#include <Keypad.h>

#include <Arduino.h>

#include <LiquidCrystal\_I2C.h>

// Motor pins

const int m1 = 13;

const int m2 = 33;

const int m3 = 19;

const int m4 = 23;

// RFID Table IDs

const String tableIDs[] = {"0B00284786E2", "0B002859ADD7", "0B0028640740", "0B00284786E2"};

String currentTableID = "";

// IR Sensor pins

const int rightSensor = 32;

const int leftSensor = 35;

// Motor control pins

const int in1 = 12;

const int in2 = 14;

const int in3 = 26;

const int in4 = 27;

// Keypad setup

#define ROW\_NUM 4

#define COLUMN\_NUM 3

char keys[ROW\_NUM][COLUMN\_NUM] = {

{'1', '2', '3'},

{'4', '5', '6'},

{'7', '8', '9'},

{'\*', '0', '#'}

};

byte pin\_rows[ROW\_NUM] = {18, 5, 17, 4};

byte pin\_column[COLUMN\_NUM] = {0, 2, 15};

Keypad keypad = Keypad(makeKeymap(keys), pin\_rows, pin\_column, ROW\_NUM, COLUMN\_NUM);

// LCD setup

LiquidCrystal\_I2C lcd(0x27, 16, 2);

// Function prototypes

void forward();

void turnRight();

void turnLeft();

void stop();

void arm();

void setup() {

// Motor pins setup

pinMode(m1, OUTPUT);

pinMode(m2, OUTPUT);

pinMode(m3, OUTPUT);

pinMode(m4, OUTPUT);

// LCD setup

lcd.init();

lcd.backlight();

// Serial setup

Serial.begin(9600);

Serial2.begin(9600);

// IR sensor setup

pinMode(rightSensor, INPUT);

pinMode(leftSensor, INPUT);

// Motor control pins setup

pinMode(in1, OUTPUT);

pinMode(in2, OUTPUT);

pinMode(in3, OUTPUT);

pinMode(in4, OUTPUT);

stop();

delay(1000);

lcd.setCursor(0, 0);

lcd.print("ROBOT WAITER");

delay(1000);

lcd.clear();

}

void loop() {

lcd.setCursor(0, 0);

lcd.print("Enter Table");

lcd.setCursor(0, 1);

lcd.print("No");

delay(1000);

char key = keypad.getKey();

if (key) {

Serial.println(key);

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("Going to Table");

lcd.setCursor(0, 1);

lcd.print(key);

delay(1000);

lcd.clear();

int tableIndex = key - '1'; // Convert key to table index

if (tableIndex >= 0 && tableIndex < 4) {

currentTableID = tableIDs[tableIndex];

}

}

while (!currentTableID.isEmpty()) {

if (digitalRead(rightSensor) == LOW && digitalRead(leftSensor) == LOW) {

forward();

} else if (digitalRead(rightSensor) == HIGH && digitalRead(leftSensor) == LOW) {

turnRight();

} else if (digitalRead(rightSensor) == LOW && digitalRead(leftSensor) == HIGH) {

turnLeft();

} else if (digitalRead(rightSensor) == HIGH && digitalRead(leftSensor) == HIGH) {

stop();

}

Serial.println("IN LOOP");

delay(1000);

if (Serial.available() > 0) {

String data = "";

while (Serial.available() > 0) {

data += char(Serial.read());

delay(10);

}

if (data.length() > 0) {

Serial.println(data);

delay(500);

if (currentTableID.equals(data)) {

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("TABLE Reached");

lcd.setCursor(0, 1);

lcd.print(currentTableID);

delay(1000);

lcd.clear();

Serial.println("TABLE");

stop();

delay(5000);

arm();

currentTableID = "";

} else if (tableIDs[2].equals(data)) {

stop();

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("KITCHEN Reached");

lcd.setCursor(0, 1);

lcd.print(currentTableID);

delay(1000);

lcd.clear();

currentTableID = "";

}

}

}

}

}

void forward() {

digitalWrite(in1, HIGH);

digitalWrite(in2, LOW);

digitalWrite(in3, LOW);

digitalWrite(in4, HIGH);

}

void turnRight() {

digitalWrite(in1, LOW);

digitalWrite(in2, HIGH);

digitalWrite(in3, LOW);

digitalWrite(in4, HIGH);

}

void turnLeft() {

digitalWrite(in1, HIGH);

digitalWrite(in2, LOW);

digitalWrite(in3, HIGH);

digitalWrite(in4, LOW);

}

void stop() {

digitalWrite(in1, LOW);

digitalWrite(in2, LOW);

digitalWrite(in3, LOW);

digitalWrite(in4, LOW);

}

void arm() {

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("SERVING FOOD");

lcd.setCursor(0, 1);

lcd.print("ARM ON");

delay(1000);

digitalWrite(m1, HIGH);

digitalWrite(m2, LOW);

delay(1000);

digitalWrite(m1, LOW);

digitalWrite(m2, LOW);

delay(1000);

digitalWrite(m3, HIGH);

digitalWrite(m4, LOW);

delay(1500);

digitalWrite(m3, LOW);

digitalWrite(m4, HIGH);

delay(2000);

digitalWrite(m3, LOW);

digitalWrite(m4, LOW);

delay(1000);

digitalWrite(m1, LOW);

digitalWrite(m2, HIGH);

delay(1000);

digitalWrite(m1, LOW);

digitalWrite(m2, LOW);

delay(1000);

lcd.clear();

}