**Приложение к статье**

ПРОСТОЙ РЕГУЛЯТОР ПОДАЧИ CO2 ДЛЯ ЛАБОРАТОРНОГО

ИНКУБАТОРА КЛЕТОК ИЗ ДОСТУПНЫХ КОМПОНЕНТОВ

© 2022 г. М. А. Суслов1*,* Г. В.Сибгатуллина1*,* Д. В. Самигуллин1, 2, \*

**Программный код для автоматизации контроля концентрации углекислого**

**газа в камере биоинкубатора**

// *подключение библиотек функций и процедур Arduino*

#include <SoftwareSerial.h>

#include <Wire.h>

#include <SPI.h>

#include <LiquidCrystal\_I2C.h>

LiquidCrystal\_I2C lcd (0X27,16,2);

SoftwareSerial swSerial(A0, A1); // RX, TX

int GatePin=13;

int k=0; // *вспомогательная переменная (флаг)*

*// настройка связи Arduino с компьютером, инициализация LCD дисплея и цифровых портов ввода/вывода*

void setup() { pinMode(GatePin,OUTPUT); digitalWrite(GatePin,LOW); Serial.begin(9600); swSerial.begin(9600); lcd.init();

lcd.backlight();

lcd.setCursor(0, 0);

lcd.print("Starting...");

// *команда отключения автокалибровки датчика MH-Z16*

byte Calibr\_OFF\_cmd[9] = {0xFF, 0x01, 0x79, 0x00, 0x00, 0x00, 0x00, 0x00, 0x86};

swSerial.write(Calibr\_OFF\_cmd,9);

// *проверка связи с датчиком с помощью вычисления контрольной суммы*

byte setrangeA\_cmd[9] = {0xFF, 0x01, 0x99, 0x00, 0x00, 0x00, 0x13, 0x88, 0xCB};

*//*

unsigned char setrangeA\_response[9];

swSerial.write(setrangeA\_cmd,9); swSerial.readBytes(setrangeA\_response, 9); int setrangeA\_i;

byte setrangeA\_crc = 0;

for

(setrangeA\_i

=

1;

setrangeA\_i

<

8;

setrangeA\_i++)

setrangeA\_crc+=setrangeA\_response[setrangeA\_i];

setrangeA\_crc = 255 - setrangeA\_crc;

setrangeA\_crc += 1;

if ( !(setrangeA\_response[0] == 0xFF && setrangeA\_response[1] == 0x99 &&

setrangeA\_response[8] == setrangeA\_crc) ) {

Serial.println("Range CRC error: " + String(setrangeA\_crc) + " / "+ String(setrangeA\_response[8]) + " (bytes 6 and 7)");

} else {

Serial.println("Range was set! (bytes 6 and 7)");

}

delay(10000);

}

// *бесконечный цикл считывания показаний датчика*

void loop()

{

digitalWrite(GatePin,LOW); delay(5000); // задержка 5 секунд swSerial.begin(9600);

byte measure\_cmd[9] = {0xFF,0x01,0x86,0x00,0x00,0x00,0x00,0x00,0x79}; *//*

*байты запроса на измерение концентрации СО2* unsigned char measure\_response[9]; swSerial.write(measure\_cmd,9); swSerial.readBytes(measure\_response, 9); int i;

byte crc = 0;

for (i = 1; i < 8; i++) crc+=measure\_response[i]; // *вычисление контрольной суммы*

crc = 255 - crc;

crc += 1;

if ( !(measure\_response[0] == 0xFF && measure\_response[1] == 0x86 &&

measure\_response[8] == crc) ) {

Serial.println("CRC error: " + String(crc) + " / "+ String(measure\_response[8]));

}

unsigned int responseHigh = (unsigned int) measure\_response[2]; unsigned int responseLow = (unsigned int) measure\_response[3]; unsigned int uart\_ppm = (256\*responseHigh) + responseLow; Serial.println("CO2 = "+ String(uart\_ppm/10000.0)+ " %"); swSerial.end();

*// вывод данных на LCD дисплей*

lcd.clear(); lcd.setCursor(0, 0); lcd.print("CO2 ");

lcd.print(String(uart\_ppm/10000.0));

lcd.print(" %");

// *управление клапаном подачи СО2 в камеру биоинкубатора (длительность открытия клапана подбирается в зависимости от текущего диапазона концентрации СО2 в камере)*

if (k==1)

{

if (uart\_ppm<=10000)

{

digitalWrite(GatePin,HIGH); delay(5000); digitalWrite(GatePin,LOW);

}

else

if (uart\_ppm<=20000)

{

digitalWrite(GatePin,HIGH); delay(3500); digitalWrite(GatePin,LOW);

}

else

if (uart\_ppm<=30000)

{

digitalWrite(GatePin,HIGH); delay(2500); digitalWrite(GatePin,LOW);

}

else

if (uart\_ppm<=40000)

{

digitalWrite(GatePin,HIGH);

delay(2000);

digitalWrite(GatePin,LOW);

}

else

if (uart\_ppm<=49800)

{

digitalWrite(GatePin,HIGH); delay(1000); digitalWrite(GatePin,LOW);

}

}

digitalWrite(GatePin,LOW);

k=1;

delay(40000);

}