



آزمایشگاه ساختار کامپیوتر و میکروپروسسور

استاد : دکتر هادی اشعریون



تاریخ: ۹۹.۱۰.۰۳

نیمسال اول تحصیلی ۱۴۰۰-۱۳۹۹

پروژه نهایی: شبیه سازی آسانسور



اعضای گروه:

سید علی بحر العلومی مفرد ۹۷۲۴۲۰۱۸

عباس اصفهانی ۹۷۲۴۲۰۱۱

نسرین کریمی ۹۷۲۳۶۰۸۱

دانشگاه شهید بهشتی - پردیس فنی و مهندسی شهید عباسپور

در این پروژه شما باید با استفاده از یک LCD کاراکتری، پنج عدد Button، یک Servo Motor، یک موتور DC و یک میکروکنترلر ATmega32، یک شبیه‌ساز آسانسور بسازید.

آسانسور موردنظر باید برای یک ساختمان ۴ طبقه (دارای یک طبقه همکف)، طراحی شود. موتور DC برای اتاقک آسانسور و Servo Motor برای باز و بسته شدن درب آسانسور در نظر گرفته شود. برای باز شدن درب آسانسور Servo Motor باید در زاویه‌ی صفر درجه و برای بسته شدن درب، Servo Motor باید در زاویه‌ی 90° قرار گیرد.

مدت زمان و جهت کارکرد موتور DC باتوجه به طبقه‌ی مبدأ و مقصد تغییر می‌کند و برای بالا رفتن آسانسور، موتور DC به صورت پادساعتگرد و برای پایین آمدن به صورت ساعتگرد می‌چرخد. برای جابه‌جایی موتور DC به ازای رد شدن از هر طبقه باید به مدت ۵ ثانیه بچرخد. به عنوان مثال برای رفتن از طبقه همکف به طبقه‌ی سوم، موتور DC باید ۱۵ ثانیه در جهت پادساعتگرد کار کند و یا برای جابه‌جا شدن از طبقه‌ی چهارم به دوم، موتور DC باید ۱۰ ثانیه به صورت ساعتگرد بچرخد.

همچنین در طول مدت جابه‌جایی، هنگام رد شدن از هر طبقه (هر ۵ ثانیه) باید طبقه‌ای که در حال حاضر در حال عبور از آن هستیم روی LCD کاراکتری مشخص شود و همچنین هنگامی که به طبقه‌ی مقصد می‌رسیم، شماره طبقه نمایش داده شود.

هنگام رسیدن به مقصد، پس از توقف کامل موتور DC، با یک تأخیر یک ثانیه‌ای باید درب آسانسور توسط Servo Motor باز شد و تا هنگامی که دکمه‌ای (به جز دکمه‌ی مربوط به طبقه‌ای که در حال حاضر آسانسور در آن قرار دارد) فشرده نشود، درب آسانسور باز بماند؛ در صورت انتخاب طبقه‌ی مقصد توسط کاربر، ابتدا درب آسانسور بوسیله‌ی Servo Motor بسته شده و پس از یک ثانیه موتور DC شروع به کار کند.

- در صورتی که کاربر دکمه‌ی مربوط به همان طبقه‌ای که در آن قرار داریم را بفشارد، لازم نیست کار خاصی صورت گیرد و درب آسانسور باز و موتور DC بدون حرکت می‌ماند.
- برای موتور DC در صورت نیاز می‌توانید از ماژول L298 به عنوان درایور استفاده کنید.
- اضافه کردن امکاناتی مانند موتور تهویه هوا با امکان خاموش و روشن کردن توسط کاربر، زنگ اضطراری، خروج سریع (با فشردن یک دکمه، هنگام بسته شدن درب و قبل از راه افتادن موتور DC درب باز شود) و ... اختیاری بوده و در نمره پروژه تأثیر مثبت خواهد داشت.

- ضمناً با توجه به آن که کلیات پروژه برای شما تعریف شده است جزییات پروژه را خودتان می‌بایست برای خودتان تعریف نموده و آنرا تکمیل کنید؛ به همین منظور استفاده از هر گونه خلاقیت و ابتکار مانعی نداشته و امتیاز خواهد داشت.

برنامه های استفاده شده برای انجام این پروژه:

- Code Vision AVR 

- Proteus 

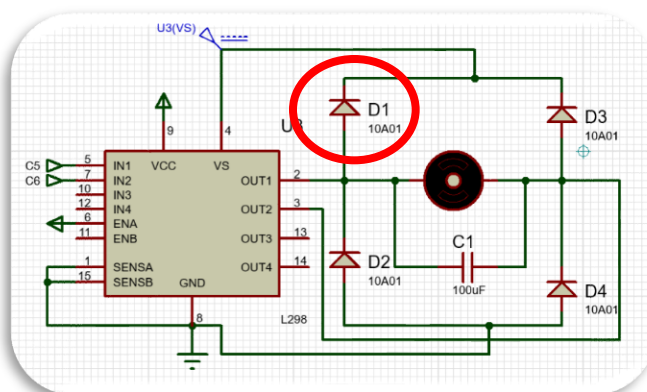
شرح پروتئوس:

اجزای به کار رفته در طراحی:

- Diode 10A01
- ATMEGA128
- CAPACITOR
- KEY-PAD2 (self made)
- KEY-PAD5 (self made)
- L298
- LED-RED
- LED-YELLOW
- LM35
- LM041L
- MOTOR
- MOTOR-PWMSERVO
- OVEN
- RLY-SPNO

* کلاک ۸ مگاهرتز در نظر گرفته شد.

🔌 Diode 10A01:



دیود :

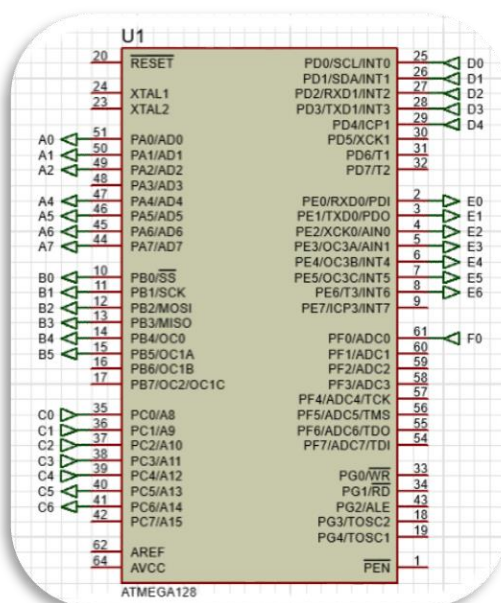
دیود (به انگلیسی: Diode)، (نام های دیگر: یکسوساز) قطعه ای الکترونیکی است که دو سر دارد. دیود، جریان الکتریکی را در یک جهت از خود عبور می دهد (در این حالت، مقاومت دیود ناچیز است) و در جهت دیگر، در مقابل گذر جریان مقاومت بسیار بالایی (در حالت ایده آل، بینهایت) از خود نشان می دهد.



🔌 چرا در این بخش از مدار از دیود استفاده کردیم؟

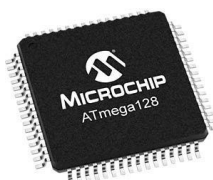
برای برقراری پل یکسوساز دیودی که در اینجا برای تغییر جهت نیاز است و این که جریان معکوس را عبور ندهد. مدار پیاده سازی شده در دیتا شیت L298 موجود است و موتور و مدارات جانبی توسط دیتا شیت به آن متصل شده اند.

ATMEGA 128:



میکرو کنترلر:

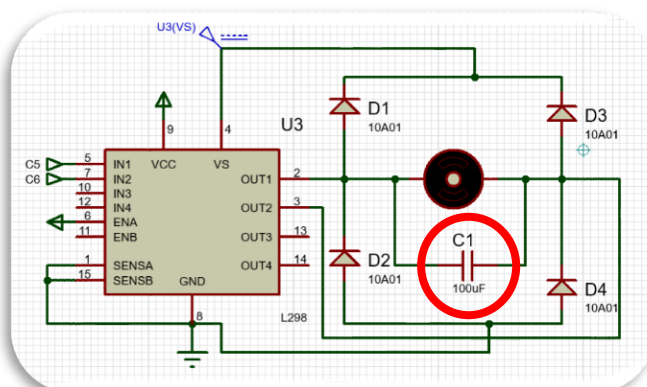
میکرو کنترلر ها که نوعی چیپ الکترونیکی قابل برنامه ریزی محسوب می شوند و قابل استفاده جهت کنترل نمودن دستگاه هایی هستند که نیاز به ذخیره اطلاعات و اندازه گیری و اجرای آن دارند. یک شبه رایانه هوشمند در ابعادی کوچک، که کاربر با برنامه نویسی قادر به کنترل عملیات آن خواهد بود. در واقع می توان گفت میکرو کنترلر AVR دارای تمام ویژگی های یک رایانه، با قابلیت پردازش و محاسبات، اما در ابعادی محدودتر است.



چرا در این بخش از ATMEGA128 استفاده کردیم؟

بعلت کم بودن پایه های atmega32 و بجای جبران این کمبود توسط مالتی پلکسر از میکرو کنترلر atmega128 استفاده کردیم.

⚡ CAPACITOR:



خازن:

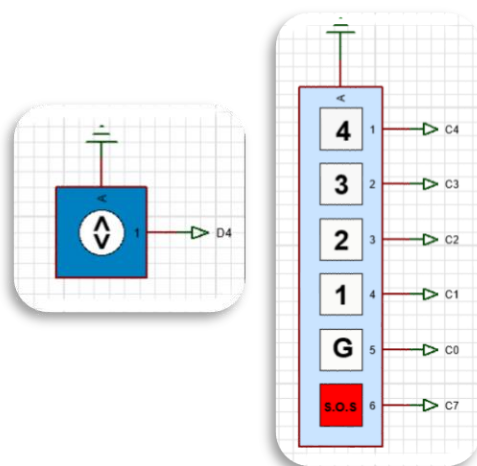
قطعه ای است که برای ذخیره انرژی الکتریکی (ولتاژ) توسط میدان الکترواستاتیکی (بار الکتریکی)، در مدار استفاده می شود و با توجه به اینکه بار الکتریکی در خازن ذخیره می شود می توان از آن برای ایجاد میدان الکتریکی یکنواخت و پایدار استفاده کرد. از خازن ها به عنوان فیلتر نیز استفاده می کنند زیرا سیگنال های متناوب یا AC را به راحتی عبور می دهند ولی مانع عبور سیگنال های مستقیم یا DC می شوند.



⚡ چرا در این بخش از CAPACITOR استفاده کردیم؟

با توجه به کاربرد ذکر شده در بالا، خازن پیاده سازی شده در دیتا شیت L298 موجود است و موتور و مدارات جانبی توسط دیتا شیت به آن متصل شده اند.

KEYPAD:



کیپد:

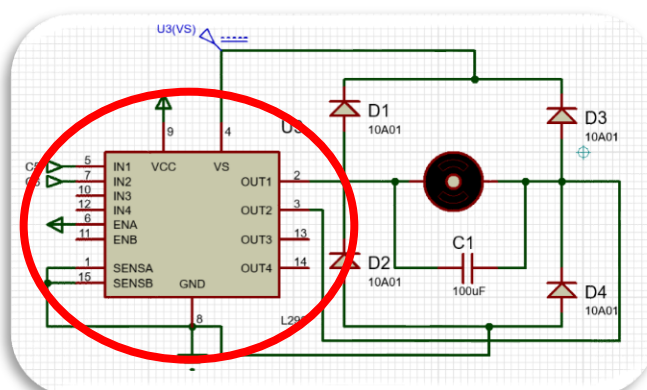
صفحه کلید ها در پایین ترین سطح خود به صورت ماتریسی از سطر ها و ستون ها تشکیل شده اند که keypad نام دارند. میکروکنترلر از طریق پورت ها می تواند به این سطر ها و ستون ها دسترسی داشته باشد.



چرا در این بخش از keypad استفاده کردیم؟

در این مرحله از شبیه سازی برحسب نیاز تعداد گزینه های کیپد را تغییر دادیم. از کیپد برای دریافت داده توسط کاربر استفاده می شود. برای مثال شماره طبقه مقصد، کلید هشدار و ...

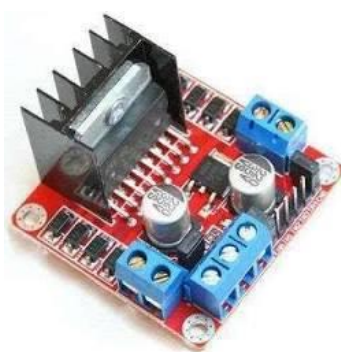
L298:



درایور موتور:

درایور موتور Motor Driver از مدارهای مهم در رباتیک و پروژه‌های مختلف می‌باشد. دسته بندی درایور موتور بر اساس نوع تراشه و میزان ولتاژ و به خصوص جریان عبوری از درایور اهمیت دارد. از آنجایی که راه اندازی موتورهای مختلف نیاز به تامین جریان و ولتاژ مجزا دارد، برای مدیریت موتور Motor هم نیاز به تراشه و مدار کنترل موتور Motor Driver می‌باشد که آن ها بر اساس جریان عبوری و ولتاژ قابل عبور توسط مازول دسته بندی می‌شوند.

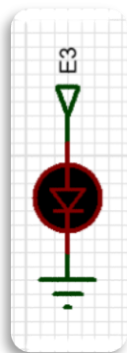
درایور موتور Motor Driver یا راه انداز موتور برای به حرکت در آوردن موتور به کار می‌روند. این نوع مدارها به ازای اطلاعات دریافتی از میکرو و یا کنترل کننده، ولتاژ مورد نیاز موتور را تامین می‌کنند.



چرا در این بخش از درایور موتور استفاده کردیم؟

برای کنترل سروو موتور و همچنین کنترل جهت چرخشش از موتور درایور استفاده کردیم.

🔌 LED:



: LED

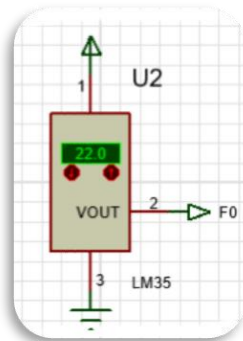
«دیود نور افشان» (Light Emitting Diode) یا LED، نوع خاصی از دیودها است که مشخصات الکتریکی بسیار مشابهی با یک دیود پیوند PN دارد. این بدین معنی است که یک LED، جریان را در جهت مستقیم عبور خواهد داد، اما آن را در جهت معکوس سد خواهد کرد. ال ای دی از یک لایه بسیار نازک از ماده نیمه‌رسانا با ناخالصی تقریباً زیاد ساخته شده و بسته به ماده نیمه‌رسانا و مقدار ناخالصی، وقتی بایاس مستقیم شود، یک نور رنگی با طول موج طیفی مشخص ساطع می‌کند.



🔌 چرا در این بخش از LED استفاده کردیم؟

برنامه را به گونه ای شبیه سازی کردیم که هنگام فشردن کلید مقصد آسانسور چراغی به رنگ زرد در آن طبقه روشن شود. در حالت اضطراری نیز هنگام فشردن کلید S.O.S ال ای دی های قرمز رنگ روشن خواهند شد.

🔧 LM35:



: LM35

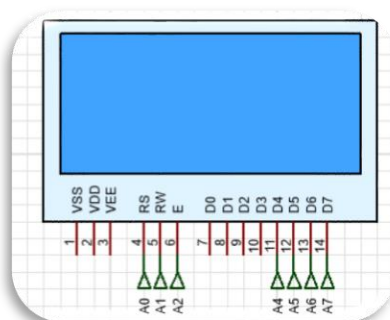
معروفترین Sensor تشخیص دما موجود در بازار Lm35 می باشد ، این سنسور تغییرات دمای مورد نظر را به ولتاژ آنالوگ تبدیل می کند.



🔧 چرا در این بخش از LM35 استفاده کردیم؟

برای تشخیص دمای کابین از این سنسور استفاده کرده و شروطی را در راستای آن برایش تعیین کردیم از قبیل: هرگاه دما بالای ۲۵ درجه شد فن شروع به کار کرده و هرگاه از ۲۰ درجه کمتر شد هیتر شروع به کار می کند.

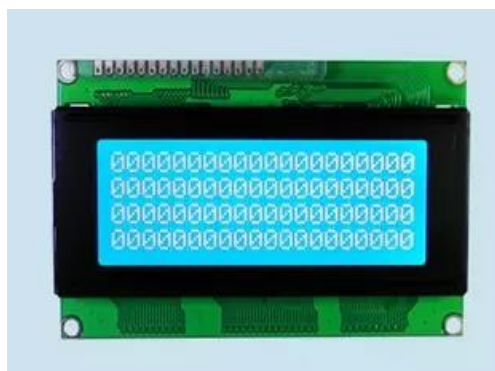
LM041L:



ال سی دی:

یک ال سی دی (نمایشگر کریستال مایع) بر اساس عکس العمل کریستال مایع در میدان الکتریکی کار می کند. هر ال سی دی دارای یک نور زمینه در پشت کریستال مایع می باشد که به عنوان منبع نور عمل می کند. هنگامی که میدان الکتریکی روی کریستال مایع اعمال می شود کریستال مایعی که قبلاً نور را از خود عبور می داد، با تغییر جهت مولکول هایش مانع از عبور نور می گردد. بنابراین با اعمال ولتاژهای مناسب قادر به کنترل عبور یا عدم عبور نور از پیکسل های مختلف ال سی دی خواهیم بود.

LCD های مختلفی در بازار موجودند مانند LCD 16×2، LCD 16×4، LCD 20×2، LCD 20×4، ال سی دی گرافیکی ۱۲۸ در ۶۴ پیکسل و



چرا در این بخش از ال سی دی استفاده کردیم؟

در این شبیه سازی از ال سی دی برای نمایش مقدار دما، شمارنده، شماره طبقه مقصد آسانسور، باز و بسته بودن در آسانسور و حرکت رو به بالا و پایین بودن را نمایش می دهد.

🔌 MOTOR:



موتور:

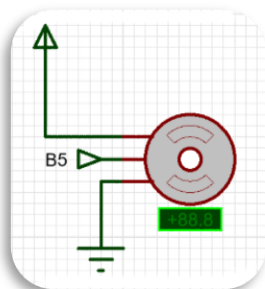
موتور DC وسیله ای است که انرژی الکتریکی را به کمک یک آهنربا به انرژی مکانیکی تبدیل میکند. به علت اینکه از جریان برق مستقیم یا DC 24 ولت استفاده میکند به آن DC میگویند.



چرا در این بخش از موتور دی سی استفاده کردیم؟

از این ابزار در دو بخش شبیه سازی استفاده کردیم. یک مورد برای فن مورد دیگر در مدار راه اندازی درایور موتور که در اینجا انرژی الکتریکی به انرژی مکانیکی تبدیل می شود و برای اتاقک آسانسور از آن استفاده شده است.

🔧 MOTOR-PWMSERVO:



سروو موتور:

سروو موتور DC یک نوع از انواع سروو موتورها می باشد که در دو نوع مختلف تولید می شود؛ سروو موتور DC جاروبکدار و سروو موتور DC بدون جاروبک (براشلس). سروو موتور DC همانند سایر سروو موتورها یک نوع از انواع موتورهای الکتریکی دارای قدرت بالا و دقیقی می باشد که به دور یک محور و با یک زاویه ی مخصوص دوران پیدا می کند، این موتورهای الکتریکی از سه بخش تشکیل شده اند:

- موتور مستقیم (DC)

- گیربکس

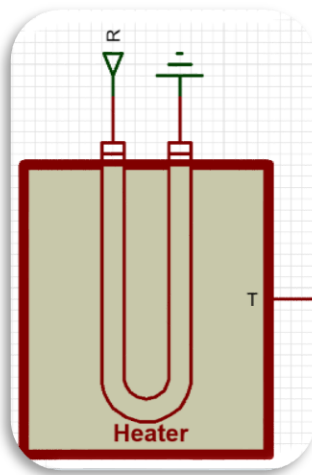
- برد الکترونیکی (میکرو کنترلر): برد الکترونیکی یا پتانسیومتر کار عمده و اساسی ای که برعهده دارد این است که اندازه ی زاویه ی گردش را تحت کنترل خود دارد؛ در واقع اصل وظیفه ی پتانسیومتر تنظیم کردن خروجی چرخ دنده است به این صورت که پس از دریافت نمودن سیگنال های مناسب، خروجی چرخ دنده را به صورتی در کنترل دارد که این خروجی به طور کامل مطابق با همان زاویه ای که مورد نیاز است، گردش داشته باشد.



🔧 چرا در این بخش از سروو موتور استفاده کردیم؟

برای شبیه سازی باز شدن و بسته شدن در آسانسور از سروو موتور استفاده کردیم. و برای باز شدن زاویه صفر درجه و بسته شدن زاویه ۹۰ درجه را قرار دادیم.

🔧 OVEN:



هیتر:

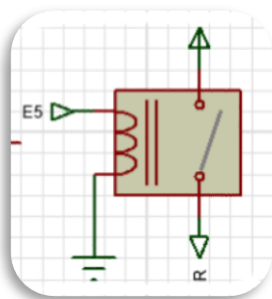
وسیله ای که برای گرم کردن محیط از آن استفاده می شود و با دماسنج می توان روشن شدن یا خاموش کردن آن را کنترل کرد.



🔧 چرا در این بخش از هیتر استفاده کردیم؟

یکی از قابلیت های جدیدی که به آسانسور اضافه کردیم روشن شدن هیتر وقتی که دماسنج دمای کمتر از ۲۰ درجه را تشخیص دهد است. هیتر روشن شده و محفظه کابین آسانسور روشن شده و دما را با رساندن به مقدار ۲۰-۲۵ درجه در حالت تعادل نگه می دارد.

🔌 RLY-SPNO:



رله:

رله یک قطعه الکترومکانیکی است که کاربرد آن در مدارات الکتریکی مانند یک کلید ساده می باشد با این تفاوت که برخلاف کلید که برای باز و بسته شدن آن باید به صورت دستی عمل کرد در رله باز و بسته شدن بوسیله جریان الکتریکی انجام می شود.

استفاده از رله این امکان را به ما می دهد که بتوانیم دو بخش مجزای یک سیستم را که دارای دو منبع ولتاژ مختلف هستند، از هم جدا کنیم. به عنوان مثال با این قطعه میتوانیم یک موتور الکتریکی ۲۲۰ ولت را به وسیله میکرو کنترلر که دارای ولتاژ ۵ ولت است بدون هرگونه اتصال فیزیکی بین آنها راه اندازی کنیم.



🔌 چرا در این بخش از رله استفاده کردیم؟

در حالت عادی میکرو نمی تواند هیتر را روشن و یا خاموش کند برای این منظور ما از یک رله استفاده کردیم تا توسط آن هیتر روشن و خاموش کنیم.

چون میکرو کنترلر بیشتر از ۵ ولت نمی تواند در هر پایه خروجی بدهد ولی ما در اینجا یک ولتاژی به پایه رله دادیم و از رله خروجی را گرفته و به هیتر انتقال دادیم که هرگاه با میکرو کنترلر فرایندی را انجام دادیم آن جریان متناسب با عملکرد ما قطع یا وصل خواهد شد و هیتر را روشن یا خاموش خواهد کرد بدون نیاز به دخالت فیزیکی برای روشن کردن هیتر.

شرح کد ویژن:

```
// ketabkhane haye morede niaz ra tarif mikonim
#include <megal28.h>
#include <delay.h>
#include <alcd.h>
#include <stdio.h>
```

در ابتدا قبل از شروع برنامه نویسی درون و اجرای برنامه لازم است کتابخانه های مربوط به دستوراتی که قرار است در نوشتن برنامه از آن ها استفاده کنیم را قرار دهیم.

کتابخانه اول مربوط به میکروکنترلر ما است که از نوع ATMEGA 128 است.

کتابخانه دوم مربوط به استفاده از تابع تاخیر در برنامه است.

کتابخانه سوم مربوط به استفاده از نمایشگر در شبیه سازی ما است که از توابع نمایشگر با تعریف کتابخانه آن استفاده کردیم.

کتابخانه آخر برای استفاده از دستور sprintf در برنامه است.

```
// moteghayer haye morede niaz ra tarif mikonim
int m=0; //counter
int flag_sos=0;
void key_check(void); // tabe chek kilid ha
int temp=0; // moteghayer haye dama
// areye haye LCD
char b[16];
char c[16];
char d[16];
void sos_key(void);
int door=0; // flag dar
void olaviat(void); // tabe olaviat
void fan_heater(void); // tabe fan va heater
void harekat_bck(void); // ahrekat be paein
void lcd_temp(void); // LCD va namayeshe dama va baghie moteghayer ha
void harekat_fow(void); // tabeh harekat ro be bala
//unsigned long int counter=0;
int tabaghe=0; // state tabaghe ra moshakhas mikonad
int key_tabaghe_g=0, key_tabaghe_l=0, key_tabaghe_2=0, key_tabaghe_3=0, key_tabaghe_4=0; // flag haye kilid
int harekat_state=1; // state harekate asansor ra moshakhas mikonad
```

در این بخش از برنامه به معرفی متغیرهای مورد استفاده، آرایه های ال سی دی، فلگ ها و توابع پرداختیم:

۱. متغیر اول که همان m است بعنوان شمارنده آن را در نظر گرفته ایم. چون ما در نوشتن این برنامه حلقه های for را درون توابع تعریف کردیم و توابع را نیز خارج از تابع اصلی void main تعریف کردیم، دیگر نمی توان معرفی شمارنده از نوع اینتیجر را درون void main قرار داد و باید بیرون از تمامی حلقه ها آن را معرفی کرد. در واقع باید از یک متغیر گلوبال استفاده می شد که در هر مرحله و تابع نیاز به معرفی آن نباشد.

۲. تابع key_check به عنوان تابعی که فاقد خروجی است از نوع void معرفی شده است. کاربرد این تابع برای این است که:

```
void key_check(void){ // age kelidi zade beshe ke shamele tabaghe E ke tooshim nashe, flag va LED ro roshan miko
  sos_key();// check kardane kelide sos
  if(((PINC.0==0)|| (PIND.0==0)) && (tabaghe!=0) ){
    key_tabaghe_g=1;//falg ro yek mikone
    PORTB.0=1;// LED tabaghe ro roshan mikone
  }
  if(((PINC.1==0)|| (PIND.1==0)) && (tabaghe!=1)){
    key_tabaghe_l=1;//falg ro yek mikone
    PORTB.1=1;// LED tabaghe ro roshan mikone
  }
  if(((PINC.2==0)|| (PIND.2==0)) && (tabaghe!=2)){
    key_tabaghe_2=1;//falg ro yek mikone
    PORTB.2=1; // LED tabaghe ro roshan mikone
  }
  if(((PINC.3==0)|| (PIND.3==0)) && (tabaghe!=3)){
    key_tabaghe_3=1; //falg ro yek mikone
    PORTB.3=1; // LED tabaghe ro roshan mikone
  }
  if(((PINC.4==0)|| (PIND.4==0)) && (tabaghe!=4)){
    key_tabaghe_4=1; //falg ro yek mikone
    PORTB.4=1; // LED tabaghe ro roshan mikone
  }
}
```

در if اول گفته می شود که اگر دکمه همکف در طبقه یا در داخل کابین فشرده شود (چون به گراند وصل کردیم یعنی هرگاه صفر شوند یعنی دکمه ها فشرده شده اند) در حالیکه طبقه ما، طبقه همکف نباشد، مقصد را طبقه همکف کرده (۵ متغیر که انتهای آن ها g، ۲، ۳، ۴ است و صرفاً فقط صفر و یک می شوند برای اینکار استفاده شده است) و ال ای دی آن طبقه مقصد، روشن خواهد شد. (به رنگ زرد)

حال همین شرط را برای طبقه های اول، دوم، سوم و چهارم تکرار کردیم.

یک تابع دیگر در همین تابع کنترل دکمه استفاده شده به نام sos_key. این تابع زمانی اجرا خواهد شد که کاربر درون کابین آسانسور در وضعیت اضطراری قرار گرفته و درخواست کمک از افراد بیرون را دارد. که با اجرای این تابع ال ای دی های قرمز همه طبقات روشن خواهند شد.

۳. متغیر دیگری تعریف کرده ایم به نام temp که دارای مقدار اولیه صفر است. در واقع این همان متغیری است که وقتی مقدار دما توسط دماسنج خوانده می شود درون این متغیر قرار می گیرد. لازم به ذکر است که برای خواندن سنسور دماسنج از ADC استفاده می کنیم (۸ بیتی استفاده شده است).

۴. متغیرهای بعدی از نوع کاراکتر تعریف شده اند. از این کاراکترها برای نمایش داده و اطلاعات بر روی ال سی دی استفاده شده است چرا که برای مثال وقتی عددی از نوع اینتیجر داریم نمیتوان آن را بصورت مستقیم در ال سی دی نمایش داد بلکه باید هر یک از آن اطلاعات را جداگانه درون آرایه هایی قرار داده و آن آرایه ها در خانه های ال سی دی (ال سی دی استفاده شده ۲۰*۴ است) نمایش می دهد.

```

void lcd_temp(){
    temp=read_adc(0);// adc ra mikhanad va dar moteghayer mirizad
    sprintf(b,"%d",temp);// moteghayer ra be char tabdil mikonad
    //sprintf(a,"%d",j);
    sprintf(c,"%d",tabaghe);//moteghayer ra be char tabdil mikonad
    //sprintf(d,"%d",counter);
    //lcd_gotoxy(0,0);
    //lcd_puts("count:");
    //lcd_puts(d);
    lcd_gotoxy(0,1);// dar khat 0 1 minevisad
    lcd_puts("temp= ");// ebarete temp: ra chaap mikone
    lcd_gotoxy(0,7);// dar khat 0 7 minevisad
    lcd_puts(b);// meghdar temp ro minevise
    lcd_gotoxy(0,2);// dar khat 0 2 minevisad
    lcd_puts("tabaghe= ");// ebarete tabaghe: ra chaap mikone
    lcd_puts(c); // meghdare tabaghe ra minevise
    lcd_gotoxy(0,3);// dar khat 0 3 minevisad
    lcd_puts("harekat= ");
    if(harekat_state==1){lcd_puts("fow");} // bala ye paen raftene asansor ra update mikonad
    if(harekat_state==2){lcd_puts("bak");}
    lcd_gotoxy(11,2);// dar khat 0 1 minevisad
    if(door==1){lcd_puts("cols");} // baz ya baste shodane dar ra update mikonad
    if(door==0){lcd_puts("open");}
    fan_heater();
}

```

همانطور که مشاهده می شود مقدار دما توسط ADC خوانده شده و درون متغیر temp قرار میگیرد. در مرحله بعدی با استفاده از دستور sprintf ما مقدار قرار داده شده در temp را به عدد دهی تبدیل کرده و درون آرایه هایی که در ابتدا از نوع کاراکتر معرفی کردیم قرار گرفته و برای نمایش در ال سی دی آماده می شوند. برای نمایش کاراکتر ها در ال سی دی ابتدا باید مختصات آن خانه ای از ال سی دی که قرار است اولین کاراکتر در آن نمایش داده شود را توسط lcd_gotoxy مشخص کرده و و توسط دستور lcd_puts مقدار داده و اطلاعاتی که میخواستیم نمایش دهیم را با آن در نمایشگر نشان دهیم.

چون در ال سی دی ما حرکت به بالا و پایین و باز و بسته بودن در را نیز نمایش دادیم، برای همین منظور آن شرط ها را قرار دادیم که شرایط و وضعیت آسانسور را در ال سی دی نمایش دهد.

۵. در مرحله بعدی برنامه فلگ door را از نوع اینتیجر تعریف کرده ایم. این فلگ مشخص می کند که در آسانسور باز است یا بسته که ۲ مقدار ۰ و ۱ می تواند داشته باشد. هرگاه مقدارش صفر شود یعنی در باز است و هرگاه یک شود یعنی در بسته است.

۶. توابعی بدون خروجی که در این مرحله از برنامه از نوع void معرفی شده است، تابع اولویت، فن و هیتر، حرکت رو به بالا و پایین و نمایشگر ماست که در ادامه به توضیح عملکرد هر یک از آن ها خواهیم پرداخت.

۷. فلگ دیگری که تعریف شده است، فلگ طبقه از نوع اینتیجر است که دارای مقدار اولیه صفر است. ما در کد همواره به این نیاز داریم که بدانیم در کدام طبقه هستیم. چون هیچ سنسوری این اطلاعات را نمی دهد. برای همین ما یک متغیری به نام

طبقه تعریف کرده ایم که هرگاه یک طبقه بالا می رویم +۱ شده و هرگاه یک طبقه پایین برویم -۱ می شود. و طبقه ۰ به این معنی است که در طبقه همکف قرار داریم و ۴ باشد یعنی ۴ طبقه رو به بالا رفته و در طبقه چهارم قرار داریم.

۸. فلگ های این بخش از برنامه، همان فلگ های هستند که در بخش ۲ توضیح دادیم و مقصد را مشخص می کنند. در ابتدا همگی آن ها به طور پیش فرض صفر هستند یعنی مقصد ما آن طبقه نیست پس نتیجه می گیریم که آسانسور حرکت نخواهد کرد تا هنگامی که کاربری کلیدی را فشار دهد و در نهایت فلگ آن طبقه یک می شود.

۹. فلگ بعدی به نام harekat_state مشخص می کند که استیت حرکت ما چگونه است؛ بالارونده یا پایین رونده. و مقدار ۱ و ۲ برای آن قرار داده می شود. هرگاه مقدار آن برابر ۱ شود یعنی روبه بالا حرکت می کنیم و هرگاه ۲ شود، رو به پایین حرکت می کنیم. با استفاده از این دستور می توانیم شرط ها و دستورات و منطق کد را عمل کنیم به این صورت که وقتی رو به بالا باید حرکت کرد آسانسور رو به پایین حرکت نکند و اولویت رو به بالا باشد.

بخش بعدی برنامه توسط بخش کدویزار کد ویژن، نوشته شده است:

```
#define ADC_VREF_TYPE ((1<<REFS1) | (1<<REFS0) | (1<<ADLAR))
// Read the 8 most significant bits
// of the AD conversion result
3 unsigned char read_adc(unsigned char adc_input)
3 {
    ADMUX=adc_input | ADC_VREF_TYPE;
    // Delay needed for the stabilization of the ADC input voltage
    delay_us(10);
    // Start the AD conversion
    ADCSRA|=(1<<ADSC);
    // Wait for the AD conversion to complete
    while ((ADCSRA & (1<<ADIF))==0);
    ADCSRA|=(1<<ADIF);
    return ADCH;
}
3 void main(void)
3 {
    DDRB=0xFF;
    DDRC=0xE0;
    PORTC=0x1F;
    DDRD=0x00;
    PORTD=0xFF;
    DDRE=0xFF;
    DDRG=0xFF;
    // Timer(s)/Counter(s) Interrupt(s) initialization
    TIMSK=(0<<OCIE2) | (0<<TOIE2) | (0<<TICIE1) | (0<<OCIE1A) | (0<<OCIE1B) | (0<<TOIE1) | (0<<OCIE0) | (1<<TOIE0);
```

```
// ADC initialization
// ADC Clock frequency: 500/000 kHz
// ADC Voltage Reference: Int., cap. on AREF
// Only the 8 most significant bits of
// the AD conversion result are used
ADMUX=ADC_VREF_TYPE;
ADCSRA=(1<<ADEN) | (0<<ADSC) | (0<<ADFR) | (0<<ADIF) | (0<<ADIE) | (1<<ADPS2) | (0<<ADPS1) | (0<<ADPS0);
SFIOR=(0<<ACME);
TCNT1 = 0;          /* Set timer1 count zero */
ICR1 = 2499;        /* Set TOP count for timer1 in ICR1 register */
/* Set Fast PWM, TOP in ICR1, Clear OC1A on compare match, clk/64 */
TCCR1A = (1<<WGM11)|(1<<COM1A1);
TCCR1B = (1<<WGM12)|(1<<WGM13)|(1<<CS10)|(1<<CS11);
OCR1A=187;
// Alphanumeric LCD initialization
// Connections are specified in the
// Project|Configure|C Compiler|Libraries|Alphanumeric LCD menu:
// RS - PORTA Bit 0
// RD - PORTA Bit 1
// EN - PORTA Bit 2
// D4 - PORTA Bit 4
// D5 - PORTA Bit 5
// D6 - PORTA Bit 6
// D7 - PORTA Bit 7
// Characters/line: 16
```

```
lcd_init(16);
asm("sei")
// Global enable interrupts
// Timer 0 overflow interrupt service routine
//interrupt [TIM0_OVF] void timer0_ovf_isr(void)
//{
//// Reinitialize Timer 0 value
//
//TCNT0=0xB2;
//// Place your code here
// counter+=1;
//}
```

حال در این بخش از برنامه به حلقه اصلی while(1) می‌رسیم:

```
////////////////////////////////while////////////////////////////////
while(1){
    lcd_temp();
    olaviat();
    //PORTB.6=1;
    //sos_key();
}
// payane void
}
```

در این بخش به برنامه دستور می دهیم که تابع اولویت و نمایشگر ال سی دی اجرا شود.

حال تابع void main به پایان رسید. در ادامه کد توابع را تعریف کرده ایم.

در ابتدا به توضیح عملکرد تابع اولویت می پردازیم:

```
void olaviat(void){
    key_check();//in tabe kelid ha ra check mikonad
    sos_key();// check kardane kelide sos
    if((key_tabaghe_g==0) && (key_tabaghe_1==0) && (key_tabaghe_2==0) && (key_tabaghe_3==0) && (key_tabaghe_4==0)
    // asansor dar yek tabaghe stop mikone ta dokme bezanan
    sos_key();// check kardane kelide sos
    }
    //dar paein shart haye harekat kardane asansor ra darim
    else{
        // baraye tabaghe g boodan varede shart zir mishavim
        if(tabaghe==0){
            //agar tabagheye maghsad morede nazar g bashad, bayad dar baz shavad
            if(key_tabaghe_g==1){
                // inja bayad bege esme tabaghe ro tabaghe
                //dar baz beshe va baste beshe
                for(; m<500 ; m+=1){
                    //OCRI1A=300;//close
                    OCRI1A=187;//open
                    key_check();
                    lcd_temp();
                    door=0;//dar baz mishe
                }
                m=0;//shomarande sefr mishe
                key_tabaghe_g=0;// flag tabaghe g bayad khamoosh beshe
                harekat_state=1;// harekat state bayad be samte bala shavad
                PORIB.0=0;// LED tabaghe g bayad khamoosh beshe
            }
            //agar maghsad g nabashad varede shart zir mishavim
            else{
                // check mikone ke tabagh chandome maghsad
                if((key_tabaghe_1==1)|| (key_tabaghe_2==1) ||(key_tabaghe_3==1)|| (key_tabaghe_4==1)){
                    // harekat be andaze yek tabaghe be samte bala
                    harekat_fow();// tabe harekat ro be bala baraye yek tabaghe
                }
            }
        }
        // payane shart haye tabaghe G
    }
    //tabaghe 1 agar bashim code varede shart zir mishavad
    else if(tabaghe==1){
```

همانطور که قابل مشاهده است ابتدا تابع اولویت، تابع key_check را اجرا می کند که یک دور رفرش می کند که بفهمد کلیدی زده شده یا نه. در مرحله بعدی یک شرط را تعریف می کند که اگر کلیدی زده شده باشد و اگر کلیدی زده شده باشد.

اگر کلیدی فشرده نشده باشد و تمامی فلگ های هر طبقه (مقصد) صفر باشد هیچ کاری نمیکند. حال اگر مقصدی برای آسانسور تعیین شود ۵ شرط برای ۵ طبقه (با حضور طبقه همکف) قرار می دهد.

*در حالت پیش فرض آسانسور در طبقه همکف شروع به کار می کند.

برای توضیح بهتر کد از یک مثال برای همراهی توضیح استفاده می کنیم:

فرض می کنیم فردی کلید طبقه سوم را فشار می دهد، در نتیجه متغیر `key_tabaghe_3==1` می شود. در نتیجه `if` ابتدایی که در ابتدا درباره عملکرد آن توضیح دادیم برقرار نیست و وارد استیت بعدی می شود.

وقتی وارد `else` می شویم در ابتدا چک می کند که در حال حاضر در کدام طبقه قرار داریم. در ابتدا ما در طبقه همکف هستیم (بصورت پیش فرض در ابتدای برنامه) و طبقه سوم فشار داده شده است. پس در نتیجه وارد شرط `tabaghe==0` خواهد شد.

حال در این شرط چک می کند که آیا مقصد همان طبقی فعلی است یا نه.

اگر مقصد خود طبقه باشد: تنها در آسانسور را باز می کند و ال ای دی طبقه را روشن و خاموش می کند. (برای باز شدن در آسانسور از این شرایط استفاده کردیم که اگر حلقه ای تعریف کنیم که برای مدت زیادی دستور دهد که در آسانسور فقط باز شود، در نتیجه در آسانسور بطور مداوم باز خواهد ماند (بجای استفاده از تابع دیلی چون اگر دیلی را قرار می دادیم قابلیت چک کردن مداوم فشرده شدن کلید را از دست می دادیم) و همانطور که در دستور کار گفته شده باید برای باز شدن زاویه صفر درجه و برای بسته شدن زاویه ۹۰ درجه قرار داده شود در نتیجه برای موتور درایور اعداد متناظر زاویه ها را قرار دادیم و اطلاعات دمایی را در نمایشگر نشان داده و دوباره رفرش می کنیم که آیا دکمه طبقه دیگری فشرده شده است یا نه و فلگ در را صفر می کنیم یعنی همان باز شدن و شمارنده را صفر کرده و طبقه مقصد را همان مقدار قرار داده و ال ای دی را خاموش می کند).

حال اگر مقصد همان طبقه نباشد (همین موردی که ما برای مثال استفاده کردیم): برنامه چک می کند که آیا طبقه مقصد فشرده شده در بالای طبقه فعلی قرار دارد یا نه (چون اینجا همکف است پس شرط اجرا می شود) و یک طبقه رو بالا رفته و تابع حرکت رو بالا را

```
//tabaghe 1 agar bashim code varede shart zir mishavad
else if(tabaghe==1){
    // age tabaghe 1 maghsad bashe varede shart zir mishe
    if(key_tabaghe_1==1){
        // inja bayad bege esme tabaghe ro tabaghe
        //dar baz beshe va baste beshe
        for(; m<500 ; m+=1){
            //OCR1A=300;//close
            OCR1A=187;//open
            key_check();
            door=0;
            lcd_temp();
        }// dar baz mishavad
        m=0; // shomarande sefr mishe
        key_tabaghe_1=0;// flag tabaghe aval sefr mishe
        PORTC.5=0;PORTC.6=0;//motor bi harekat mishe
        OCR1A=187;// dar baz mishe vali in khat tekrarie yadam rafte pakesh konam :)
        PORTB.1=0; //LED tabaghe aval khamoosh mishe
    }
    // age maghsad tabaghe dige E bashe mire tu shart zir
    else{
        //check mikone bebine harekat state ro be balast ya paEn
        if(harekat_state==1){
            // age be bala bood harekat, aval mibine ke maghsadesh aya tabaghe bala hast ya na
            if( (key_tabaghe_2==1) || (key_tabaghe_3==1) || (key_tabaghe_4==1) ){
                harekat_fow();
            }
        }
    }
}
```

اجرا میکند (طبقه یک واحد رفته به سمت بالا و فلگ طبقه را نیز یک واحد افزایش می دهد و فلگ استیت حرکت را ۱ می کند) که در ادامه به توضیح آن خواهیم پرداخت، پس ما نه تنها موتور را به اندازه ی واحد به بالا می کشیم و به مدت ۵ ثانیه آن را می چرخانیم، متغیر طبقه را نیز در نهایت +۱ می کنیم.

```
if(harekat_state==1){
    // age be bala bood harekat, aval mibine ke maghsadesh aya tabaghe bala hast ya na
    if( (key_tabaghe_2==1) || (key_tabaghe_3==1) || (key_tabaghe_4==1)){
        harekat_fow();
    }
    //age bashe ke mire bala va age na, miad state ro taghir mide ke bere paein ro check kone
    else{
        harekat_state=2;
    }
}
// age harekat state be paein bashe, mire to shart zir
else if(harekat_state==2){
    //chek mikone bebine maghsad aya pen hast ya na?
    if( key_tabaghe_g==1){
        // harekat be andaze yek tabaghe be samte paein
        harekat_bck();
    }
    //agar bood ke mire paein vali age nabood taghire state mide
    else{
        harekat_state=1;
    }
}
}
//payene shart haye tabaghe aval
}
//age dar tabaghe dovom bashim varede shart zir mishe
```

پس در نتیجه وقتی متغیر طبقه +۱ می شود ما از شرط `tabaghe==0` خارج می شویم و وارد شرط بعدی می شویم. در شرط بعدی مطرح می کند که آیا طبقه مقصد ما طبقه یک است یا نه. حال اگر مقصد همان طبقه یک باشد: در آسانسور را با استفاده از حلقه ای که برای مدت ۵ ثانیه حدودا در نظر گرفتیم باز می کند و نیز برای باز شدن عدد متناظر صفر درجه برای درایور موتور را قرار می دهیم. دوباره شمارنده را صفر می کنیم، صفحه را رفرش می کنیم که آیا دکمه جدیدی فشار داده شده است یا نه و اطلاعات دمایی را در نمایشگر نمایش می دهیم فلگ بازشدن در که همان صفر بودن است را قرار می دهیم و فلگ طبقه مقصد که همان طبقه اول است را صفر قرار می دهیم و موتور را بی حرکت قرار می دهیم و ال ای دی طبقه اول را خاموش می کند. و اگر مقصد طبقه اول نباشد:

در ابتدا بررسی میکند که اگر استیت حرکت ۱ باشد یعنی رو به بالا باشد و اگر طبقه مقصد یکی از طبقات ۲، ۳ و ۴ باشد (آیا این طبقات زده شده اند یا نه)، تابع حرکت رو به بالا داشته باشد (یک طبقه رو به بالا رفته و فلگ طبقه را یک واحد افزایش دهد و فلگ استیت حرکت را ۱ می کند). اگر استیت حرکت رو به بالا نباشد و دکمه های بالا فشار داده نشده باشد در نتیجه حرکت رو به پایین است و مقدار ۲ را در آن قرار می دهد.

از آن جایی که در مثال طبقه مقصد ما ۳ بود، پس در نتیجه شرط اول اجرا شده و حرکت رو به بالا داشته، فلگ طبقه را یک واحد افزایش دهد و فلگ استیت حرکت را ۱ می کند و چون فلگ طبقه یک واحد افزایش داده می شود پس از حلقه `tabaghe==1` خارج شده و وارد استیت بعدی می شویم. حال همانند مراحل قبل استیت هارا بررسی میکند تا به طبقه مقصد می رسیم.

در این مرحله از برنامه، موردی را بررسی می کنیم که حرکت رو به بالا است، در طبقه دوم قرار داریم و فردی طبقه همکف را فشار می دهد. حال اینجا این مورد را بررسی می کند که طبقه های پایین تر از طبقه فعلی مقصد هستند یا خیر. اگر مقصد بودند حرکت رو پایین را اجرا میکند یعنی یک طبقه می کاهد، یک واحد فلگ طبقه را کاهش می دهد و فلگ استیت حرکت را برابر ۲ می کند. و اگر مقصد نبودند پس حرکت رو به بالا است و استیت حرکت را برابر ۱ قرار می دهد.

پس از بررسی استیت ها وقتی به مقصد مورد نظر رسیدیم همان استیتی که مقدارش را یک کرده بودیم (در این مثال key_tabaghe_3==1 بود) را مقدارش را برابر صفر قرار می دهیم در نتیجه تمام برنامه برمیگردد به حالتی که مقدار همه مقصد ها صفر بودند؛ یعنی مقصدی نداشته و منتظر فشار داده شدن کلید توسط کاربر هستیم تا از ابتدا شرط ها را چک کرده و به سمت مقصد حرکت کند و در اینجا اولویت حرکت به ترتیب برابر با طبقه ۴ و ۲ و ۱ است زیرا قبل از اینکه کار تمام شود استیت حرکت ۱ بود یعنی حرکت رو به بالا.

بهتر است این دو مورد را نیز بیان کرد که در طبقه همکف فقط حرکت رو به بالا بررسی می شود و در طبقه چهارم طبقه رو به پایین. چون وقتی برای مثال در طبقه همکف هستیم تنها مقصد ما میتواند طبقه های ۱، ۲، ۳ و ۴ باشد یعنی حتما حرکت رو به بالا خواهد بود و هرگاه برای مثال در طبقه چهارم باشیم تنها مقصد ما طبقه های ۱، ۲، ۳ و همکف می تواند باشد یعنی قطعاً حرکت رو به پایین است. در نتیجه تابع اولویت ما تمام می شود.

حال به بررسی تابع حرکت فوروارد harekat_fow (رو به بالا) می پردازیم:

```
//in tabe bareye harekat be samte bala be ezaye 1 tabaghe be kar miravad
void harekat_fow(void){
    // ebtada chek mikone bebine dar baze ya na
    if(((key_tabaghe_g=1) || (key_tabaghe_l=1) || (key_tabaghe_2=1) || (key_tabaghe_3=1) || (key_tabaghe_4=1)) && (door==0)){
        // age dar baz bashe bayed dar ro bebandim
        for(; m<300 ; m+=1){
            OCRA=300;//close
            //OCRA=187;//open
            key_check();
            door=1;// dar bashe mishe
            lcd_temp();
        }
        m=0;// counter sefr mishe
    }
    lcd_temp();
    //delay
    // dar inja be jaye dealy az yek fore estefade karde em
    //PORTB.6=1;
    for(; m<4000 ; m+=1){
        //second=((m*5)/3999)+1;
        sprintf(d,"%d",m);// shomarande ro minevise
        lcd_gotoxy(0,0);
        lcd_puts(d);
        key_check();// kelid ha dar harekat ham update mishe
        PORTC.5=1;PORTC.6=0;// ahrekat be bala
    }

    fan_heater();//heater va fan check shavad
    lcd_temp();
    m=0;// counter sefr mishe
    PORTC.5=0;PORTC.6=0;//motor stop
    // delay
    harekat_state=1;// state ra ro be bala taghir midahad
    tabaghe+=1; // shomare tabaghe ra ezafe mikonad
    //PORTB.6=0;
```

در این کد ابتدا بررسی می کند که در بسته است یا نه. یعنی شرط بسته بودن را وقتی آند می کند یعنی حتما باید در بسته باشد. یعنی در این شرط بررسی میکند که یکی از در ها فشار داده شده است و همزمان در باید بسته باشد. یعنی در اینجا در بسته است، مقصد مشخص شده در نتیجه نیاز به حرکت داریم. حال اگر در باز باشد، که در شرط اول آن را بررسی میکند یعنی اینکه اگر `door==0` این یعنی در باز بوده و باید بسته شود. در نتیجه وارد حلقه ای می شویم که در را برای حدودا ۵ ثانیه می بندد. صفحه را رفرش میکند که کلیدی زده شده است یا خیر سپس اطلاعات را در صفحه نمایش نشان می دهد و در را میبندد یعنی `door=1` و وقتی از حلقه خارج شد شمارنده را برابر صفر قرار می دهد و دوباره اطلاعات را بر روی نمایشگر نمایش می دهد. حال که در بسته شده است وارد حلقه دیگری می شویم که آسانسور را یک طبقه به بالا می برد. برای مدت تقریبا ۴ ثانیه آسانسور را به بالا می برد، شمارنده را به فرم دهدهی درون آرایه هایی قرار داده و با انتخاب مختصات اولیه ال سی دی، اطلاعات را در نمایشگر نشان می دهد و همزمان کلید هارا چک می کند که اگر فردی کلیدی دیگری را فشرد (مقصد دیگری را تعیین کرد) مشخص شود و موتور به صورت پادساعتگرد می چرخد و باعث می شود یک طبقه رو به بالا حرکت کنیم. در انتها دوباره شمارنده را صفر میکند و موتور را می ایستاند. `Harekat_state=1` قرار می دهد و فلگ طبقه را یک واحد افزایش می دهد. حال برنامه این تابع تمام می شود.

حال به بررسی تابع حرکت بکوارد `harekat_bck` (رو به پایین) می پردازیم:

```
//in tabe bareye harekat be sante paein be esaye 1 tabaghe be kar miravad
void harekat_bck(void){
    // aval check mikone ke dar baze ya baste
    if(((key_tabaghe_1==1) || (key_tabaghe_2==1) || (key_tabaghe_3==1) || (key_tabaghe_4==1)) && (door==0)){
        for(; m<300 ; m+=1){
            OCRA1A=300;//close
            //OCRA1A=187;//open
            key_check();
            door=1;// dar ro mibande
            lcd_temp();
            fan_heater();//heater va fan check shavad
        }
        door=1;// flag dar ro yek mikone
        m=0;// counter ro 0 mikone
    }
    lcd_temp();
    //delay
    // be jaye dealy az for estefade kardim
    for(; m<4000 ; m+=1){
        //second=(m*5)/3999+1;
        sprintf(d,"%d",m);// meghdar counter ro ro LCD minevise
        lcd_gotoxy(0,0);
        lcd_puts(d);
        key_check();
        PORTC.5=0;PORTC.6=1;// motor ro be bala
    }

    lcd_temp();
    fan_heater();//heater va fan check shavad
    m=0;// counter ro seft mikone
    PORTC.5=0;PORTC.6=0;
    // delay
    harekat_state=2;// sate ro ro be paein mikone
    tabaghe=1; // tabaghe ro yeki kam mikone
}
```

حرکت به سمت پایین مشابه حرکت به سمت بالا صورت می گیرد. دقیقا همانند حرکت رو به بالا ابتدا چک میکند که آیا در بسته است یا نه و اگر در باز بود آن را می بندد. و وقتی در را می بندد متغیر `door=1` قرار می دهد تا بفهمیم در قفل شده و در نهایت پس

از خارج شن از حلقه بستن در شمارنده را صفر می کند تا در شمارنده های بعدی از آن استفاده کنیم و در نهایت اطلاعات را در نمایشگر نشان می دهد.

حال در این مرحله از برنامه، با استفاد از یک حلقه برای مدت حدودا ۵ ثانیه یک طبقه آسانسور را به پایین منتقل می کند و برای همین موتور بصورت ساعتگرد گردش می کند. در نهایت دوباره پس از اتمام کار شمارنده را صفر می کند و موتور را متوقف می کند (برای اطمینان در نظر گرفتیم که هرگاه کار چرخش موتور تمام شد بایستد) و اطلاعات جدید را با فراخوانی تابع ال سی دی در نمایشگر نمایش می دهد و استیت حرکت را برابر ۲ قرار می دهد یعنی رو به پایین و فلگ طبقه را یک واحد کاهش می دهد.

توجه داریم که تابع فن و هیتر را درون هر دو تابع فروروارد و بک وارد (بالارونده و پایین رونده) قرار دادیم تا آسانسور در حین حرکت نیز بتواند دمای داخل کابین را تنظیم و در صورت نیاز تهویه کند.

حال به بررسی تابع fan_heater می پردازیم:

```
void fan_heater(void){
    temp=read_adc(0); // ebtada megdhar giri mikonad
    if(temp>25){PORTE.5=0;PORTE.6=1;} // age dama bala bashe fan ra roshan mikonad
    else if(temp<20) {PORTE.5=1;PORTE.6=0;} // age dama paein bashe heater ra roshan mikonad
    else {PORTE.5=0;PORTE.6=0;} // age garm beshe khamoosh mikone
    sos_key(); // check kardane kelide sos
}
```

تابع فن و هیتر چون دارای خروجی نیست یک تابع از نوع void تعریف شده است. در این تابع ابتدا گفته می شود ک مقدار دما توسط دماسنج خوانده شود. اگر دما بیشتر از ۲۵ درجه باشد فن را روشن کند (از یک موتور دی سی برای چرخش و فن استفاده کردیم که مستقیم به میکرو آن را متصل کردیم) و اگر دما از ۲۰ درجه کمتر باشد هیتر را که توسط یک رله به میکرو وصل کرده بودیم را فعال کند. در غیر اینصورت موتور دی سی (فن) و هیتر خاموش باشد و اگر کلید خطر فشار داده شود تابع S.O.S را فراخوانی می کند که در ادامه به توضیح آن خواهیم پرداخت.

حال به بررسی تابع sos_key می پردازیم:

```
void sos_key(void){
    if(PIND.5==0){
        PORTE.0=1;
        PORTE.1=1;
        PORTE.2=1;
        PORTE.3=1;
        PORTE.4=1;
        flag_sos=1;
    }
    else if(PIND.5==1){
        PORTE.0=0;
        PORTE.1=0;
        PORTE.2=0;
        PORTE.3=0;
        PORTE.4=0;
        flag_sos=0;
    }
}
```

این تابع به گونه ای برنامه ریزی شده است که اگر دکمه S.O.S واقع در داخل کابین فشرده شود، تمامی ال ای دی های هر طبقه به رنگ قرمز روشن شوند و اگر فشار داده نشود خاموش بمانند. و این تابع در در توابع دیگر فراخوانی کردیم تا در لحظه حرکت نیز امکان درخواست کمک برای فرد وجود داشته باشد.

نکات اضافه تر از دستور کار برای بهبود عملکرد آسانسور:

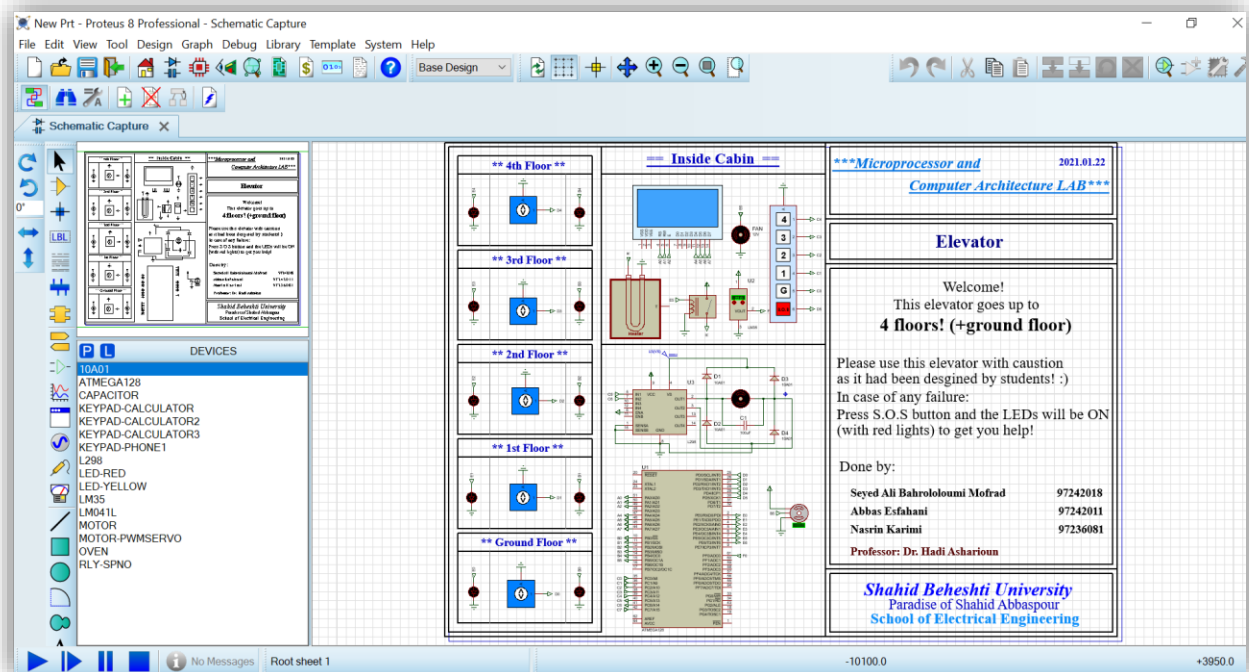
- تابع اولویت برای تشخیص طبقات نزدیک تر که بطور کلی در بخش معرفی و توضیح عملکرد توابع توضیح داده شد.
- قابلیت دریافت همزمان چند مقصد
- قرار دادن فن و سیستم تهویه و خنک کننده هوا
- قرار دادن هیتر برای گرم کردن هوای داخل کابین
- آپدیت شدن لحظه ای فن و هیتر
- قرار دادن دکمه وضعیت اضطراری برای کمک در مواقع خطر که عملکرد آن در لحظه امکان دارد؛ حتی حین حرکت!
- قرار دادن دماسنج داخل کابین
- استفاده از رله برای روشن و خاموش کردن هیتر
- قرار دادن ال ای دی هایی برای روشن شدن چراغ مقصد
- نمایش دما، باز بودن یا بسته بودن در و شمارنده در ال سی دی
- در پروتئوس کپد خاص این پروژه برای آسانسور (داخل و خارج کابین) طراحی شده است.

در نوشتن این برنامه تنها از PWM استفاده شده است که از آن برای باز و بسته شدن در آسانسور استفاده کردیم. در واقع PWM جزئی از تایمر بوده یعنی میکرو تایمری دارد که PWM جزئی از آن است. در واقع آن تولید پالس مربعی کرده و با توجه به دیوتی سایکلی که آن پالس دارد، سرو موتور به گونه ای طراحی شده است که برای مثال اگر آن دیوتی سایکل نصف باشد، در زاویه ۹۰ درجه می ایستد. یا برای مثال اگر ۰.۷۵ باشد در ۱۸۰ درجه قرار می گیرد.

طبق محاسباتی که برای این برنامه انجام شد برای قرار گرفتن در زاویه صفر درجه باید $OCR1A=187$ و زاویه ۹۰ درجه باید

$OCR1A=300$

نمای کلی پروتئوس:



کد اصلی پروژه :

```
ketabkhane haye morede niaz ra tarif mikonim //
#include <mega128.h#
#include <delay.h#
#include <alcd.h#
#include <stdio.h#
moteghayer haye morede niaz ra tarif mikonim //
int m=0; //counter
;int flag_sos=0
void key_check(void);// tabe chek kilid ha
int temp=0 ;// moteghayer haye dama
areye haye LCD //
;[ '\0']char b
;[ '\0']char c
;[ '\0']char d
;void sos_key(void)
int door=0;// flag dar
void olaviat(void);// tabe olaviat
void fan_heater(void);//tabe fan va heater
void harekat_bck(void); // ahrekat be paein
void lcd_temp(void); // LCD va namayeshe dama va baghie moteghayer
ha
void harekat_fow(void); // tabeh harekat ro be bala
```

```
;unsigned long int counter=0//  
int tabaghe=0; // state tabaghe ra moshakhas mikonad  
  
int  
key_tabaghe_g=0 ,key_tabaghe_1=0,key_tabaghe_2=0,key_tabaghe_3=  
0,key_tabaghe_4=0 ;// flag haye kilid
```

```
int harekat_state=1; // state harekate asansor ra moshakhas mikonad
```

Voltage Reference: Int., cap. on AREF //

```
define ADC_VREF_TYPE ((1<<REFS1) | (1<<REFS0) | #  
(1<<ADLAR))
```

Read the 8 most significant bits //

of the AD conversion result //

```
unsigned char read_adc(unsigned char adc_input)
```

```
}
```

```
;ADMUX=adc_input | ADC_VREF_TYPE
```

Delay needed for the stabilization of the ADC input voltage //

```
;(10)delay_us
```

Start the AD conversion //

```
;ADCSRA|=(1<<ADSC)
```

Wait for the AD conversion to complete //

```
;while ((ADCSRA & (1<<ADIF))==0)
```

```
;ADCSRA|=(1<<ADIF)
```

```
;return ADCH
```

```
{
```

```
void main(void)
```

```
}
```

```
;DDRB=0xFF
```

```
;DDRC=0xE0
```

```
;PORTC=0x1F
```

```
;DDRD=0x00
```

```
;PORTD=0xFF
```

```
;DDRE=0xFF
```

```
;DDRG=0xFF
```

```
Timer(s)/Counter(s) Interrupt(s) initialization //
```

```
TIMSK=(0<<OCIE2) | (0<<TOIE2) | (0<<TICIE1) | (0<<OCIE1A) |  
;(0<<OCIE1B) | (0<<TOIE1) | (0<<OCIE0) | (1<<TOIE0)
```

```
ADC initialization //
```

```
ADC Clock frequency: 500/000 kHz //
```

```
ADC Voltage Reference: Int., cap. on AREF //
```

```
Only the 8 most significant bits of //
```

```
the AD conversion result are used //
```

```
;ADMUX=ADC_VREF_TYPE
```



```
ADCSRA=(1<<ADEN) | (0<<ADSC) | (0<<ADFR) | (0<<ADIF) |  
;(0<<ADIE) | (1<<ADPS2) | (0<<ADPS1) | (0<<ADPS0)  
;SFIOR=(0<<ACME)
```

```
/* TCNT1 = 0;      /* Set timer1 count zero  
/* ICR1 = 2499;    /* Set TOP count for timer1 in ICR1 register
```

```
Set Fast PWM, TOP in ICR1, Clear OC1A on compare match, */  
/* clk/64
```

```
;TCCR1A = (1<<WGM11)|(1<<COM1A1)
```

```
;TCCR1B = (1<<WGM12)|(1<<WGM13)|(1<<CS10)|(1<<CS11)  
;OCR1A=187
```

```
Alphanumeric LCD initialization //
```

```
Connections are specified in the //
```

```
:Project|Configure|C Compiler|Libraries|Alphanumeric LCD menu //
```

```
RS - PORTA Bit 0 //
```

```
RD - PORTA Bit 1 //
```

```
EN - PORTA Bit 2 //
```

```

D4 - PORTA Bit 4 //
D5 - PORTA Bit 5 //
D6 - PORTA Bit 6 //
D7 - PORTA Bit 7 //
Characters/line: 16 //
;())lcd_init
asm("sei")#
Global enable interrupts //
Timer 0 overflow interrupt service routine //
interrupt [TIM0_OVF] void timer0_ovf_isr(void)//
{//
Reinitialize Timer 0 value ////
//
;TCNT0=0xB2//
Place your code here ////
;counter+=1 //
{//

////////////////////while////////////////////
}())while
;()lcd_temp
;()olaviat
;PORTB.6=1//

```

```
;()sos_key//
```

```
{
```

```
payane void //
```

```
{
```

```
////////////////////////tavabe////////////////////////
```

```
olaviat tabe manteghe aslie code mast //
```

```
}void olaviat(void)
```

```
key_check();//in tabe kelid ha ra check mikonad
```

```
sos_key();// check kardane kelide sos
```

```
if((key_tabaghe_g==0) && (key_tabaghe_1==0) &&  
(key_tabaghe_2==0) && (key_tabaghe_3==0) &&  
{(key_tabaghe_4==0))
```

```
asansor dar yek tabaghe stop mikone ta dokme bezanan //
```

```
sos_key();// check kardane kelide sos
```

```
{
```

```
dar paein shart haye harekat kardane asansor ra darim//
```

```
}else
```

```
baraye tabaghe g boodan varede shart zir mishavim //
```

```

}if(tabaghe==0)
agar tabagheye maghsad morede nazar g bashad, bayad dar baz //
shavad

}if(key_tabaghe_g==1)
inja bayad bege esme tabaghe ro tabaghe //
dar baz beshe va baste beshe//

}for(; m<500 ; m+=1)
OCR1A=300;//close//
OCR1A=187;//open
;()key_check
;()lcd_temp
door=0;//dar baz mishe
{
m=0;//shomarande sefr mishe
key_tabaghe_g=0;// flag tabaghe g bayad khamoosh beshe
harekat_state=1;// harekat state bayad be samte bala shavad
PORTB.0=0;// LED tabaghe g bayad khamoosh beshe
{
agar maghsad g nabashad varede shart zir mishavim//
}else
check mikone ke tabagh chandome maghsad //
if((key_tabaghe_1==1)|| (key_tabaghe_2==1)
)|| (key_tabaghe_3==1)|| (key_tabaghe_4==1))
harekat be andaze yek tabaghe be samte bala //

```

```

harekat_fow();// tabe harekat ro be bala baraye yek tabaghe
{
{
payane shart haye tabaghe G //
{
tabaghe 1 agar bashim code varede shart zir mishavad//
}else if(tabaghe==1)
age tabaghe 1 maghsad bashe varede shart zir mishe //
}elseif(key_tabaghe_1==1)
inja bayad bege  esme tabaghe ro tabaghe //
dar baz beshe va baste beshe//
}for(; m<500 ; m+=1)
OCR1A=300;//close//
OCR1A=187;//open
;()key_check
;door=0
;()lcd_temp
dar baz mishavad //{
m=0; // shomarande sefr mishe
key_tabaghe_1=0;// flag tabaghe aval sefr mishe
PORTC.5=0;PORTC.6=0;//motor bi harekat mishe
OCR1A=187;// dar baz mishe vali in khat tekrarie yadam
(: rafte pakesh konam

```

```

PORTB.1=0; //LED tabaghe aval khamoosh mishe
{
age maghsad tabaghe dige E bashe mire tu shart zir //
}else
check mikone bebine harekat state ro be balast ya paEn//
}if(harekat_state==1)
age be bala bood harekat, aval mibine ke maghsadesh aya //
tabaghe bala hast ya na
if( (key_tabaghe_2==1)
)||(key_tabaghe_3==1)||(key_tabaghe_4==1))
;()harekat_fow
{
age bashe ke mire bala va age na, miad state ro taghir //
mide ke bere paein ro check kone
}else
;harekat_state=2
{
{
age harekat state be paein bashe, mire to shart zir //
}else if(harekat_state==2)
?chek mikone bebine maghsad aya pen hast ya na//
}if( key_tabaghe_g==1)
harekat be andaze yek tabaghe be samte paein //
;()harekat_bck

```

```

{
agar bood ke mire paein vali age nabood taghire state mide//
}else
;harekat_state=1
{
{
{
payene shart haye tabaghe aval//
{
age dar tabaghe dovom bashim varede shart zir mishe//
}else if(tabaghe==2)
age naghsad tabaghe dovom baghe mire tu shart zir //
}if(key_tabaghe_2==1)
inja bayad bege esme tabaghe ro tabaghe //
dar baz beshe va baste beshe//
}for(; m<900 ; m+=1)
OCR1A=300;//close//
OCR1A=187;//open
;()key_check
door=0;// dar baz mishe
;()lcd_temp
{
m=0;//counter ro sefr mikone

```

```

key_tabaghe_2=0;// flag tabaghe 2 sefr mishe
PORTB.2=0;//LED tabaghe dovom khamoosh mishe
OCR1A=187; // dar baz mishe
{
hala age maghsad tabaghe 2 nabashe vali dar tabaghe dovom //
bashim
}else
chek mikone bebine state chie //
}if(harekat_state==1)
age state bala bood va maghsad tabaghe bala bood, mire //
to shart zir
}if( (key_tabaghe_3==1) || (key_tabaghe_4==1) )
harekat be andaze yek tabaghe be samte bala //
;)harekat_fow
{
age na, state ro 2 mikone ke harekat ro be paein check //
baeshe
}else
;harekat_state=2
{
{
age harekat ro be paein bashe,mire shart zir//
}else if(harekat_state==2)
age maghsad paen bashe mire paEn//

```



```
}if( (key_tabaghe_1==1) || (key_tabaghe_g==1))  
harekat be andaze yek tabaghe be samte paein //  
;()harekat_bck  
{  
age na state ro 1 mikone ke bala raftan check beshe //  
}else  
;harekat_state=1  
{  
{  
{  
shaert haye tabaghe 2 tamoom shod //  
{  
age to tabaghe 3 bashim mire to shart zir //  
}else if(tabaghe==3)  
age maghsad khode tabaghe 3 bashe //  
}if(key_tabaghe_3==1)  
inja bayad bege esme tabaghe ro tabaghe //  
dar baz beshe va baste beshe//  
}for(; m<500 ; m+=1)  
OCR1A=300;//close//  
OCR1A=187;//open  
;()key_check  
door=0;// dar baz mishe
```

```

;()lcd_temp
{
m=0;//counter sefr mishe
key_tabaghe_3=0; // fals tabaghe sefr mishe
PORTB.3=0;//LED  tabaghe 3 sefr mishe
OCR1A=187; // dar baz mishe
{
}else
,hala age maghdas tabaghe 3 nabashe //
}if(harekat_state==1)
age state bala bashe miad tabaghe haye bala ro chek //
mikone
}if(key_tabaghe_4==1)
;()harekat_fow
{
age na miad state ro 2 mikone //
}else
;harekat_state=2
{
{
age state 2 bashe miad tabaghe haye paein ro check mikone //
}else if(harekat_state==2)
age maghsad paen bashe ke mire paein //

```

```
if( (key_tabaghe_2==1) || (key_tabaghe_1==1) ||  
  )(key_tabaghe_g==1))  
harekat be andaze yek tabaghe be samte paein //  
;())harekat_bck  
{  
age na miad chek mikone state 2 ro//  
}else  
;harekat_state=1  
{  
{  
{  
payene shart haye tabaghe 3 //  
{  
age to tabaghe 4 bashim shart zir ejra mishavad //  
}else if(tabaghe==4)  
age maghsad tabaghe 4 bashe mire shart zir //  
}if(key_tabaghe_4==1)  
inja bayad bege esme tabaghe ro tabaghe //  
dar baz beshe va baste beshe//  
}for(; m<500 ; m+=1)  
OCR1A=300;//close//  
OCR1A=187;//open  
;())key_check
```

```

;()lcd_temp
door=0;// dar baz mishe
{
m=0; // shomarande sefr mishe
key_tabaghe_4=0;// flag sefr mishe
PORTB.4=0;//LED roshan, khamoosh mishe baraye tabaghe4
harekat_state=2;// state harekat 2 mishe ke bere faghat pein
dige
OCR1A=187; // dar baz mishe
{
chon tamame tabaghat ghaatan paein tar az 4 hastan, bayad //
tabaghe haye paein tar ro bayad check kone
else if((key_tabaghe_3==1) || (key_tabaghe_2==1) ||
)(key_tabaghe_1==1) || (key_tabaghe_g==1))
harekat be andaze yek tabaghe be samte paein //
;()harekat_bck
{
{
payane shart haye tabaghe 4//
payane shart haye harekat dar tabaghat //
{
payane tabe olaviate harekat //
{

```

```

in tabe etelaate LCD ra update mikonad //
}()void lcd_temp
temp=read_adc(0);// adc ra mikhanad va dar moteghayer mirizad
sprintf(b,"%d",temp);// moteghayer ra be char tabdil mikonad
;sprintf(a,"%d",j)//
sprintf(c,"%d",tabaghe);//moteghayer ra be char tabdil mikonad
;sprintf(d,"%d",counter)//
;(·'·)lcd_gotoxy//
;lcd_puts("count:")//
;lcd_puts(d)//
lcd_gotoxy(0,1);// dar khat 0 1 minevisad
lcd_puts("temp= ");// ebarete temp: ra chaap mikone
lcd_gotoxy(0,7);// dar khat 0 7 minevisad
lcd_puts(b);// meghdar temp ro minevise
lcd_gotoxy(0,2);// dar khat 0 2 minevisad
lcd_puts("tabaghe= ");// ebarete tabaghe: ra chaap mikone
lcd_puts(c); // meghdare tabaghe ra minevise
lcd_gotoxy(0,3);// dar khat 0 3 minevisad
;lcd_puts("harekat= ")
if(harekat_state==1){lcd_puts("fow");}// bala ye paen raftene asansor
ra update mikonad
if(harekat_state==2){lcd_puts("bak");}
lcd_gotoxy(11,2);// dar khat 0 1 minevisad

```

```

if(door==1){lcd_puts("cols");} // baz ya baste shodane dar ra update
mikonad

if(door==0){lcd_puts("open");}

;()fan_heater

{

in tabe baraye fan va hiter ast ke ba tavajohi be meghdare dama an ha //
ra tanzim mikonad

}void fan_heater(void)

temp=read_adc(0);// ebteda meghdar giri mikonad

if(temp>25){PORTE.5=0;PORTE.6=1;} // age dama bala bashe fan
ra roshan mikonad

else if(temp<20) {PORTE.5=1;PORTE.6=0;}// age dama paein
bashe heater ra roshan mikonad

else {PORTE.5=0;PORTE.6=0;}// age garm beshe khamoosh mikone

sos_key();// check kardane kelide sos

{

in tabe kelid ha ro chek mikone va flag ha va LED ha ra update //
mikone

}void key_check(void)

age kelidi zade beshe ke shamele tabaghe E ke tooshim nashe, flag //
va LED ro roshan mikone

sos_key();// check kardane kelide sos

}if(((PINC.0==0)||(PIND.0==0)) && (tabaghe!=0) )

```

```
key_tabaghe_g=1;//falg ro yek mikone
PORTB.0=1;// LEd tabaghe ro roshan mikone
{

}if(((PINC.1==0)||(PIND.1==0)) && (tabaghe!=1))
key_tabaghe_1=1;//falg ro yek mikone
PORTB.1=1;// LEd tabaghe ro roshan mikone
{
}if(((PINC.2==0)||(PIND.2==0)) && (tabaghe!=2))
key_tabaghe_2=1;//falg ro yek mikone
PORTB.2=1; // LEd tabaghe ro roshan mikone
{
}if(((PINC.3==0)||(PIND.3==0)) && (tabaghe!=3))
key_tabaghe_3=1; //falg ro yek mikone
PORTB.3=1; // LEd tabaghe ro roshan mikone
{
}if(((PINC.4==0)||(PIND.4==0)) && (tabaghe!=4))
key_tabaghe_4=1; //falg ro yek mikone
PORTB.4=1; // LEd tabaghe ro roshan mikone
{

{
```

in tabe bareye harekat be samte bala be ezaye 1 tabaghe be kar miravad//

```
}void harekat_fow(void)
```

ebteda chek mikone bebine dar baze ya na //

```
if(((key_tabaghe_g==1) || (key_tabaghe_1==1) ||  
(key_tabaghe_2==1) || (key_tabaghe_3==1) ||  
(key_tabaghe_4==1))&&(door==0))
```

age dar baz bashe bayed dar ro bebandim //

```
}for(; m<300 ; m+=1)
```

```
OCR1A=300;//close
```

```
OCR1A=187;//open//
```

```
;()key_check
```

```
door=1;// dar bashe mishe
```

```
;()lcd_temp
```

```
{
```

```
m=0;// counter sefr mishe
```

```
{
```

```
;()lcd_temp
```

```
delay//
```

dar inja be jaye dealy az yek fore estefade karde em //

```
;PORTB.6=1//
```

```
}for(; m<4000 ; m+=1)
```

```
;second=((m*5)/3999)+1//
```

```
sprintf(d,"%d",m);// shomarande ro minevise
```

```
;(·'·)lcd_gotoxy
```



```

;lcd_puts(d)
key_check();// kelid ha dar harekat ham update mishe
PORTC.5=1;PORTC.6=0;// ahrekat be bala

{
fan_heater();//heater va fan check shavad
;()lcd_temp
m=0;// counter sefr mishe
PORTC.5=0;PORTC.6=0;//motor stop
delay //
harekat_state=1;// state ra ro be bala taghir midahad
tabaghe+=1; // shomare tabaghe ra ezafe mikonad
;PORTB.6=0//

{

in tabe bareye harekat be samte paein be ezaye 1 tabaghe be kar //
miravad
}void harekat_bck(void)
aval check mikone ke dar baze ya baste //
if(((key_tabaghe_g==1) || (key_tabaghe_1==1) ||
(key_tabaghe_2==1) || (key_tabaghe_3==1) ||
}(key_tabaghe_4==1))&&(door==0))
}for(; m<300 ; m+=1)

```

```

OCR1A=300;//close
OCR1A=187;//open//
;()key_check
door=1;// dar ro mibande
;()lcd_temp
fan_heater();//heater va fan check shavad
{
door=1;// flag dar ro yek mikone
m=0;// counter ro 0 mikone
{
;()lcd_temp
delay//
be jaye dealy az for estefade kardim //
}for(; m<4000 ; m+=1)
;second=((m*5)/3999)+1//
sprintf(d,"%d",m);// meghdar counter ro ro LCD minevise
;(·'·)lcd_gotoxy
;lcd_puts(d)
;()key_check
PORTC.5=0;PORTC.6=1;// motor ro be bala

{
;()lcd_temp

```

```
fan_heater();//heater va fan check shavad
m=0;// counter ro seft mikone
;PORTC.5=0;PORTC.6=0
delay //
harekat_state=2;// sate ro ro be paein mikone
tabaghe-=1; // tabaghe ro yeki kam mikone
{
led sos//
}void sos_key(void)
}if(PIND.5==0)
;PORTE.0=1
;PORTE.1=1
;PORTE.2=1
;PORTE.3=1
;PORTE.4=1
;flag_sos=1
{
}else if(PIND.5==1)
;PORTE.0=0
;PORTE.1=0
;PORTE.2=0
;PORTE.3=0
;PORTE.4=0
```

```
;flag_sos=0
```

```
{
```

```
{
```

```
////////////////////////////////////
```