

به نام خدا

دانشگاه صنعتی امیرکبیر (پلی تکنیک تهران)

دانشکده مهندسی کامپیوتر



آزمایشگاه ریزپردازنده

گزارش کار نیم پروژه

یعقوب برقی رایگان

استاد درس: مهندس معصوم زاده

مهدی رحمانی / ۹۷۳۱۷۰۱

محمد امین رضائی / ۹۷۳۱۰۲۴

یعقوب برقی رایگان!



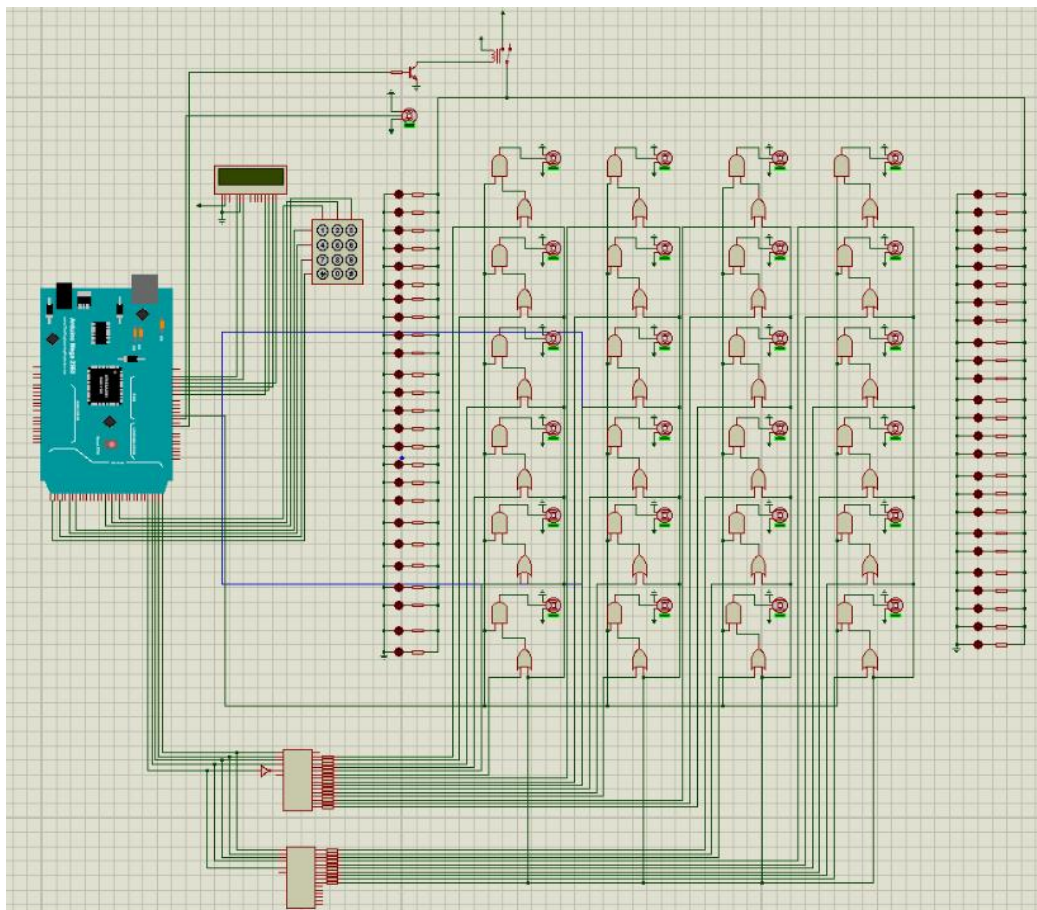
هدف پروژه

سیر کردن دانشجو ها. یک گذرواژه هفت رقمی (کیبورد) در را باز می کند (یک سروو موتور) همه پیچش ها را (تعداد زیادی سروو موتور) به عقب می چرخاند تا غذا کار گذاشته شود. همه ی چراغها (LED) در این مرحله خاموش است. دکمه دوباره در را می بندد و چراغها روشن می شوند. سپس دانشجو ها با زدن شماره های دورقمی غذا ها، آن را تحویل می گیرند. همه چیز روی LCD نشان داده شود و دکمه پاک و تایید داشته باشد.

قطعات مورد نیاز:

- بورد Arduino Mega2560
- دیود نورانی (LED) به مقدار لازم
- تعداد زیادی سروو موتور
- کیبورد
- LCD کاراکتری

برای پیاده سازی این پروژه ابتدا مدار زیر را می بندیم:

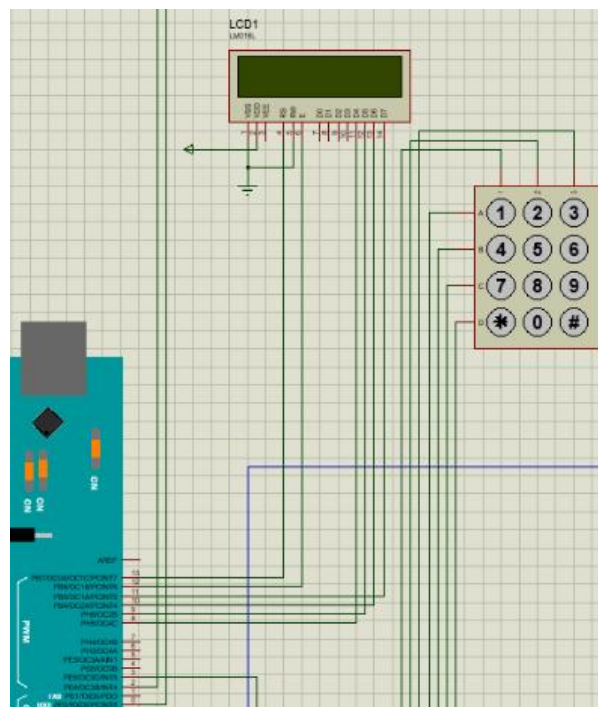


مشاهده می شود که طبق دستور کار، این مدار از تعداد زیادی سروو موتور و دیود تشکیل شده

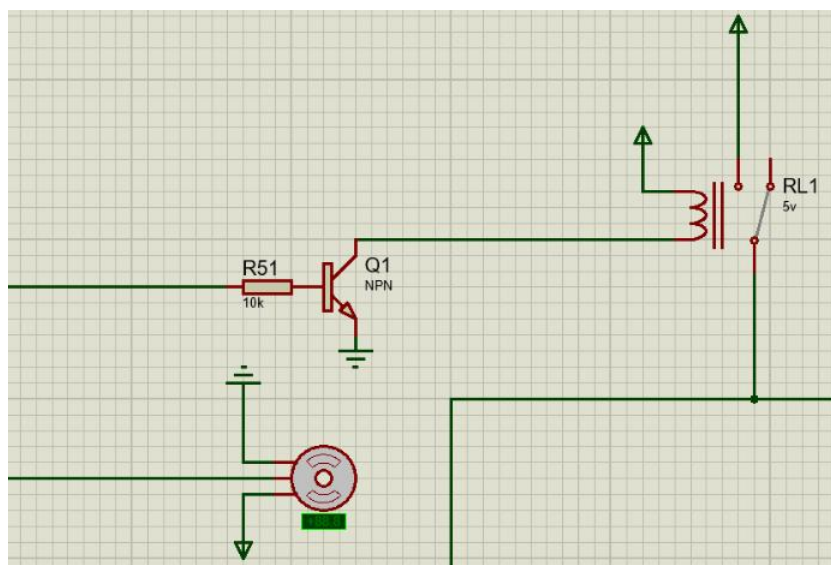
است.

عملکرد کلی مدار به این صورت می باشد که، ابتدا تمام ۲۴ جایگاه خوراکی پر در نظر گرفته می شود و در هر جایگاه ماکسیمم ۵ خوراکی وجود دارد، یعنی تا ۵ بار می شود خوراکی آن خانه را برداشت و بعد از آن پیغام sold out نشان داده می شود. ادمین ابتدا رمز 1397310 را وارد کرده و ال ای دی ها خاموش می شوند، سپس تا وقتی دکمه مربع زده نشده در یعقوب برقی باز است و ال ای دی ها خاموش میباشند. دانشجویان نیز با زدن شماره خوراکی مورد نظرشان، به صورت : 01, 05, 11, 15, 24 می توانند خوراکی بردارند و پیغام done چاپ می شود. هر بار که شارژ انجام میشود، سروو موتور روی ۹۰ قرار میگیرد و با هر بار برداشت به سمت منفی ۹۰ درجه حرکت میکند و با صفر شدن مقدار خوراکی در آن خانه، مقدار منفی ۹۰ می شود.

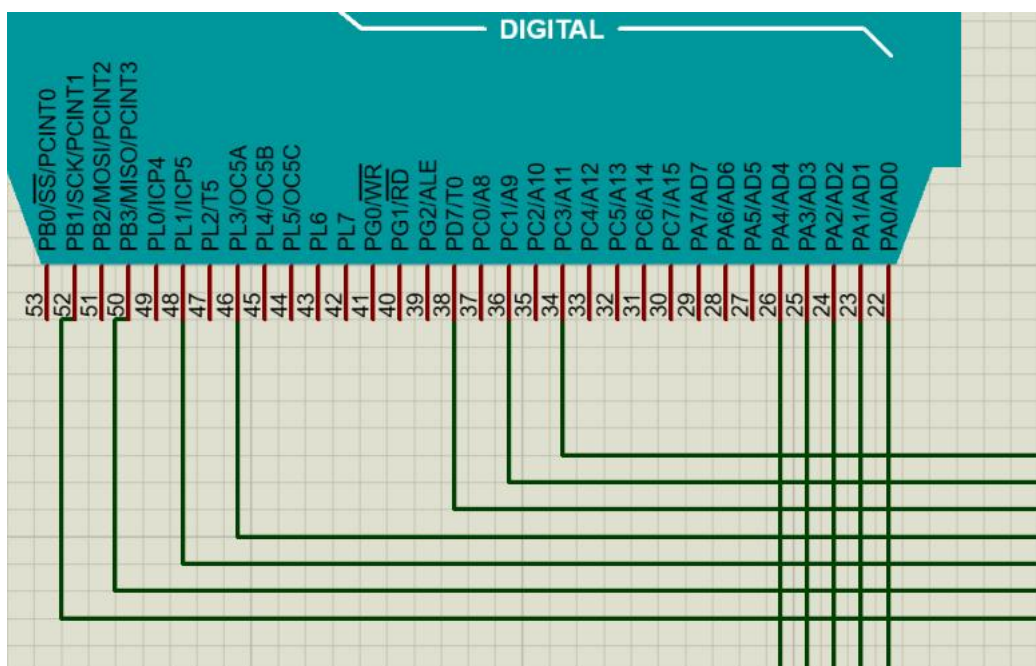
حال اجزای مختلف مدار را به صورت جزئی تر بررسی میکنیم:



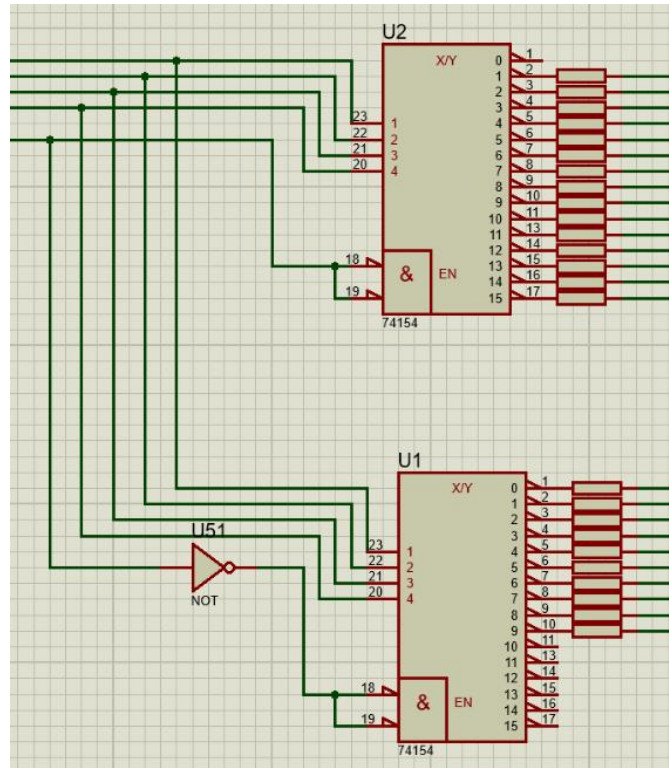
در این بخش می بینیم که ال سی دی و کیبورد به برد آردوینو وصل شده و ۲ پایه نیز برای کنترل ال ای دی ها و یک سروو موتور که وظیفه باز کردن درب یعقوب برقی را دارد در نظر گرفته شده است.



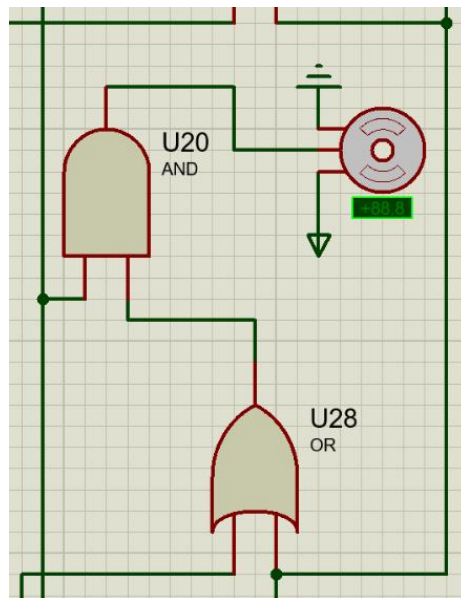
در این بخش ما سروو موتور گفته شده را میبینیم که وظیفه باز کردن در را دارد، و همچنین یک ترانزیستور و یک رله را میبینیم که وظیفه ی روشن و خاموش کردن چراغ های ال ای دی را دارند.



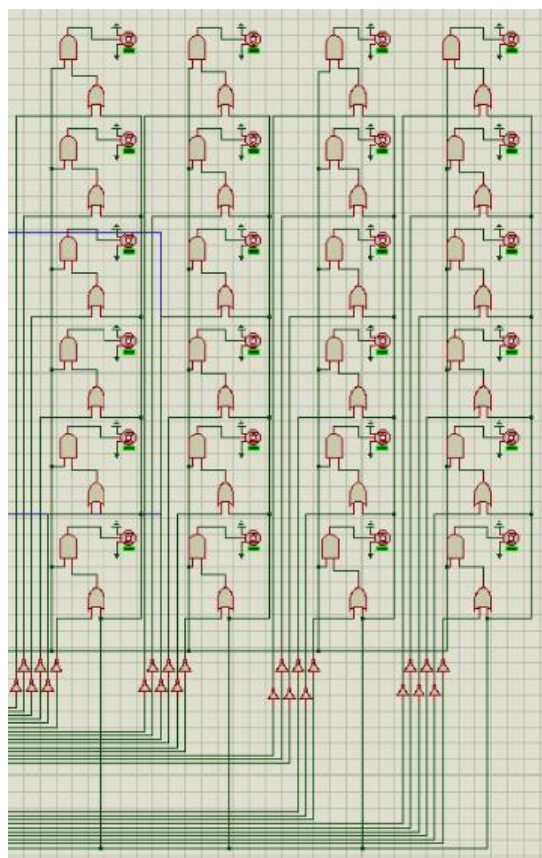
این بخش، پورت های خروجیه دیجیتال برد آردوینو را نشان می‌دهد. برای اینکه ما به ۲۴ پورت دیجیتال برای کنترل ۲۴ سروو موتور خود نیاز داریم، از دیکودر برای آدرس دهی استفاده کردیم.



برای کنترل سروو موتور ها، ما از ۵ سیگنال و ۲ دیکودر ۴ به ۱۶ استفاده کردیم و از ۲۵ تا ۳۲ پورت خروجی دیکودر ها استفاده کردیم.



ساختار کلی کنترل سروو موتور به صورت بالا میباشد که به توضیح آن میپردازیم.



برای حرکت دادن سروو موتور ها ما خروجی شماره ۳ از پایه pwm را در نظر گرفتیم که همیشه آن را با خروجی حاصل از or کردن بیت کنترلی مورد نظر موتور و یک بیت کنترلی دیکودر که وظیفه اش فعال کردن همزمان موتور هاست، and میکنیم تا بتوانیم هر موتور را به موقع بچرخانیم و همزمان هم بتوانیم برای عملیات پر کردن این کار را انجام دهیم.

حال باید به بررسی کد بپردازیم:

```
#include <Keypad.h>
#include <LiquidCrystal.h>
#include <Servo.h>

#define RS_PIN 13
#define E_PIN 12
#define DB7_PIN 11
#define DB6_PIN 10
#define DB5_PIN 9
#define DB4_PIN 8

#define LEDS 0

#define I4 26
#define I3 25
#define I2 24
#define I1 23
#define I0 22

#define FULL_CAPACITY 5

const byte ROWS = 4; //four rows
const byte COLS = 3; //three columns

char keys[ROWS][COLS] = {
  {'1','2','3'},
  {'4','5','6'},
  {'7','8','9'},
  {'*','0','#'}
};

byte rowPins[ROWS] = {46, 48, 50, 52};
byte colPins[COLS] = {34, 36, 38};
```

در این بخش، کتابخانه های مورد نیاز اضافه شده و تعدادی از پین های مورد استفاده تعریف می شوند. همچنین ظرفیت ماکسیمم ۵ تعریف میشود. در ادامه کیبورد تعریف شده و پورت هایش مشخص میشوند.

```

Keypad keypad = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS );
LiquidCrystal lcd(RS_PIN, E_PIN, DB4_PIN, DB5_PIN, DB6_PIN, DB7_PIN);
Servo door;
Servo myservo;

const int passWord[] = {1, 3, 9, 7, 3, 1, 0}; // so is password is 1397310
int lenght = 0;
int input[7] = {-1, -1, -1, -1, -1, -1, -1};
int capacity[24];
int opened = 0;

void setup() {
    pinMode(LEDs, OUTPUT);

    pinMode(I0, OUTPUT);
    pinMode(I1, OUTPUT);
    pinMode(I2, OUTPUT);
    pinMode(I3, OUTPUT);
    pinMode(I4, OUTPUT);

    for (int i = 0; i < 24; i++)
    {
        capacity[i] = FULL_CAPACITY;
    }

    lcd.begin(16, 2);
    myservo.attach(3, 1000, 2000);
    myservo.writeMicroseconds(700);
    door.attach(2, 1000, 2000);
    door.writeMicroseconds(700);
    lcd.clear();
    delay(100);
    closeDoor();

    activeTable();
    delay(15);
    myservo.write(0);
    delay(15);
}

```

در ادامه کیبورد ساخته شده و پین های ال سی دی تعریف میشوند. سپس ۲ سروو، یکی برای باز کردن در و دیگری برای چرخاندن خانه مورد نظر دانشجویان تعریف میکنیم. در ادامه رمز تعریف شده، سپس یک طول تعریف شده برای کار با طول ورودی و مقدار ورودی اولیه تعریف می شود و یک آرایه با ظرفیت ۲۴ برای کار با خانه ها ساخته میشود. سپس یک متغیر open برای باز یا بسته بودن در تعریف میشود.

در بخش setup، خروجی ال ای دی ها و ۵ خروجی برای کنترل موتورها تعریف میشوند. سپس تمام خانه ها در یک حلقه پر میشوند. در ادامه سروو ها تنظیم شده و ال سی دی تعریف میشود.

در ادامه برای شروع کار، دو تابع activeTable برای مقداردهی اولیه به دیکودرها و

closeDoor برای بستن در فراخوانی میشوند. در ادامه ابتدا توابع را تعریف کرده، سپس به سراغ

loop می رویم.


```

void openDoor()
{
    int i = door.read();

    for (; i >= 0; i -= 5)
    {
        door.write(i);
        delay(15);
    }

    activeTable();
    delay(15);
    myservo.write(0);
    delay(15);
    for (int i = 0; i < 24; i++)
    {
        capacity[i] = FULL_CAPACITY;
    }
    digitalWrite(LEDs, LOW);
}

void closeDoor()
{
    int i = door.read();
    for (; i <= 180; i += 5)
    {
        door.write(i);
        delay(15);
    }
    digitalWrite(LEDs, HIGH);
}

```

در تابع openDoor ظرفیت ها پرمی شود و موتور در را باز کند و هر سروو موتور که برای ۲۴ خانه بود، تنظیم می گردد. در تابع closeDoor، مقدار door تنظیم شده و چراغ ها روشن می شود.

```

void mapTo(int number)
{
    int number2 = 0;
    switch(number)
    {
        case 1: number2 = 1;
        break;
        case 2: number2 = 7;
        break;
        case 3: number2 = 13;
        break;
        case 4: number2 = 19;
        break;
        case 5: number2 = 2;
    }
}

```

تابع mapTo وظیفه ی مپ کردن ورودی دانشجو به مدل پیاده شده ما برای آدرس دهی را دارد. در این بخش مقادیر مپ میشوند.

```

switch(number2)
{
    case 1:
        digitalWrite(I0, HIGH);
        digitalWrite(I1, LOW);
        digitalWrite(I2, LOW);
        digitalWrite(I3, LOW);
        digitalWrite(I4, LOW);
        break;
    case 2:
        digitalWrite(I0, LOW);
        digitalWrite(I1, HIGH);
        digitalWrite(I2, LOW);
        digitalWrite(I3, LOW);
        digitalWrite(I4, LOW);
        break;
    case 3:
        digitalWrite(I0, HIGH);
        digitalWrite(I1, HIGH);
        digitalWrite(I2, LOW);
        digitalWrite(I3, LOW);

```

در این بخش نیز بیت های کنترلی مورد نظر باتوجه به عدد مپ شده در عکس قبل، مقدار دهی میشوند.

```

void activeTable()
{
    digitalWrite(I0, HIGH);
    digitalWrite(I1, LOW);
    digitalWrite(I2, LOW);
    digitalWrite(I3, HIGH);
    digitalWrite(I4, HIGH);
}

```

تابع activeTable هم برای مقداردهی اولیه است و بیت های کنترلی غیر استفاده را مقدار میدهد.

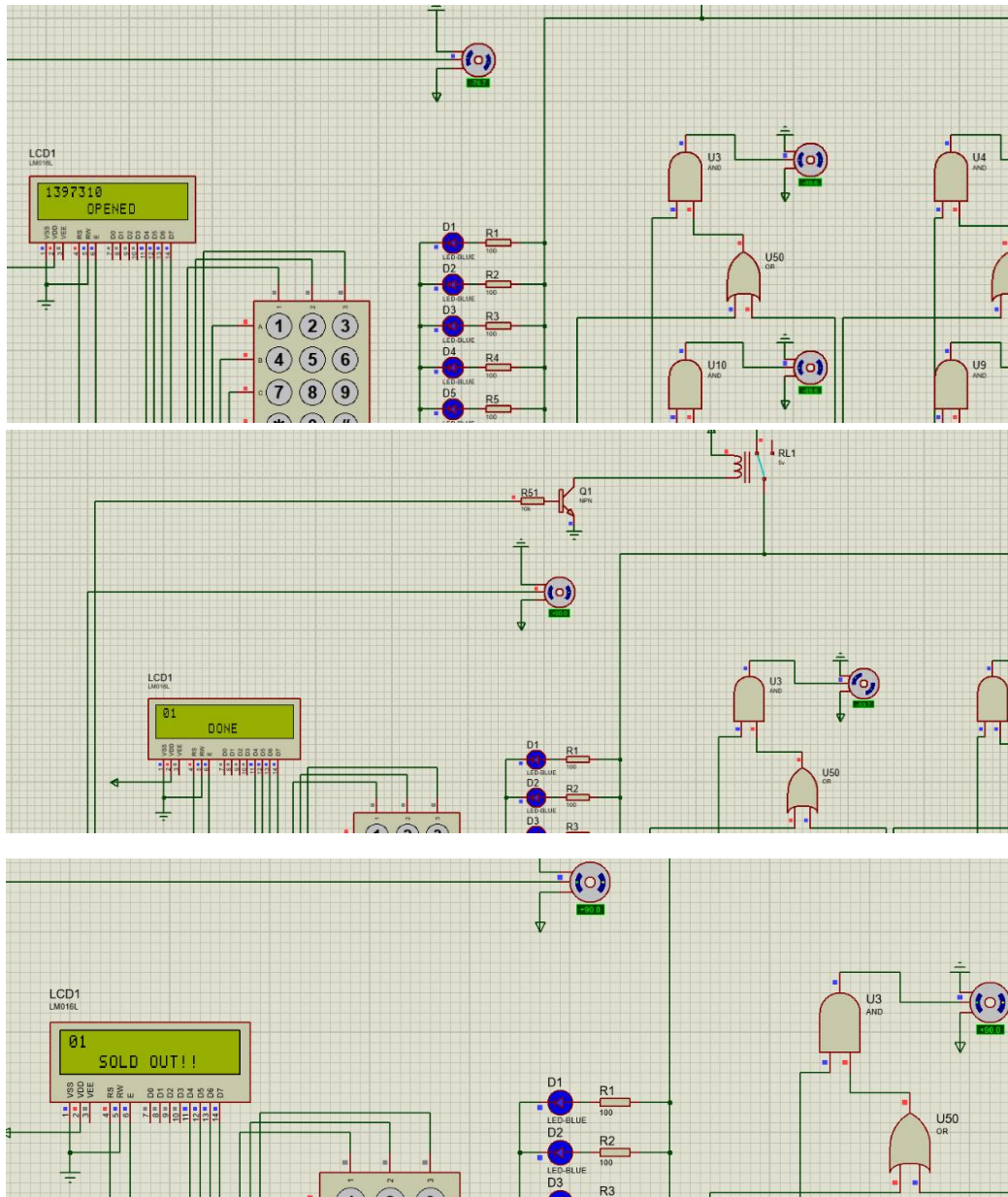
```

78 void loop() {
79   char recieve = keypad.getKey();
80   if (recieve)
81   {
82     if (recieve >= '0' && recieve <= '9' && opened == 0)
83     {
84       if (lenght < 7)
85       {
86         lcd.write(recieve);
87         input[lenght] = recieve - 48;
88         lenght++;
89       }
90     }
91     else if (recieve == '*' && opened == 0) // confirm
92     {
93       if (lenght == 2)
94       {
95         int number = input[0] * 10 + input[1];
96         if (number > 0 && number < 25)
97         {
98           mapTo(number);
99           delay(15);
100
101           if (capacity[number - 1] == 0)
102           {
103             //myservo.write(0);
104             lcd.setCursor(0, 1);
105             lcd.print("   SOLD OUT!!   ");
106             delay(800);
107             lcd.clear();
108           }
109           else
110           {
111             lcd.setCursor(0, 1);
112             lcd.print("   DONE       ");
113             myservo.write((FULL_CAPACITY + 1 - capacity[number - 1])
114               * (180 / FULL_CAPACITY));
115             delay(15);
116             capacity[number - 1] = capacity[number - 1] - 1;
117             delay(800);
118
119
120             lcd.clear();
121           }
122         }
123       }
124       else if (lenght == 7)
125       {
126         int res = 0;
127         for (int i = 0; i < 7; i++)
128           if (input[i] != passWord[i])
129           {
130             res = -1;
131             break;
132           }
133         if (res == 0)
134         {
135           lcd.setCursor(0, 1);
136           lcd.print("   OPENED   ");
137           openDoor();
138           opened = 1;
139           delay(800);
140           lcd.clear();
141         }
142       }
143       else
144       {
145         lcd.setCursor(0, 1);
146         lcd.print("WRONG PASSWORD!!");
147         delay(800);
148         lcd.clear();
149       }
150     }
151     else
152     {
153       lcd.setCursor(0, 1);
154       lcd.print(" WRONG INPUT!! ");
155       delay(800);
156       lcd.clear();
157     }
158     lenght = 0;
159     for (int i = 0; i < 7; i++)
160       input[i] = -1;
161   }
162   else if (recieve == '#') // delete
163   {
164     for (int i = 0; i < 7; i++)
165       input[i] = -1;
166
167     if (opened == 1)
168     {
169       lcd.setCursor(0, 1);
170       lcd.print("   CLOSED   ");
171       closeDoor();
172       opened = 0;
173       delay(800);
174     }
175     lcd.clear();
176     lenght = 0;
177   }
178 }
179 }
180 }

```

حال در loop، چک میشود که اگر ورودی بین ۰ تا ۹ بود، ابتدا اگر طول ورودی کمتر از ۷ بود، ورودی در ال سی دی نمایش داده شده و ورودی درون رشته input ریخته میشود و طول ورودی یکی زیاد می شود. سپس چک می شود که اگر در بسته بود و ورودی ستاره بود، بیا ببین اگر طول ورودی ۲ بود، یعنی دانشجو غذا خواسته، بیا ببین اگر عدد بین ۲۵ تا صفر بود، تابع mapTo را فراخوان، اگر ظرفیت خانه مورد نظر صفر بود، چاپ کن SOLD OUT ، در غیر این صورت، ظرفیت ها را تنظیم کن و سروو موتور را بچرخان. حال اگر ورودی ۷ رقمی باشد، درستی و نادرستی و معتبر بودن آن چک می شود و پیغام های مناسب چاپ می شود.

در آخر هم مدار در حال کار را میبینیم:



یاعلی

پایان