#include <Wire.h>

#include <LiquidCrystal\_I2C.h>

// Set the LCD address to 0x27 for a 16 chars and 2 line display

LiquidCrystal\_I2C lcd(0x3F, 16, 2);

#include <DS1302.h>

int Hour;

int Min;

int pset = 8; // pushbutton for setting alarm

int phour = 9; // pushbutton for hour

int pmin = 10; // pushbutton for minutes

int pexit = 11; // pushbutton for exit of set alarm

int buzzer = 6;

int h;

int m;

int buttonforset = 0; // pushbutton state for setting alarm

int buttonforhour = 0; // pushbutton state for hour

int buttonformin = 0;// pushbutton state for minutes

int buttonforexit = 0; // pushbutton state for exit of set alarm

int activate=0;

Time t;

// Motor connections

int enA = 5;

int in1 = 12;

int in2 = 13;

// Init the DS1302

DS1302 rtc(2, 3, 4);

void setup()

{

pinMode(pset, INPUT);

pinMode(phour, INPUT);

pinMode(pmin, INPUT);

pinMode(pexit, INPUT);

// Set the clock to run-mode, and disable the write protection

rtc.halt(false);

rtc.writeProtect(false);

// Set all the motor control pins to outputs

pinMode(enA, OUTPUT);

pinMode(in1, OUTPUT);

pinMode(in2, OUTPUT);

// Turn off motors - Initial state

digitalWrite(in1, LOW);

digitalWrite(in2, LOW);

// Setup LCD to 16x2 characters

lcd.begin();

// The following lines can be commented out to use the values already stored in the DS1302

//rtc.setDOW(SATURDAY); // Set Day-of-Week to FRIDAY

//rtc.setTime(10, 0, 0); // Set the time to 12:00:00 (24hr format)

//rtc.setDate(11, 11, 2017); // Set the date to August 6th, 2010

}

void loop()

{

directionControl();

delay(1000);

if (activate == 0) {

// Display time on the right conrner upper line

lcd.setCursor(0, 0);

lcd.print("Time: ");

lcd.setCursor(6, 0);

lcd.print(rtc.getTimeStr());

// Display abbreviated Day-of-Week in the lower left corner

//lcd.setCursor(0, 1);

//lcd.print(rtc.getDOWStr(FORMAT\_SHORT));

// Display date in the lower right corner

lcd.setCursor(0, 1);

lcd.print("Date: ");

lcd.setCursor(6, 1);

lcd.print(rtc.getDateStr());

t = rtc.getTime();

Hour = t.hour;

Min = t.min;

buttonforset = digitalRead(pset);

} // setting button pressed

if (buttonforset == HIGH) {

activate =1;

lcd.clear(); }

while(activate== 1){

lcd.setCursor(0,0);

lcd.print("Set Alarm");

lcd.setCursor(0,1);

lcd.print("Hour= ");

lcd.setCursor(9,1);

lcd.print("Min= ");

buttonforhour = digitalRead(phour); // set hour for alarm

if (buttonforhour == HIGH){

h++;

lcd.setCursor(5,1);

lcd.print(h);

if (h>23){

h=0;

lcd.clear(); }

delay(100);

}

buttonformin = digitalRead(pmin); // set minutes for alarm

if (buttonformin == HIGH){

m++;

lcd.setCursor(13,1);

lcd.print(m);

if (m>59){

m=0;

lcd.clear();}

delay(100);

}

lcd.setCursor(5,1);

lcd.print(h);

lcd.setCursor(13,1);

lcd.print(m);

buttonforexit = digitalRead(pexit); // exit from set alarm mode

if (buttonforexit == HIGH){

activate = 0;

lcd.clear();

}

}

if (Hour== h && Min== m) {

tone(6,400,300);

// This function lets you control spinning direction of motors

void directionControl() {

// Set motors to maximum speed

// For PWM maximum possible values are 0 to 255

analogWrite(enA, 255);

// Turn on motor

digitalWrite(in1, HIGH);

digitalWrite(in2, LOW);

delay(1800000);

// Now change motor directions

digitalWrite(in1, LOW);

digitalWrite(in2, HIGH);

delay(2000);

// Turn off motor

digitalWrite(in1, LOW);

digitalWrite(in2, LOW);

}

delay (500);

}