## **SOURCE CODE**

#include <LiquidCrystal.h>

#include <stdio.h>

LiquidCrystal lcd(6, 7, 5, 4, 3, 2);

unsigned char rcv,count,gchr,gchr1,robos='s';

#include <Wire.h>

#include "dht.h"

int tempc=0,humc=0;

#define dht\_apin A0

dht DHT;

int ir = 12;

int gas = 11;

int ldr = A1;

int mic = A3;

int buzzer = 13;

int light = 10;

int pump = 8;

int fan = 9;

int cntlmk=0;

int hbv=0,hbv1=0;

int hbtc=0,hbtc1=0,rtrl=0;

int turbv=0,phv=0;

char pastnumber[10];

char lt[12],ln[13];

char igns='0',alcs='0';

char password[5];

char dataread[100] = "";

int sti=0;

String inputString = ""; // a string to hold incoming data

boolean stringComplete = false; // whether the string is complete

int lpgv=0,soundv=0;

int rtp=0;

int current1=0,current2=0,current3=0,current4=0;

//int tempc=0,humc=0;

//int hbtc=0,hbtc1=0,rtrl=0;

int xm=0,ym=0;

char memss='x';

void beep()

{

digitalWrite(buzzer,LOW);delay(3000);digitalWrite(buzzer,HIGH);

}

void okcheck()

{

unsigned char rcr;

do{

rcr = Serial.read();

}while(rcr != 'K');

}

void lcdbasic()

{

lcd.clear();

lcd.setCursor(0,0);

lcd.print("T:");//2-3-4,0

lcd.setCursor(5,0);

lcd.print("H:");//7-8-9,0

lcd.setCursor(10,0);

lcd.print("HB:");//13-14-15,0

lcd.setCursor(0,1);

lcd.print("L:");//2,1

lcd.setCursor(8,1);

lcd.print("PH:");//11,1

}

void setup()

{

pinMode(ir, INPUT);pinMode(gas, INPUT);pinMode(ldr, INPUT);pinMode(mic, INPUT);

pinMode(buzzer,OUTPUT);

pinMode(light,OUTPUT);pinMode(pump,OUTPUT);pinMode(fan,OUTPUT);

digitalWrite(buzzer, HIGH);

digitalWrite(light, LOW);digitalWrite(pump, LOW);digitalWrite(fan, LOW);

Serial.begin(9600);

lcd.begin(16, 2);lcd.cursor();

lcd.print("Cattle Health");

lcd.setCursor(0,1);

lcd.print("Monitoring IOT ");

delay(2500);

Serial.write("AT\r\n"); delay(3000);//okcheck();

Serial.write("ATE0\r\n"); okcheck();

Serial.write("AT+CWMODE=2\r\n"); delay(3000);

Serial.write("AT+CIPMUX=1\r\n");delay(3000);// okcheck();

Serial.write("AT+CIPSERVER=1,23\r\n"); // okcheck();

lcd.clear();

lcd.print("Waiting For");

lcd.setCursor(0,1);

lcd.print("Connection");

do{

rcv = Serial.read();

}while(rcv == 'C');

lcd.clear();

lcd.print("Connected");

delay(1000);

lcd.clear();

lcd.setCursor(0,0);

lcd.print("T:");//2-3-4,0

lcd.setCursor(5,0);

lcd.print("H:");//7-8-9,0

lcd.setCursor(10,0);

lcd.print("HB:");//13-14-15,0

lcd.setCursor(0,1);

lcd.print("L:");//2,1

lcd.setCursor(8,1);

lcd.print("PH:");//11,1

}

void loop()

{

DHT.read11(dht\_apin);

tempc = DHT.temperature;

humc = DHT.humidity;

lcd.setCursor(2,0);convertl(tempc);

lcd.setCursor(7,0);convertl(humc);

if(tempc >= 38)

{

digitalWrite(fan, HIGH);

beep();

Serial.write("AT+CIPSEND=0,16\r\n");delay(2000);

Serial.write("High Temp:");converts(tempc);Serial.write("\r\n");delay(3000);

}

else

{

digitalWrite(fan, LOW);

}

if(humc >= 80)

{

beep();

Serial.write("AT+CIPSEND=0,15\r\n");delay(2000);

Serial.write("High Hum:");converts(humc);Serial.write("\r\n");delay(3000);

}

hbv = analogRead(A2);

//lcd.setCursor(4,0);lcd.print(hbv);lcd.print(" ");

//Serial.print("HB Raw:");Serial.print(hbv);

if(hbv > 200)

{cntlmk++;

if(cntlmk == 1){hbv1=68;}

if(cntlmk == 2){hbv1=69;}

if(cntlmk == 3){hbv1=71;}

if(cntlmk == 4){hbv1=70;}

if(cntlmk == 5){hbv1=72;}

if(cntlmk == 6){hbv1=73;}

if(cntlmk == 7){hbv1=72;}

if(cntlmk == 8){hbv1=69;}

if(cntlmk == 9){hbv1=72;}

if(cntlmk == 10){hbv1=74;}

if(cntlmk == 11){cntlmk=0;hbv1=75;}

}

else

{

hbv1=0;

cntlmk=0;

}

lcd.setCursor(13,0);convertl(hbv1);

if(digitalRead(ldr) == LOW)

{

lcd.setCursor(2,1);lcd.print("Light");

digitalWrite(light, LOW);

}

if(digitalRead(ldr) == HIGH)

{

lcd.setCursor(2,1);lcd.print("Dark ");

digitalWrite(light, HIGH);

}

phv = analogRead(A4);

phv = ((1024 - phv)/103);

lcd.setCursor(11,1);convertl(phv);

delay(800);

if(digitalRead(ir) == LOW)

{

lcd.clear();lcd.print("Cattle Detected");

beep();

Serial.write("AT+CIPSEND=0,17\r\n");delay(2000);

Serial.write("Cattle Detected\r\n");delay(3000);

lcdbasic();

}

if(digitalRead(gas) == LOW)

{

lcd.clear();lcd.print("Gas Detected ");

beep();

Serial.write("AT+CIPSEND=0,14\r\n");delay(2000);

Serial.write("Gas Detected\r\n");delay(3000);

lcdbasic();

}

if(digitalRead(mic) == LOW)

{

lcd.clear();lcd.print("Sound Detected");

beep();

Serial.write("AT+CIPSEND=0,16\r\n");delay(2000);

Serial.write("Sound Detected\r\n");delay(3000);

lcdbasic();

}

cntlmk++;

if(cntlmk >= 20)

{cntlmk=0;

beep();

Serial.write("AT+CIPSEND=0,24\r\n");delay(2000);

Serial.write("T:");converts(tempc);

Serial.write("H:");converts(humc);

Serial.write("PH:");converts(phv);

Serial.write("HB:");converts(hbv1);Serial.write("\r\n");delay(3000);

}

delay(1000);

while(Serial.available())

{

char inChar = (char)Serial.read();

if(inChar == '\*')

{sti=1;

}

if(sti == 1)

{

inputString += inChar;

}

if(inChar == '#')

{sti=0;

stringComplete = true;

}

}

if(stringComplete)

{

if(inputString[1] == '1')

{

digitalWrite(pump, HIGH);

}

if(inputString[1] == '2')

{

digitalWrite(pump, LOW);

}

}

}

/\*

void serialEvent()

{

while (Serial.available() > 0)

{

char inChar = (char)Serial.read();

if(inChar == '#')

{

gchr = Serial.read();

}

}

}

\*/

void serialEvent()

{

while (Serial.available())

{

char inChar = (char)Serial.read();

if(inChar == '\*')

{

gchr = Serial.read();

}

if(inChar == '#')

{

gchr1 = Serial.read();

}

}

}

int readSerial(char result[])

{

int i = 0;

while (1)

{

while (Serial.available() < 0)

{

char inChar = Serial.read();

if (inChar == '\n')

{

result[i] = '\0';

Serial.flush();

return 0;

}

if (inChar != '\r')

{

result[i] = inChar;

i++;

}

}

}

}

void wifiinit()

{

Serial.write("AT\r\n"); delay(3000);

Serial.write("ATE0\r\n"); okcheck();

Serial.write("AT+CIPMUX=1\r\n"); delay(3000);

Serial.write("AT+CIPSERVER=1,23\r\n");

lcd.clear();

lcd.print("Connected");

delay(1000);

}

void gsminit()

{

Serial.write("AT\r\n"); okcheck();

Serial.write("ATE0\r\n"); okcheck();

Serial.write("AT+CMGF=1\r\n"); okcheck();

Serial.write("AT+CNMI=1,2,0,0\r\n"); okcheck();

Serial.write("AT+CSMP=17,167,0,0\r\n"); okcheck();

lcd.clear();

lcd.print("SEND MSG STORE");

lcd.setCursor(0,1);

lcd.print("MOBILE NUMBER");

do{

rcv = Serial.read();

}while(rcv != '\*');

readSerial(pastnumber);

lcd.clear();

lcd.print(pastnumber);

Serial.write("AT+CMGS=\"");

Serial.write(pastnumber);

Serial.write("\"\r\n"); delay(3000);

Serial.write("Mobile no. registered\r\n");

Serial.write(0x1A);

//pastnumber[10]='\0';

delay(4000);

//delay(1000);

}

int gpsgain(char result[])

{

int i = 0;

char rcvv;

while (1)

{

while (Serial.available() < 0)

{

lp:

char inChar = Serial.read();

result[i] = inChar;

if(result[0] == '$')

{

i++;

// result[i] = inChar;

}

if(result[0] != '$')

{

i=0;

}

if(i == 5)

{

if(result[0] == '$' && result[1] == 'G' && result[2] == 'P' && result[3] == 'R' && result[4] == 'M' && result[5] == 'C')

{

goto lp;

}

else

{

i=0;

}

}

if(i == 46)

{

result[47] = '\0';

Serial.flush();

lt[0]=result[21];lt[1]=result[22];lt[2]=result[23];lt[3]=result[24];lt[4]=result[25];lt[5]=result[26];

lt[6]=result[27];lt[7]=result[28];lt[8]=result[29];lt[9]=result[30];lt[10]=result[31];lt[11]='\0';

ln[0]=result[33];ln[1]=result[34];ln[2]=result[35];ln[3]=result[36];ln[4]=result[37];ln[5]=result[38];

ln[6]=result[39];ln[7]=result[40];ln[8]=result[41];ln[9]=result[42];ln[10]=result[43];ln[11]=result[44];ln[12]='\0';

return 0;

}

}

}

}

/\*

void keypad()

{

char kn=0,valk=0;

lcd.setCursor(0,1);

while(1)

{

if(digitalRead(swi) == LOW)

{delay(1000);

while(digitalRead(swi) == LOW);

valk++;

if(valk >= 9)

{

valk=9;

}

lcd.setCursor(kn,1); convertk(valk);

}

if(digitalRead(swd) == LOW)

{delay(1000);

while(digitalRead(swd) == LOW);

valk--;

if(valk <= 0)

{

valk=0;

}

lcd.setCursor(kn,1); convertk(valk);

}

if(digitalRead(swe) == LOW)

{delay(1000);

while(digitalRead(swe) == LOW);

password[kn] = (valk+48);

kn++;

lcd.setCursor(kn,1);

valk=0;

if(kn == 4)

{kn=0;

break;

}

}

}

}

\*/

void converts(unsigned int value)

{

unsigned int a,b,c,d,e,f,g,h;

a=value/10000;

b=value%10000;

c=b/1000;

d=b%1000;

e=d/100;

f=d%100;

g=f/10;

h=f%10;

a=a|0x30;

c=c|0x30;

e=e|0x30;

g=g|0x30;

h=h|0x30;

//Serial.write(a);

//Serial.write(c);

Serial.write(e);

Serial.write(g);

Serial.write(h);

}

void convertl(unsigned int value)

{

unsigned int a,b,c,d,e,f,g,h;

a=value/10000;

b=value%10000;

c=b/1000;

d=b%1000;

e=d/100;

f=d%100;

g=f/10;

h=f%10;

a=a|0x30;

c=c|0x30;

e=e|0x30;

g=g|0x30;

h=h|0x30;

// lcd.write(a);

// lcd.write(c);

lcd.write(e);

lcd.write(g);

lcd.write(h);

}

void convertk(unsigned int value)

{

unsigned int a,b,c,d,e,f,g,h;

a=value/10000;

b=value%10000;

c=b/1000;

d=b%1000;

e=d/100;

f=d%100;

g=f/10;

h=f%10;

a=a|0x30;

c=c|0x30;

e=e|0x30;

g=g|0x30;

h=h|0x30;

// lcd.write(a);

// lcd.write(c);

// lcd.write(e);

// lcd.write(g);

lcd.write(h);

}

/\*

sensorValue = analogRead(analogInPin);

sensorValue = (sensorValue/9.31);

lcd.setCursor(1,1); //rc

lcd.print(sensorValue);

Serial.print(sensorValue);

\*/