

Содержимое файла "Simvols.h"

```
0001 #ifndef Simvols_h
0002 #define Simvols_h
0003 #include <Arduino.h>
0004 // описываем букву "б"
0005 byte simvol_b[8] = {
0006 0b000000,
0007 0b000000,
0008 0b11110,
0009 0b10000,
0010 0b11110,
0011 0b10001,
0012 0b11110,
0013 0b000000
0014 };
0015 // описываем букву "в"
0016 byte simvol_w[8] = {
0017 0b000000,
0018 0b000000,
0019 0b11100,
0020 0b10010,
0021 0b11100,
0022 0b10010,
0023 0b11100,
0024 0b000000
0025 };
0026 // описываем букву "д"
0027 byte simvol_d[8] = {
0028 0b000000,
0029 0b000000,
0030 0b00110,
0031 0b01010,
0032 0b01010,
0033 0b11111,
0034 0b10001,
0035 0b000000
0036 };
0037 // описываем букву "ж"
0038 byte simvol_zh[8] = {
0039 0b000000,
0040 0b000000,
0041 0b10101,
0042 0b01110,
0043 0b00100,
0044 0b01110,
0045 0b10101,
0046 0b000000
0047 };
0048 // описываем букву "з"
```

```

0049 byte simvol_z[8] = {
0050 0b000000,
0051 0b000000,
0052 0b01110,
0053 0b000001,
0054 0b00110,
0055 0b000001,
0056 0b01110,
0057 0b000000
0058 };
0059 // описываем букву "и"
0060 byte simvol_i[8] = {
0061 0b000000,
0062 0b000000,
0063 0b10001,
0064 0b10011,
0065 0b10101,
0066 0b11001,
0067 0b10001,
0068 0b000000
0069 };
0070 // описываем букву "й"
0071 byte simvol_iy[8] = {
0072 0b01010,
0073 0b00100,
0074 0b10001,
0075 0b10011,
0076 0b10101,
0077 0b11001,
0078 0b10001,
0079 0b000000
0080 };
0081 // описываем букву "к"
0082 byte simvol_k[8] = {
0083 0b000000,
0084 0b000000,
0085 0b10010,
0086 0b10100,
0087 0b11000,
0088 0b10100,
0089 0b10010,
0090 0b000000
0091 };
0092 // описываем букву "л"
0093 byte simvol_l[8] = {
0094 0b000000,
0095 0b000000,
0096 0b00111,
0097 0b01001,
0098 0b01001,
0099 0b01001,
0100 0b10001,

```

```

0101 0b000000
0102 };
0103 // описываем букву "м"
0104 byte simvol_m[8] = {
0105 0b000000,
0106 0b000000,
0107 0b10001,
0108 0b11011,
0109 0b10101,
0110 0b10001,
0111 0b10001,
0112 0b000000
0113 };
0114 // описываем букву "н"
0115 byte simvol_n[8] = {
0116 0b000000,
0117 0b000000,
0118 0b10001,
0119 0b10001,
0120 0b11111,
0121 0b10001,
0122 0b10001,
0123 0b000000
0124 };
0125 // описываем букву "п"
0126 byte simvol_p[8] = {
0127 0b000000,
0128 0b000000,
0129 0b11111,
0130 0b10001,
0131 0b10001,
0132 0b10001,
0133 0b10001,
0134 0b000000
0135 };
0136 // описываем букву "т"
0137 byte simvol_t[8] = {
0138 0b000000,
0139 0b000000,
0140 0b11111,
0141 0b00100,
0142 0b00100,
0143 0b00100,
0144 0b00100,
0145 0b000000
0146 };
0147 // описываем букву "у"
0148 byte simvol_u[8] = {
0149 0b000000,
0150 0b000000,
0151 0b01001,
0152 0b01001,

```

```

0153 0b00111,
0154 0b00001,
0155 0b01111,
0156 0b00000
0157 };
0158 // описываем букву "ц"
0159 byte simvol_c[8] = {
0160 0b00000,
0161 0b00000,
0162 0b10010,
0163 0b10010,
0164 0b10010,
0165 0b11111,
0166 0b00001,
0167 0b00000
0168 };
0169 // описываем букву "ч"
0170 byte simvol_ch[8] = {
0171 0b00000,
0172 0b00000,
0173 0b01001,
0174 0b01001,
0175 0b00111,
0176 0b00001,
0177 0b00001,
0178 0b00000
0179 };
0180 // описываем букву "ы"
0181 byte simvol_bl[8] = {
0182 0b00000,
0183 0b00000,
0184 0b10001,
0185 0b10001,
0186 0b11101,
0187 0b10101,
0188 0b11101,
0189 0b00000
0190 };
0191 // описываем букву "ь"
0192 byte simvol_m_zn[8] = {
0193 0b00000,
0194 0b00000,
0195 0b10000,
0196 0b10000,
0197 0b11100,
0198 0b10010,
0199 0b11100,
0200 0b00000
0201 };
0202 //описываем букву "щ"
0203 byte simvol_big_ch[8] = {
0204 0b01001,

```

```
0205 0b01001,
0206 0b01001,
0207 0b00111,
0208 0b00001,
0209 0b00001,
0210 0b00001,
0211 0b00000
0212 };
0213 //описываем букву "д"
0214 byte simvol_big_d[8] = {
0215 0b00100,
0216 0b01010,
0217 0b01010,
0218 0b01010,
0219 0b11111,
0220 0b10001,
0221 0b10001,
0222 0b00000
0223 };
0224 // описываем букву "ю"
0225 byte simvol_yu[8] = {
0226 0b00000,
0227 0b00000,
0228 0b10010,
0229 0b10101,
0230 0b11101,
0231 0b10101,
0232 0b10010,
0233 0b00000
0234 };
0235 // описываем букву "я"
0236 byte simvol_ya[8] = {
0237 0b00000,
0238 0b00000,
0239 0b00111,
0240 0b01001,
0241 0b00111,
0242 0b00101,
0243 0b01001,
0244 0b00000
0245 };
0246 // описываем букву "Я"
0247 byte simvol_big_ya[8] = {
0248 0b00111,
0249 0b01001,
0250 0b01001,
0251 0b00111,
0252 0b01001,
0253 0b01001,
0254 0b10001,
0255 0b00000
0256 };
```

```
0257 #endif
```

Содержимое файла "coursework.ino"

```
0001 #include <Thread.h> // Подключаем библиотеку для работы с потоками
0002 #include <Wire.h> // Подключаем библиотеку для работы с шиной I2C
0003 //библиотека для управления жидкокристаллическим дисплеем по протоколу I2C
0004 #include <LiquidCrystal_I2C.h> // Подключаем библиотеку IRremote
0005 #include <Simvols.h> // Подключаем библиотеку символов
0006 #include <IRremote.h> //библиотека для работы с инфракрасным приемником
0007 //библиотека для работы с датчиком температуры и влажности DHT22
0008 #include <DHT22.h>
0009 #define DHT22_PIN 2 //определение входа 2 для датчика DHT22
0010 #define ledPin1 6 //определение входа 6 для светодиода
0011 #define ledPin2 5 //определение входа 5 для светодиода
0012 #define ledPin3 4 //определение входа 4 для светодиода
0013 byte seconds = 0; //секунды
0014 byte minutes = 59; //минуты
0015 byte hours = 9; //часы
0016 unsigned long time; //время пройденное от запуска устройства
0017 byte c1, c2, c3;
0018 byte index;
0019 Thread threadTime = Thread(); // Создаем объект потока threadTime для
управления временем
0020 Thread threadUpdates = Thread(); // Создаем объект потока threadUpdates
для обновлений
0021 Thread threadLed1 = Thread(); // Создаем объект потока threadLed1 для
управления светодиодом
0022 Thread threadLed2 = Thread(); // Создаем объект потока threadLed1 для
управления светодиодом
0023 Thread threadLed3 = Thread(); // Создаем объект потока threadLed1 для
управления светодиодом
0024 // Создаем объект lcd для управления жидкокристаллическим дисплеем.
0025 LiquidCrystal_I2C lcd(0x27,20,4);
0026 const int RECV_PIN = 7; // Указываем, к какому входу подключен ИК-приемник
```

```

0027 // Создаем объект irrecv для приема сигналов с ИК-приемника, подключенного
к выходу RECV_PIN
0028 IRrecv irrecv(RECV_PIN);
0029 // Создаем объект results для хранения расшифрованных данных ИК-сигналов
0030 decode_results results;
0031 int flag = 0;
0032 bool flagChange = true;
0033 bool flagEn = true;
0034 int sensorValue = 20;
0035 float temperature = 23.1;
0036 float humidity = 54.5;
0037 char buffer[10];
0038 int rainPin = A0;
0039 // Создаем объект myDHT22 для работы с датчиком температуры и влажности
DHT22
0040 DHT22 myDHT22(DHT22_PIN);
0041 byte inputMin = 0;
0042 byte inputHour = 0;
0043 byte valueOfOk = 0;
0044 byte valueOfNum = 0;
0045 byte lowTemperature = 19; //выставление начальной нижней границы для
температуры
0046 byte highTemperature = 30; //выставление начальной нижней границы для
температуры
0047 byte lowHumidite = 40; //выставление начальной нижней границы для
температуры
0048 byte highHumidite = 60; //выставление начальной нижней границы для
температуры
0049 byte lowSoilMoisture = 10; //выставление начальной нижней границы для
температуры
0050 byte highSoilMoisture = 50; //выставление начальной нижней границы для
температуры
0051 void ledPin1Updates(){
0052     static bool ledStatus = false; //состояние светодиода
0053     ledStatus = !ledStatus; //инвертируем значение

```

```

0054    digitalWrite(ledPin1, ledStatus); //установка состояния светодиода
0055 }
0056 void ledPin2Updates(){
0057     static bool ledStatus = false;
0058     ledStatus = !ledStatus;
0059     digitalWrite(ledPin2, ledStatus);
0060 }
0061 void ledPin3Updates(){
0062     static bool ledStatus = false;
0063     ledStatus = !ledStatus;
0064     digitalWrite(ledPin3, ledStatus);
0065 }
0066 void dataUpdates(){
0067     // Читаем данные с датчика температуры и влажности DHT22 и обновляем
    значения в объекте myDHT22
0068     myDHT22.readData();
0069     // Считываем аналоговое значение с датчика влажности почвы
0070     sensorValue = analogRead(rainPin);
0071     // Преобразуем аналоговое значение
0072     sensorValue = (int)((1 - (float)((sensorValue - 10.0)/1022.0)) * 100);
0073     temperature = myDHT22.getTemperatureC(); // Получаем значение температу
0074     humidity = myDHT22.getHumidity(); / Получаем значение влажности
0075 }
0076 //функция подсчета времени
0077 void timeUploads(){
0078     if (millis() - time > 1000) {
0079         time = millis();
0080         seconds++;
0081         if (seconds > 59) { seconds = 0; minutes++; }
0082         if (minutes > 59) { minutes = 0; hours++; }
0083         if (hours > 23) { hours = 0; }
0084         if (hours < 10) { c1 = 8; }

```



```

0086         if (hours >= 10) { c1 = 7; }
0087         if (minutes < 10) { c2 = 11; }
0088         if (minutes >= 10) { c2 = 10; }
0089     }
0090 }
0091 void setup()
0092 {
0093     lcd.init(); // Инициализируем LCD-дисплей
0094     lcd.backlight(); // Включаем подсветку LCD-дисплея
0095     lcd.clear(); // Очищаем содержимое LCD-дисплея
0096     irrecv.enableIRIn(); // Включаем приемник ИК-сигналов
0097     irrecv.blink13(true); // Включаем мигание светодиода на приемнике ИК-
сигналов
0098     pinMode(rainPin, INPUT); // Устанавливаем режим пина rainPin как входной
0099     pinMode(ledPin1, OUTPUT); // Устанавливаем режим пина ledPin1 как
выходной
0100     pinMode(ledPin2, OUTPUT); // Устанавливаем режим пина ledPin2 как
выходной
0101     pinMode(ledPin3, OUTPUT); // Устанавливаем режим пина ledPin3 как
выходной
0102     time = millis(); // Сохраняем текущее время в переменной time
0103     // Назначаем функцию timeUploads для выполнения в потоке threadTime
0104     threadTime.onRun(timeUploads);
0105 // Устанавливаем интервал выполнения потока threadTime на 100 миллисекунд
0106     threadTime.setInterval(100);
0107     // Читаем данные с датчика температуры и влажности DHT22 и обновляем
значения
0108     myDHT22.readData();
0109 // Считываем аналоговое значение с датчика влажности почвы
0110     sensorValue = analogRead(rainPin);
0111     // Преобразуем аналоговое значение с датчика
0112     sensorValue = (int)((1 - (float)((sensorValue - 10.0)/1022.0)) * 100);
0113     // Получаем значение температуры в градусах Цельсия
0114     temperature = myDHT22.getTemperatureC();

```

```

0115    humidity = myDHT22.getHumidity(); // Получаем значение влажности
0116    // Назначаем функцию dataUpdates для выполнения в потоке threadUpdates
0117    threadUpdates.onRun(dataUpdates);
0118    // Устанавливаем интервал выполнения потока threadUpdates на 2000
миллисекунд
0119    threadUpdates.setInterval(2000);
0120    // Назначаем функцию ledPin1Updates для выполнения в потоке threadLed1
0121    threadLed1.onRun(ledPin1Updates);
0122    // Устанавливаем интервал выполнения потока threadLed1 на 500 миллисекунд
0123    threadLed1.setInterval(500);
0124    // Назначаем функцию ledPin2Updates для выполнения в потоке threadLed2
0125    threadLed2.onRun(ledPin2Updates);
0126    // Устанавливаем интервал выполнения потока threadLed2 на 500 миллисекунд
0127    threadLed2.setInterval(500);
0128    // Назначаем функцию ledPin3Updates для выполнения в потоке threadLed3
0129    threadLed3.onRun(ledPin3Updates);
0130    // Устанавливаем интервал выполнения потока threadLed3 на 500 миллисекунд
0131    threadLed3.setInterval(500);
0132    }
0133    //вывод на экран надписи "Soil moisture"
0134    void showSoilMoistureWindowEn(){
0135        lcd.setCursor(1,1);
0136        lcd.print("Soil moisture");
0137    }
0138    //вывод на экран надписи "Влажность земли"
0139    void showSoilMoistureWindowRu(){
0140        lcd.createChar(0, simvol_m);
0141        lcd.createChar(2, simvol_t);
0142        lcd.createChar(3, simvol_l);
0143        lcd.createChar(4, simvol_zh);
0144        lcd.createChar(5, simvol_n);
0145        lcd.createChar(6, simvol_m_zn);

```

```

0146     lcd.createChar(7,simvol_z);
0147     lcd.createChar(1,simvol_i);
0148     lcd.setCursor(1,1);
0149     lcd.print("B");
0150     lcd.write(3);
0151     lcd.print("a");
0152     lcd.write(4);
0153     lcd.write(5);
0154     lcd.print("oc");
0155     lcd.write(2);
0156     lcd.write(6);
0157     lcd.print(" ");
0158     lcd.write(7);
0159     lcd.print("e");
0160     lcd.write(0);
0161     lcd.write(3);
0162     lcd.write(1);
0163 }
0164 //вывод на экран надписи "Humidity"
0165 void showHumidityWindowEn(){
0166     lcd.setCursor(1,1);
0167     lcd.print("Humidity");
0168 }
0169 //вывод на экран надписи "Влажность"
0170 void showHumidityWindowRu(){
0171     lcd.createChar(3,simvol_l);
0172     lcd.createChar(4,simvol_zh);
0173     lcd.createChar(5, simvol_n);
0174     lcd.createChar(6,simvol_m_zn);
0175     lcd.setCursor(1,1);
0176     lcd.print("B");
0177     lcd.write(3);

```

```

0178     lcd.print("a");
0179     lcd.write(4);
0180     lcd.write(5);
0181     lcd.print("oc");
0182     lcd.write(2);
0183     lcd.write(6);
0184 }
0185 //вывод на экран надписи "Temperature"
0186 void showTemperatureWindowEn(){
0187     lcd.setCursor(1,1);
0188     lcd.print("Temperature");
0189 }
0190 //вывод на экран надписи "Температура"
0191 void showTemperatureWindowRu(){
0192     lcd.createChar(0, simvol_m);
0193     lcd.createChar(1, simvol_p);
0194     lcd.createChar(2, simvol_t);
0195     lcd.setCursor(1,1);
0196     lcd.print("Te");
0197     lcd.write(0);
0198     lcd.write(1);
0199     lcd.print("epa");
0200     lcd.write(2);
0201     lcd.print("ypa");
0202     lcd.print("  45.2");
0203     lcd.write(223);
0204 }
0205 //вывод на экран меню выбора языка
0206 void showInputLanguageMenuEn(){
0207     lcd.setCursor(5,0);
0208     lcd.print("Menu:");
0209     lcd.createChar(0,simvol_k);

```

```

0210     lcd.createChar(1,simvol_i);
0211     lcd.createChar(2,simvol_iy);
0212     lcd.setCursor(1,1);
0213     lcd.print("Pycc");
0214     lcd.write(0);
0215     lcd.write(1);
0216     lcd.write(2);
0217     lcd.setCursor(1,2);
0218     lcd.print("English");
0219 }
0220 //вывод на экран окна выбора изменения диапазона на английском
0221 void showInputDiapazonMenuEn(){
0222     lcd.setCursor(5,0);
0223     lcd.print("Menu:");
0224     lcd.setCursor(1,1);
0225     lcd.print("Temperature");
0226     lcd.setCursor(1,2);
0227     lcd.print("Humidity");
0228     lcd.setCursor(1,3);
0229     lcd.print("Soil moisture");
0230 }
0231 void showErrorWindowEn(){
0232     lcd.setCursor(6,1);
0233     lcd.print("Invalid");
0234     lcd.setCursor(7,2);
0235     lcd.print("value!");
0236 }
0237 //вывод на экран главного меню
0228 void showMainWindowEn(){
0239     lcd.setCursor(1,1);
0240     lcd.print("Temperature");
0241     lcd.setCursor(1,2);

```

```

0242     lcd.print("Humidity");
0243     lcd.setCursor(1,3);
0244     lcd.print("Soil moisture");
0245 }
0246 //вывод на экран меню настроек на английском языке
0247 void showSettingWindowEn(){
0248     lcd.setCursor(5,0);
0249     lcd.print("Menu:");
0250     lcd.setCursor(1,1);
0251     lcd.print("Time");
0252     lcd.setCursor(1,2);
0253     lcd.print("Range");
0254     lcd.setCursor(1,3);
0255     lcd.print("Language");
0256 }
0257 //вывод на экран главного окна на русском языке
0258 void showMainWindowRu(){
0259     //Температура
0260     lcd.createChar(0, simvol_m);
0261     lcd.createChar(1, simvol_p);
0262     lcd.createChar(2, simvol_t);
0263     lcd.setCursor(1,1);
0264     lcd.print("Te");
0265     lcd.write(0);
0266     lcd.print("nepa");
0267     lcd.write(2);
0268     lcd.print("ypa");
0269     //Влажность
0270     lcd.createChar(3, simvol_l);
0271     lcd.createChar(4, simvol_zh);
0272     lcd.createChar(5, simvol_n);
0273     lcd.createChar(6, simvol_m_zn);

```

```
0274     lcd.setCursor(1,2);
0275     lcd.print("B");
0276     lcd.write(3);
0277     lcd.print("a");
0278     lcd.write(4);
0279     lcd.write(5);
0280     lcd.print("oc");
0281     lcd.write(2);
0282     lcd.write(6);
0283     //Влажность почвы
0284     lcd.createChar(7,simvol_z);
0285     lcd.createChar(1,simvol_i);
0286     lcd.setCursor(1,3);
0287     lcd.print("B");
0288     lcd.write(3);
0289     lcd.print("a");
0290     lcd.write(4);
0291     lcd.write(5);
0292     lcd.print("oc");
0293     lcd.write(2);
0294     lcd.write(6);
0295     lcd.print(" ");
0296     lcd.write(7);
0297     lcd.print("e");
0298     lcd.write(0);
0299     lcd.write(3);
0300     lcd.write(1);
0301 }
0302 void showSettingWindowRu(){
0303     //Часы
0304     lcd.createChar(0,simvol_big_ch);
0305     lcd.createChar(1,simvol_bl);
```

```

0306     lcd.setCursor(1,1);
0307     lcd.write(0);
0308     lcd.print("ac");
0309     lcd.write(1);
0310     //Диапазон
0311     lcd.createChar(2,simvol_big_d);
0312     lcd.createChar(3,simvol_i);
0313     lcd.createChar(4,simvol_z);
0314     lcd.createChar(5,simvol_n);
0315     lcd.setCursor(1,2);
0316     lcd.write(2);
0317     lcd.write(3);
0318     lcd.print("ana");
0319     lcd.write(4);
0320     lcd.print("o");
0321     lcd.write(5);
0322     //Язык
0323     lcd.createChar(6,simvol_big_ya);
0324     lcd.createChar(7,simvol_k);
0325     lcd.setCursor(1,3);
0326     lcd.write(6);
0327     lcd.write(4);
0328     lcd.write(1);
0329     lcd.write(7);
0330 }
0331 // вывод на экран окна ввода минут на английском языке
0332 void showInputTimeMinWindowEn(){
0333     lcd.setCursor(0,1);
0334     lcd.print("Input minutes:");
0335 }
0336 // вывод на экран окна ввода часов на английском языке
0337 void showInputTimeHourWindowEn(){

```



```

0338     lcd.setCursor(0,1);
0339     lcd.print("Input hours:");
0340 }
0341 // вывод на экран окна ввода минут на русском языке
0342 void showInputTimeMinWindowRu(){
0343     //Введите минуты
0344     lcd.createChar(0,simvol_d);
0345     lcd.createChar(1,simvol_i);
0346     lcd.createChar(2,simvol_t);
0347     lcd.createChar(3,simvol_w);
0348     lcd.createChar(4,simvol_m);
0349     lcd.createChar(5,simvol_n);
0350     lcd.createChar(6,simvol_bl);
0351     lcd.setCursor(0,1);
0352     lcd.print("B");
0353     lcd.write(3);
0354     lcd.print("e");
0355     lcd.write(0);
0356     lcd.write(1);
0357     lcd.write(2);
0358     lcd.print("e ");
0359     lcd.write(4);
0360     lcd.write(1);
0361     lcd.write(5);
0362     lcd.print("y");
0363     lcd.write(2);
0364     lcd.write(6);
0365     lcd.print(":");
0366 }
0367 // вывод на экран окна ввода часов на русском языке
0368 void showInputTimeHourWindowRu(){
0369     //Введите часы

```

```

0370     lcd.createChar(0,simvol_d);
0371     lcd.createChar(1,simvol_i);
0372     lcd.createChar(2,simvol_t);
0373     lcd.createChar(3,simvol_w);
0374     lcd.createChar(4,simvol_ch);
0375     lcd.createChar(5,simvol_bl);
0376     lcd.setCursor(0,1);
0377     lcd.print("B");
0378     lcd.write(3);
0379     lcd.print("e");
0380     lcd.write(0);
0381     lcd.write(1);
0382     lcd.write(2);
0383     lcd.print("e ");
0384     lcd.write(4);
0385     lcd.print("ac");
0386     lcd.write(5);
0387     lcd.print(": ");
0388 }
0389 void showErrorWindowRu(){
0390     //Неверное число
0391     lcd.createChar(0,simvol_w);
0392     lcd.createChar(1,simvol_n);
0393     lcd.createChar(2,simvol_ch);
0394     lcd.createChar(3,simvol_i);
0395     lcd.createChar(4,simvol_l);
0396     lcd.setCursor(6,1);
0397     lcd.print("He");
0398     lcd.write(0);
0399     lcd.print("ep");
0400     lcd.write(1);
0401     lcd.print("oe");

```

```

0402     lcd.setCursor(7,2);
0403     lcd.write(2);
0404     lcd.write(3);
0405     lcd.print("c");
0406     lcd.write(4);
0407     lcd.print("o!");
0408 }
0409 //окно выбора изменения одного из параметров диапазона на русском языке
0410 void showInputDiapazonMenuRu(){
0411     //Температура
0412     lcd.createChar(0, simvol_m);
0413     lcd.createChar(1, simvol_p);
0414     lcd.createChar(2, simvol_t);
0415     lcd.setCursor(1,1);
0416     lcd.print("Te");
0417     lcd.write(0);
0418     lcd.print("непа");
0419     lcd.write(2);
0420     lcd.print("ура");
0421     //Влажность
0422     lcd.createChar(3, simvol_l);
0423     lcd.createChar(4, simvol_zh);
0424     lcd.createChar(5, simvol_n);
0425     lcd.createChar(6, simvol_m_zn);
0426     lcd.setCursor(1,2);
0427     lcd.print("В");
0428     lcd.write(3);
0429     lcd.print("а");
0430     lcd.write(4);
0431     lcd.write(5);
0432     lcd.print("оч");
0433     lcd.write(2);

```

```

0434     lcd.write(6);
0435     //Влажность земли
0436     lcd.createChar(7,simvol_z);
0437     lcd.createChar(1,simvol_i);
0438     lcd.setCursor(1,3);
0439     lcd.print("B");
0440     lcd.write(3);
0441     lcd.print("a");
0442     lcd.write(4);
0443     lcd.write(5);
0444     lcd.print("oc");
0445     lcd.write(2);
0446     lcd.write(6);
0447     lcd.print(" ");
0448     lcd.write(7);
0449     lcd.print("e");
0450     lcd.write(0);
0451     lcd.write(3);
0452     lcd.write(1);
0453 }
0454 //вывод окна установки верхней границы диапазона на английском языке
0455 void showInputDiapazonLowNumberEn(){
0456     lcd.setCursor(0,1);
0457     lcd.print("Enter the lower");
0458     lcd.setCursor(0,2);
0459     lcd.print("range: ");
0460 }
0461 //вывод окна установки верхней границы диапазона на русском языке
0462 void showInputDiapazonLowNumberRu(){
0463     //Введите нижний предел
0464     lcd.createChar(0,simvol_w);
0465     lcd.createChar(1,simvol_d);

```

```

0466     lcd.createChar(2,simvol_i);
0467     lcd.createChar(3,simvol_t);
0468     lcd.createChar(4,simvol_n);
0469     lcd.createChar(5,simvol_zh);
0470     lcd.createChar(6,simvol_iy);
0471     lcd.createChar(7,simvol_l);
0472     lcd.setCursor(0,1);
0473     lcd.print("B");
0474     lcd.write(0);
0475     lcd.print("e");
0476     lcd.write(1);
0477     lcd.write(2);
0478     lcd.write(3);
0479     lcd.print("e ");
0480     lcd.write(4);
0481     lcd.write(2);
0482     lcd.write(5);
0483     lcd.write(4);
0484     lcd.write(2);
0485     lcd.write(6);
0486     lcd.setCursor(0,2);
0487     lcd.print(" npe");
0488     lcd.write(1);
0489     lcd.print("e");
0490     lcd.write(7);
0491     lcd.print(":");
0492 }
0493 void showInputDiapazonHighNumberEn(){
0494     lcd.setCursor(0,1);
0495     lcd.print("Enter the upper");
0496     lcd.setCursor(0,2);
0497     lcd.print("range: ");

```

```

0498  }
0499  void showInputDiapazonHighNumberRu(){
0500  //Введите верхний предел
0501      lcd.createChar(0,simvol_w);
0502      lcd.createChar(1,simvol_d);
0503      lcd.createChar(2,simvol_i);
0504      lcd.createChar(3,simvol_t);
0505      lcd.createChar(4,simvol_n);
0506      lcd.createChar(5,simvol_zh);
0507      lcd.createChar(6,simvol_iy);
0508      lcd.createChar(7,simvol_l);
0509      lcd.setCursor(0,1);
0510      lcd.print("B");
0511      lcd.write(0);
0512      lcd.print("e");
0513      lcd.write(1);
0514      lcd.write(2);
0515      lcd.write(3);
0516      lcd.print("e ");
0517      lcd.write(0);
0518      lcd.print("epx");
0519      lcd.write(4);
0520      lcd.write(2);
0521      lcd.write(6);
0522      lcd.setCursor(0,2);
0523      lcd.print(" npe");
0524      lcd.write(1);
0525      lcd.print("e");
0526      lcd.write(7);
0527      lcd.print(":");
0528  }
0529  void showInputLanguageMenuRu(){

```

```

0530    //Русский
0531    lcd.createChar(0,simvol_k);
0532    lcd.createChar(1,simvol_i);
0533    lcd.createChar(2,simvol_iy);
0534    lcd.createChar(3,simvol_n);
0535    lcd.createChar(4,simvol_yu);
0536    lcd.setCursor(5,0);
0537    lcd.print("Me");
0538    lcd.write(3);
0539    lcd.write(4);
0540    lcd.print(":");
0541    lcd.setCursor(1,1);
0542    lcd.print("Pycc");
0543    lcd.write(0);
0544    lcd.write(1);
0545    lcd.write(2);
0546    lcd.setCursor(1,2);
0547    lcd.print("English");
0548 }
0549 void loop()
0550 {
0551   if(threadTime.shouldRun())
0552     threadTime.run();
0553   if(threadUpdates.shouldRun())
0554     threadUpdates.run();
0555   if (!(temperature <= highTemperature && temperature >= lowTemperature))
0556   {
0557     if(threadLed1.shouldRun())
0558       threadLed1.run();
0559   }
0560   else
0561   {

```

```

0562     digitalWrite(ledPin1, false);
0563 }
0564 if (!(humidity <= highHumidite && humidity >= lowHumidite))
0565 {
0566     if(threadLed2.shouldRun())
0567         threadLed2.run();
0568 }
0569 else
0570 {
0571     digitalWrite(ledPin2, false);
0572 }
0573 if (!(sensorValue <=highSoilMoisture && sensorValue >= lowSoilMoisture))
0574 {
0575     if(threadLed3.shouldRun())
0576         threadLed3.run();
0577 }
0578 else
0579 {
0580     digitalWrite(ledPin3, false);
0582 0581     }
0583     switch(flag)
0584     {
0585         case 0:
0586             if(flagChange){
0587                 lcd.clear();
0588                 if(flagEn)
0589                     showMainWindowEn();
0590             else
0591                 showMainWindowRu();
0592             flagChange = false;
0593         }
0594         if(flagEn){

```



```

0595         lcd.setCursor(14,1);
0596         dtostrf(temperature, 4, 1, buffer);
0597         lcd.print(buffer);
0598         lcd.write(223);
0599         lcd.setCursor(14,2);
0600         dtostrf(humidity, 4, 1, buffer);
0602 0601         lcd.print(buffer);
0603         lcd.write(37);
0604         lcd.setCursor(16,3);
0605         lcd.print(sensorValue);
0606         lcd.write(37);
0607     }
0608     else{
0609         lcd.setCursor(15,1);
0610         dtostrf(temperature, 4, 1, buffer);
0611         lcd.print(buffer);
2         lcd.write(223);
0613         lcd.setCursor(15,2);
0614         dtostrf(humidity, 4, 1, buffer);
0615         lcd.print(buffer);
0616         lcd.write(37);
0617         lcd.setCursor(17,3);
0618         lcd.print(sensorValue);
0619         lcd.write(37);
0620     }
0621     lcd.setCursor(7, 0);
0622     lcd.print("0");
0623     lcd.setCursor(c1, 0);
0624     lcd.print(hours);
0625     lcd.print(":");
0626     lcd.setCursor(10, 0);
0627     lcd.print("0");

```

```

0628     lcd.setCursor(c2, 0);
0629     lcd.print(minutes);
0630     break;
0631     case 1:
0632     if(flagChange){
0633         lcd.clear();
0634         if(flagEn)
0635             showTemperatureWindowEn();
0636         else
0637             showTemperatureWindowRu();
0638         flagChange = false;
0639     }
0640     lcd.setCursor(14,1);
0641     dtostrf(temperature, 4, 1, buffer);
0642     lcd.print(buffer);
0643     lcd.write(223);
0644     break;
0645     case 2:
0646     if(flagChange){
0647         lcd.clear();
0648         if(flagEn)
0649             showHumidityWindowEn();
0650         else
0651             showHumidityWindowRu();
0652         flagChange = false;
0653     }
0654     lcd.setCursor(14,1);
0655     dtostrf(humidity, 4, 1, buffer);
0656     lcd.print(buffer);
0657     lcd.write(37);
0658     break;
0659     case 3:

```

```

0660     if(flagChange){
0661         lcd.clear();
0662         if(flagEn)
0663             showSoilMoistureWindowEn();
0664         else
0665             showSoilMoistureWindowRu();
0666         flagChange = false;
0667     }
0668     lcd.setCursor(17,1);
0669     lcd.print(sensorValue);
0670     lcd.write(37);
0671     break;
0672     case 4:
0673     if(flagChange){
0674         lcd.clear();
0675         if(flagEn)
0676             showSettingWindowEn();
0677         else
0678             showSettingWindowRu();
0679         flagChange = false;
0680     }
0681     lcd.setCursor(0,index);
0682     lcd.write(165);
0683     break;
0684     case 5:
0685     if(flagChange){
0686         lcd.clear();
0687         if(flagEn)
0688             showInputLanguageMenuEn();
0689         else
0690             showInputLanguageMenuRu();
0691         flagChange = false;

```

```

0692     }
0693     lcd.setCursor(0,index);
0694     lcd.write(165);
0695     break;
0696     case 7:
0697         if(flagChange){
0698             lcd.clear();
0697             if(flagEn)
0698                 showInputDiapazonMenuEn();
0699             else
0700                 showInputDiapazonMenuRu();
0701             flagChange = false;
0702         }
0703     lcd.setCursor(0,index);
0704     lcd.write(165);
0705     break;
0706     case 8:
0707         if(flagChange){
0708             lcd.clear();
0709             switch(valueOfOk){
0710                 case 0:
0711                     inputHour = 0;
0712                     lcd.clear();
0713                     valueOfNum=0;
0714                     if(flagEn)
0716                         showInputDiapazonLowNumberEn();
0717                     else
0718                         showInputDiapazonLowNumberRu();
0719                     flagChange = false;
0720                     break;
0721                 case 1:
0722                     inputMin = 0;

```

```

0723         valueOfNum = 0;
0724         if(flagEn)
0725             showInputDiapazonHighNumberEn();
0726         else
0727             showInputDiapazonHighNumberRu();
0728         flagChange = false;
0729         break;
0730         case 2:
0731             if (inputMin <= 99 && inputMin>=0 && inputHour<=99 &&
inputHour>=0){
0732                 switch(index){
0733                     case 1:
0734                         lowTemperature = inputHour;
0735                         highTemperature = inputMin;
0736                         break;
0737                     case 2:
0738                         lowHumidite = inputHour;
0739                         highHumidite = inputMin;
0740                         break;
0741                     case 3:
0742                         lowSoilMoisture = inputHour;
0743                         highSoilMoisture = inputMin;
0744                         break;
0745                 }
0746                 flag = 0;
0747                 flagChange = true;
0748             }
0749         else
0750         {
0751             if(flagEn)
0752                 showErrorWindowEn();
0753             else

```

```

0754         showErrorWindowRu();
0755         delay(900);
0756         flag = 0;
0757         flagChange = true;
0758     }
0759     break;
0760 }
0761 }
0762 lcd.setCursor(16,2);
0763 if (valueOfOk == 0)
0764     lcd.print(inputHour);
0765     else
0766     lcd.print(inputMin);
0767 break;
0768
0769 case 6:
0770 if(flagChange){
0771     lcd.clear();
0772     switch(valueOfOk){
0773         case 0:
0774             lcd.clear();
0775             valueOfNum=0;
0776             if(flagEn)
0777                 showInputTimeHourWindowEn();
0778             else
0779                 showInputTimeHourWindowRu();
0780             flagChange = false;
0781             break;
0782         case 1:
0783             inputMin = 0;
0784             valueOfNum = 0;
0785             if(flagEn)

```

```

0786         showInputTimeMinWindowEn();
0787     else
0788         showInputTimeMinWindowRu();
0789     flagChange = false;
0790     break;
0791     case 2:
0792         if (inputMin <= 59 && inputMin>=0 && inputHour<=24 &&
inputHour>=0){
0793             seconds = 0;
0794             minutes = inputMin;
0795             hours = inputHour;
0796             flag = 0;
0797             flagChange = true;
0798         }
0799     else
0800     {
0801         if(flagEn)
0802             showErrorWindowEn();
0803         else
0804             showErrorWindowRu();
0805         delay(900);
0806         flag = 0;
0807         flagChange = true;
0808     }
0809     break;
0810 }
0811 }
0812 }
0813 lcd.setCursor(16,1);
0814 if (valueOfOk == 0)
0815     lcd.print(inputHour);
0816 else
0817     lcd.print(inputMin);

```

```

0818     break;
0819 }
0820 if (irrecv.decode(&results))
0821 {
0822     switch(results.value)
0823     {
0824         case 0xFFA25D:                                     // Кнопка "1"
0825             if (flag == 6 || flag == 8){
0826                 if(valueOfNum == 0)
0827                 {
0828                     if (valueOfOk == 1)
0829                         inputMin = 1;
0830                     else
0831                         inputHour = 1;
0832                     valueOfNum = 1;
0833                 }
0834             else{
0835                 if(valueOfNum == 1){
0836                     if(valueOfOk == 1)
0837                         inputMin = inputMin*10+1;
0838                     else
0839                         inputHour = inputHour*10+1;
0840                     valueOfNum = 2;
0841                 }
0842             }
0843         }
0844     else{
0845         flag = 1;
0846         flagChange = true;
0847     }
0848 }
0849     switch(results.value)

```



```

0850     {
0851         case 0xFF629D:                                     // Кнопка "2"
0852             if (flag == 6 || flag == 8){
0853                 if(valueOfNum == 0)
0854                     {
0855                         if (valueOfOk == 1)
0856                             inputMin = 2;
0857                         else
0858                             inputHour = 2;
0859                         valueOfNum = 1;
0860                     }
0861                 else{
0862                     if(valueOfNum == 1){
0863                         if (valueOfOk == 1)
0864                             inputMin = inputMin*10+2;
0865                         else
0866                             inputHour = inputHour*10+2;
0867                         valueOfNum = 2;
0868                     }
0869                 }
0870             }
0871             else{
0872                 flag = 2;
0873                 flagChange = true;
0874             }
0875         }
0876         switch(results.value)
0877         {
0878             case 0xFFE21D:                                     // Кнопка "3"
0879                 if (flag == 6 || flag == 8){
0880                     if(valueOfNum == 0)
0881                         {

```

```

0882         if (valueOfOk == 1)
0883             inputMin = 3;
0884         else
0885             inputHour = 3;
0886             valueOfNum = 1;
0887     }
0888     else{
0889         if(valueOfNum == 1){
0890             if (valueOfOk == 1)
0891                 inputMin = inputMin*10+3;
0892             else
0893                 inputHour = inputHour*10+3;
0894                 valueOfNum = 2;
0895         }
0896     }
0897 }
0898 else{
0899     flag = 3;
0900     flagChange = true;
0901 }
0902 }
0903 switch(results.value)
0904 {
0905     case 0xFF22DD:                                     // Кнопка "4"
0906         if (flag == 6 || flag == 8){
0907             if(valueOfNum == 0)
0908             {
0909                 if (valueOfOk == 1)
0910                     inputMin = 4;
0911                 else
0912                     inputHour = 4;
0913                 valueOfNum = 1;

```

```

0914         }
0915     else{
0916         if(valueOfNum == 1){
0917             if (valueOfOk == 1)
0918                 inputMin = inputMin*10+4;
0919             else
0920                 inputHour = inputHour*10+4;
0921             valueOfNum = 2;
0922         }
0923     }
0924 }
0925 }
0926 switch(results.value)
0927 {
0928     case 0xFF02FD:                                     // Кнопка "5"
0929         if (flag == 6 || flag == 8){
0930             if(valueOfNum == 0)
0931             {
0932                 if (valueOfOk == 1)
0933                     inputMin = 5;
0934                 else
0935                     inputHour = 5;
0936                 valueOfNum = 1;
0937             }
0938         else{
0939             if(valueOfNum == 1){
0940                 if (valueOfOk == 1)
0941                     inputMin = inputMin*10+5;
0942                 else
0943                     inputHour = inputHour*10+5;
0944                 valueOfNum = 2;
0945             }

```

```

0946         }
0947     }
0948 }
0949 switch(results.value)
0950 {
0951     case 0xFFC23D:                                     // Кнопка "6"
0952         if (flag == 6 || flag == 8){
0953             if(valueOfNum == 0)
0954             {
0955                 if (valueOfOk == 1)
0956                     inputMin = 6;
0957                 else
0958                     inputHour = 6;
0959                 valueOfNum = 1;
0960             }
0961         }else{
0962             if(valueOfNum == 1){
0963                 if (valueOfOk == 1)
0964                     inputMin = inputMin*10+6;
0965                 else
0966                     inputHour = inputHour*10+6;
0966                 valueOfNum = 2;
0967             }
0968         }
0969     }
0970 }
0971 switch(results.value)
0972 {
0973     case 0xFFE01F:                                     // Кнопка "7"
0974         if (flag == 6 || flag == 8){
0975             if(valueOfNum == 0)
0976             {

```

```

0977         if (valueOfOk == 1)
0978             inputMin = 7;
0979         else
0980             inputHour = 7;
0981             valueOfNum = 1;
0982     }
0983     else{
0984         if(valueOfNum == 1){
0985             if (valueOfOk == 1)
0986                 inputMin = inputMin*10+7;
0987             else
0988                 inputHour = inputHour*10+7;
0989             valueOfNum = 2;
0990         }
0991     }
0992 }
0993 }
0994 switch(results.value)
0995 {
0996     case 0xFFA857:                                     // Кнопка "8"
0997         if (flag == 6 || flag == 8){
0998             if(valueOfNum == 0)
0999                 {
1000                     if (valueOfOk == 1)
1001                         inputMin = 8;
1002                     else
1003                         inputHour = 8;
1004                     valueOfNum = 1;
1005                 }
1006             else{
1007                 if(valueOfNum == 1){
1008                     if (valueOfOk == 1)

```

```

1009         inputMin = inputMin*10+8;
1010     else
1011         inputHour = inputHour*10+8;
1012         valueOfNum = 2;
1013     }
1014 }
1015 }
1016 }
1017 switch(results.value)
1018 {
1019     case 0xFF906F:                                     // Кнопка "9"
1020         if (flag == 6 || flag == 8){
1021             if(valueOfNum == 0)
1022             {
1023                 if (valueOfOk == 1)
1024                     inputMin = 9;
1025                 else
1026                     inputHour = 9;
1027                 valueOfNum = 1;
1028             }
1029             else{
1030                 if(valueOfNum == 1){
1031                     if (valueOfOk == 1)
1032                         inputMin = inputMin*10+9;
1033                     else
1034                         inputHour = inputHour*10+9;
1035                     valueOfNum = 2;
1036                 }
1037             }
1039         }
1040     }
1041     switch(results.value)

```

```

1042     {
1043         case 0xFF9867:                                     // Кнопка "0"
1044             if (flag == 6 || flag == 8){
1045                 if(valueOfNum == 0)
1046                     {
1047                         if (valueOfOk == 1)
1048                             inputMin = 0;
1049                         else
1050                             inputHour = 0;
1051                         valueOfNum = 1;
1052                     }
1053                 else{
1054                     if(valueOfNum == 1){
1055                         if (valueOfOk == 1)
1056                             inputMin = inputMin*10+0;
1057                         else
1058                             inputHour = inputHour*10+0;
1059                         valueOfNum = 2;
1060                     }
1061                 }
1062             }
1063         else{
1064             flag = 0;
1065             flagChange = true;
1066         }
1067     }
1068     switch(results.value)
1069     {
1070         case 0xFF6897:                                     // Кнопка "*"
1071             flag = 4;
1072             index = 1;
1073             flagChange = true;

```

```

1074     }
1075     switch(results.value)
1076     {
1077         case 0xFF18E7:                                     // Кнопка "up"
1078             if (flag == 4 || flag == 7){
1079                 if(index == 1)
1080                     index = 3;
1081             else
1082                 index--;
1083         }
1084     else{
1085         if(index == 1)
1086             index = 2;
1087         else
1088             index = 1;
1089     }
1090     flagChange = true;
1091     }
1092     switch(results.value)
1093     {
1094         case 0xFF4AB5:                                     // Кнопка "down"
1095             if (flag == 4 || flag == 7){
1096                 if(index == 3)
1097                     index = 1;
1098             else
1099                 index++;
1100     }
1101     else{
1102         if (index == 2)
1103             index = 1;
1104         else
1105             index = 2;

```



```

1106     }
1107     flagChange = true;
1108 }
1109 switch(results.value)
1110 {
1111     case 0xFF38C7:                                     // Кнопка "ok"
1112 switch(flag) {
1113     case 5:
1114         if(index == 1) {
1115             flagEn = false;
1116             delay(500);
1117             flag = 0;
1118         } else {
1119             flagEn = true;
1120             delay(500);
1121             flag = 0;
1122         }
1123     break;
1124     case 4:
1125         if(index == 3) {
1126             flag = 5;
1127             index = 1;
1128         }
1129     else
1130         if(index == 1){
1131             flag = 6;
1132             valueOfNum = 0;
1133             valueOfOk = 0;
1134             inputMin = 0;
1135             inputHour = 0;
1136         }
1137     else

```

```
1137         if(index == 2){
1138             valueOfOk = 0;
1139             flag = 7;
1140         }
1141         break;
1142     case 7:
1143         flag = 8;
1144         valueOfOk = 0;
1145         break;
1146     case 8:
1147         valueOfOk++;
1148         break;
1149     case 6:
1150         valueOfOk++;
1151         break;
1152     }
1153     flagChange = true;
1154 }
1155 irrecv.resume();
1156 }
1157 }
```