 0xf8,  
    0x01, 0xf0, 0x06, 0x3c, 0xf0, 0x00, 0xf0, 0x0e, 0x70,  
    0x70, 0x00, 0xf0, 0x0e, 0x70, 0x70, 0x00,  
    0x38, 0x3c, 0x70, 0x70, 0x00, 0x38, 0x3c, 0x60, 0x30,  
    0x00, 0x38, 0x7c, 0x60, 0x30, 0x00, 0x3d,  
    0xfc, 0x60, 0x30, 0x00, 0x1f, 0xf8, 0x60, 0x30, 0x00,  
    0x1f, 0xf0, 0x70, 0x70, 0x01, 0x1f, 0xc0,  
    0x7c, 0xf0, 0x01, 0xcf, 0x00, 0x7f, 0xe0, 0x00, 0xcc,  
    0x00, 0x3f, 0xe0, 0x00, 0xcc, 0x00, 0x1f,  
    0xc0, 0x01, 0xcc, 0x00, 0x1f, 0xc0, 0x01, 0xcc, 0x00,  
    0x0f, 0x80, 0x01, 0x0c, 0x00, 0x01, 0xc0,  
    0x01, 0x1c, 0x00, 0x01, 0xc0, 0x03, 0x1c, 0x00, 0x01,  
    0xc0, 0x02, 0x3c, 0x00, 0x00, 0xe0, 0x38,  
    0x38, 0x00, 0x00, 0xf0, 0x30, 0xf8, 0x00, 0x00, 0xf8,  
    0x01, 0xf0, 0x00, 0x00, 0x7c, 0x03, 0xf0,  
    0x00, 0x00, 0x3f, 0x8f, 0xe0, 0x00, 0x00, 0x1f, 0xff,  
    0xc0, 0x00, 0x00, 0x1f, 0xff, 0xc0, 0x00,  
    0x00, 0x0f, 0xff, 0x00, 0x00, 0x00, 0x03, 0xfc, 0x00,  
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,  
    0x00, 0x00, 0x00, 0x00,  
};
```

```
void setup() {

    Wire.begin();
    Serial.begin(9600);
    display.begin(SSD1306_SWITCHCAPVCC, 0x3C); // обнаружение
I2C адреса 0x3C (для 128x64)
    bme.begin();
    mySerial.begin(9600);

}
```

```
void displayHighPpm() {

    int responseHigh = (int) response[2];
    int responseLow = (int) response[3];
    int ppm = (256 * responseHigh) + responseLow;

    display.clearDisplay();
    display.setTextColor(WHITE);
    display.setTextSize(4);
    display.setCursor(5, 15);
    Serial.print(ppm);
    display.print(ppm);
    display.setTextColor(WHITE);
    display.setTextSize(2);
    display.setCursor(80, 45);
    display.print(" CO2");

}
```

```
void displayTemp() {

    float temp(NAN), hum(NAN), pres(NAN);
    BME280::TempUnit tempUnit(BME280::TempUnit_Celsius);
    BME280::PresUnit presUnit(BME280::PresUnit_Pa);
    bme.read(pres, temp, hum, tempUnit, presUnit);

    display.clearDisplay();
    display.drawBitmap(13, 5, Tempa, 24, 47, WHITE);

}
```

```

display.setTextColor(WHITE);
display.setTextSize(3);
display.setCursor(55, 20);
Serial.print(temp, 0);
display.print(temp, 0);
display.setCursor(100, 20);
display.print("C");
}

void displayHumid() {

    float temp(NAN), hum(NAN), pres(NAN);
    BME280::TempUnit tempUnit(BME280::TempUnit_Celsius);
    BME280::PresUnit presUnit(BME280::PresUnit_Pa);
    bme.read(pres, temp, hum, tempUnit, presUnit);

    display.clearDisplay();
    display.drawBitmap(5, 7, Humi, 40, 52, WHITE);
    display.setTextColor(WHITE);
    display.setTextSize(3);
    display.setCursor(55, 20);
    Serial.print(hum, 0);
    display.print(hum, 0);
    display.setCursor(100, 20);
    display.print("%");
}

void displayPpm() {

    int responseHigh = (int) response[2];
    int responseLow = (int) response[3];
    int ppm = (256 * responseHigh) + responseLow;

    display.clearDisplay();
    display.drawBitmap(5, 10, CO2, 40, 36, WHITE);
    display.setTextColor(WHITE);
    display.setTextSize(3);
    display.setCursor(55, 20);

```

```

Serial.print(ppm);
display.print(ppm);

void loop() {

    mySerial.write(cmd, 9); //запрос PPM CO2
    mySerial.readBytes(response, 9);
    int responseHigh = (int) response[2];
    int responseLow = (int) response[3];
    int ppm = (256*responseHigh)+responseLow;

    if (ppm > 1000) {
//установка уровня CO2, выше которого появляется противогаз
        display.clearDisplay();
        display.drawBitmap(0, 0, Air, 128, 64, WHITE);
        display.display();
        delay(1000);
        for (int i = 0; i < 2; i++)
        {
            delay(200);
        }
        display.clearDisplay();
        display.drawBitmap(0, 0, Gas, 128, 64, WHITE);
        display.display();
        delay(1000);
        for (int i = 0; i < 2; i++)
        {
            delay(200);
        }
        display.clearDisplay();
        display.drawBitmap(0, 0, Air, 128, 64, WHITE);
        display.display();
        delay(1000);
        for (int i = 0; i < 2; i++)
        {
            delay(200);
        }
    }
}

```

```
displayHighPpm();
display.display();
delay(3000);
}

else {
display.clearDisplay();
displayTemp();
display.display();
delay(3000);

display.clearDisplay();
displayHumid();
display.display();
delay(3000);

display.clearDisplay();
displayPpm();
display.display();
delay(3000);
}
}
```