

به نام بی نام او

فاز دوم پروژه درس ریزپردازنده ها و زبان اسمبلی

پیاده سازی پروژه خانه هوشمند

ندا صدری (4029036099) - سپهر عباسپور (4029036082)

استاد ضرابی

زمستان 1402

مقدمه:

با توجه به اینکه هر دو نفر ما در فاز اول پروژه بر روی خانه های هوشمند ایده پردازی کرده بودیم، در این فاز نیز، طبق دستور العمل پروژه، برای پیاده سازی از همین مورد کمک می گیریم و هدفمان، پیاده سازی سیستمی است که اتوماسیون یک خانه هوشمند را مدیریت و کنترل می کند. مثلاً لامپ ها و انواع وسایل الکترونیکی را خاموش و روشن میکند.

راه های مختلف و زیادی برای کنترل دستگاه ها (لوازم خانگی و ...) وجود دارد. به طور کلی میشود آن ها را به دو دسته تقسیم کرد: سیمی و بی سیم.

برای مثال در ارتباط بی سیم، ما سیگنال ها را با ارتباط رادیویی RF و ... انتقال میدهیم و در سیمی، از سیم استفاده میکنیم.

- **ارتباط موازی:** در این نوع ارتباط ما بسته به اندازه داده ها در پین ها از سیم های زیادی استفاده میکنیم.

- **ارتباط سریال:** در ارتباط سریالی ما فقط از 2 سیم برای انتقال و دریافت داده ها استفاده میکنیم زیرا داده ها در این نوع ارتباط به صورت سریال به صورت بیتی ارسال میشوند.

نظر به اینکه گروه ما دو نفره است، با توجه به گفتگو و حسن توجه استاد، بر آن شدیم تا بدون کسر نمره، از بخش پیاده سازی عملی گذر کنیم و در اینجا، با توجه به علاقه بیشتر، در جریان جستجو های متوالی مان با مفاهیمی آشنا شدیم که در پیاده سازی سخت افزاری نیز به کار می آیند و به طور خلاصه و ضمنی به آن ها نیز اشاره می کنیم.

برای پیاده سازی سخت افزاری پروژه می توان از 3 لامپ صفر وات استفاده کرد که هر کدام نشان دهنده یک لامپ تلویزیون و فن هستند. همچنین می توان از آردوینو UNO به عنوان کنترل کننده این پروژه استفاده کرد.

قطعات مورد نیاز پروژه خانه هوشمند به کمک آردوینو و برد های آن:

- برد آردوینو UNO
- کابل سریال
- ترانزیستور ULN2003

- ULN2003
- رله 5 ولت
- لامپ و نگهدارنده
- ال سی دی کاراکتری 16*2
- کامپیوتر

پروتوتایپ کامل مدار در فایل ضمیمه قرار داده شده است که در آن، از یک ماژول LCD 16×2 برای نمایش وضعیت لوازم خانگی استفاده می کنیم که مستقیماً در حالت 4 بیتی به آردوینو متصل می شود. پین های داده ال سی دی یعنی RS، EN، D4، D5، D6، D7 به پین دیجیتالی Arduino شماره 7، 6، 5، 4، 3، 2 وصل می شوند. (توضیحات بیش تر درانتهای همین گزارش، پیوست گردیده است.)

برای ارسال دستورات به آردوینو از لپ تاپ یا کامپیوتر از کابل USB استفاده می کنیم. و یک درایور رله IC ULN2003 نیز برای درایو کردن رله ها استفاده می شود. رله های 5 ولت 3 SPDT برای کنترل لامپ، فن و تلویزیون استفاده می شوند و رله ها برای کنترل لامپ، فن و تلویزیون به ترتیب به پین شماره 3، 4 و 5 از طریق درایور رله IC ULN2003 وصل می شوند.

لازم به ذکر است که رله یک سوئیچ است که برای اتصال الکترونیک به برق استفاده میشود. رله ها دارای سیم پیچ و بعضاً هسته های سوئیچینگ دارند. در این پروژه رله ی ما دارای 5 پین است که 2 پین برای سیم پیچ میباشد. 1 پین برای قطب و 2 پین دیگر "معمولاً متصل" و "معمولاً باز" هستند.

نحوه کار پروژه:

در این پروژه ، ما به تعداد و نام های دلخواه دستگاه تعریف میکنیم. تابع و روتین تعریف دستگاه در قسمت توضیحات کد آورده شده است. در اینجا از ارتباط سریال برای کنترل وسایل منزل استفاده می شود. ما برای کنترل وسایل خانگی AC دستوراتی را ارسال می کنیم. پس از دریافت دستورات داده شده. آردوینو دستور را به رله هایی ارسال می کند که وظیفه روشن یا خاموش کردن وسایل را بر عهده دارند.

توضحات کد:

توضیحات کد زیر را به 4 بخش اصلی تقسیم میکنیم:

بخش های قرمز رنگ، پروتوتایپ کد هستند و صرفاً برای پاره ای از توضیحات قرار داده شده اند و کد اصلی در ادامه آن ها قرار داده شده است.

```
#include <LiquidCrystal.h>
#include <string.h>
LiquidCrystal lcd(13, 12, 11, 10, 9, 8);

#define light 3
#define fan 4
#define TV 5

char temp;
char str[10];
char i = 0;

void setup() {
    lcd.begin(16, 2);
    Serial.begin(9600);
    pinMode(light, OUTPUT);
    pinMode(fan, OUTPUT);
    pinMode(TV, OUTPUT);
    lcd.print(" Home Automation ");
    lcd.setCursor(0, 1);
    lcd.print("      Using PC      ");
    delay(2000);
    lcd.clear();
```

```

lcd.print("Keywords For ");
lcd.setCursor(0, 1);
lcd.print("Controlling");
delay(2000);
lcd.clear();
lcd.print("1. LIGHT ON");
lcd.setCursor(0, 1);
lcd.print("2. LIGHT OFF");
delay(2000);
lcd.clear();
lcd.print("3. FAN ON");
lcd.setCursor(0, 1);
lcd.print("4. FAN OFF");
delay(2000);
lcd.clear();
lcd.print("5. TV ON");
lcd.setCursor(0, 1);
lcd.print("6. TV OFF");
delay(2000);
default();
delay(1000);
}

```

1. Set up function:

- صفحه نمایش LCD و ارتباط سریال را راه اندازی می کند.
- پین های دیجیتال نور تلویزیون، فن یا سایر دستگاه های الکتریکی را به عنوان خروجی تنظیم می کند.
- نمایش یک سری پیام بر روی LCD برای نشان دادن کنترل های اتوماسیون خانگی.

```

void loop() {
    if (temp == 1) {
        // Commands for controlling fan, light, and TV
        // ...
    }
}

void loop() {
    if (temp == 1) {
        str[i] = '\0'; // Null-terminate the string
        // Fan control
    }
}

```

```

    if (strncmp(str, "FAN ON", 6) == 0) {
        digitalWrite(fan, HIGH);
        lcd.clear();
        lcd.print("Fan Turned On");
    } else if (strncmp(str, "FAN OFF", 7) == 0) {
        digitalWrite(fan, LOW);
        lcd.clear();
        lcd.print("Fan Turned Off");
    }
    // Light control
    else if (strncmp(str, "LIGHT ON", 8) == 0) {
        digitalWrite(light, HIGH);
        lcd.clear();
        lcd.print("Light Turned On");
    } else if (strncmp(str, "LIGHT OFF", 9) == 0) {
        digitalWrite(light, LOW);
        lcd.clear();
        lcd.print("Light Turned Off");
    }
    // TV control
    else if (strncmp(str, "TV ON", 5) == 0) {
        digitalWrite(TV, HIGH);
        lcd.clear();
        lcd.print("TV Turned On");
    } else if (strncmp(str, "TV OFF", 6) == 0) {
        digitalWrite(TV, LOW);
        lcd.clear();
        lcd.print("TV Turned Off");
    } else {
        lcd.clear();
        lcd.print("Invalid Input");
        lcd.setCursor(0, 1);
        lcd.print("Try Again");
    }
    delay(3000);
    defaultDisplay();
}
}

```

2 . Main Loop:

- ورودی کاربر را برای کنترل دستگاه هایی مانند نور، فن و تلویزیون بررسی می کند.
- رشته ورودی را با دستورات از پیش تعریف شده مقایسه می کند و دستگاه مربوطه را بر اساس آن عمل می کند.
- وضعیت دستگاه را روی LCD کامپیوتر نمایش می دهد.

```
void serialEvent() {  
    // Serial event processing  
    // ...  
}  
void serialEvent() {  
    while (Serial.available()) {  
        char Inchar = Serial.read();  
        if (i < sizeof(str) - 1) { // Prevent buffer  
overflow  
            str[i++] = Inchar;  
            lcd.print(Inchar);  
        }  
        if (Inchar == '\n') { // End of command  
            temp = 1;  
            i = 0; // Reset the index for the next command  
        }  
    }  
}
```

3.Serial Event Handling:

- داده های سریال ورودی را برای ساخت رشته کامند پردازش می کند.
- هر کاراکتر دریافتی را روی LCD نمایش می دهد.

```
void default() {  
    // Default LCD display setup  
    // ...  
}  
void defaultDisplay() {  
    lcd.clear();  
}
```

```

    lcd.print("Enter Your Choice:");
    lcd.setCursor(0, 1);
    lcd.cursor();
    memset(str, 0, sizeof(str)); // Clear the buffer
    temp = 0;
}

```

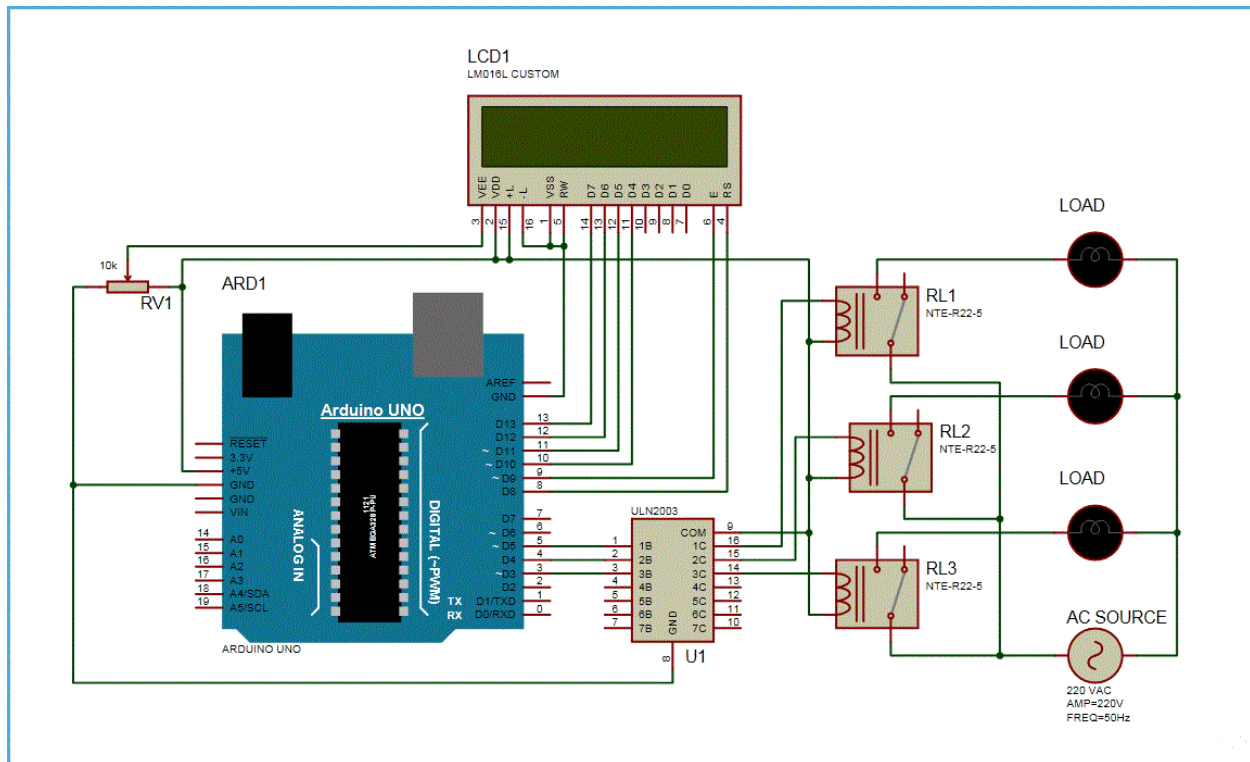
4.Default LCD Display Setup:

- ال سی دی صفحه نمایش را پاک می کند و آن را طوری تنظیم می کند که یک درخواست برای ورودی کاربر نمایش دهد.

در ادامه مواردی که در اجرای بهینه تر کد میتواند به ما کمک کند را در نظر می گیریم:

- مطمئن شویم که رشته str پیش از استفاده از متود strcmp، به طور کامل null شده باشد.
- ارور هندلری را برای موقعیت های خاصی مثل سرریز در بافر (buffer overflow) یا ورودی های سریالی توابع در نظر بگیریم.
- برای جلوگیری از ایجاد ابهام، می توانیم نام تمامی توابع default را به عناوینی دیگر تغییر دهیم.(مثل defaultDisplay)
- تاثیر داکيومنت کردن کد و کامنت گذاری را هرگز فراموش نکنیم:))

تصویری از شماتیک پروتوتایپ مدار و توضیحات آن که به دلیل علاقه شخصی، جستجو کردیم و به موارد زیر که در اشاره می شود، درباره آن پی بردیم:



این نمودار، یک شماتیک الکتریکی برای یک سیستم است که از میکروکنترلر Arduino UNO (ARD1) برای کنترل بارهای AC از طریق رله های RL1, RL2, RL3 استفاده می کند که تلاش کرده ایم یک توضیح کلی از اجزای اصلی و نحوه عملکرد آنها با یکدیگر ارائه کنیم:

1. Arduino UNO (ARD1):

این کنترل کننده مرکزی است که با استفاده از داده های ورودی از سنسورها یا رابط ها برنامه ریزی می شود تا رله ها را کنترل کند. میکروکنترلر مغز این سیستم است. این ورودی ها را دریافت می کند، آنها را مطابق با منطق برنامه ریزی شده پردازش می کند و خروجی ها را ارسال می کند. پین های دارای برچسب "Digital (PWM ~)" پین های دیجیتالی هستند که می توانند یک سیگنال دیجیتال HIGH یا LOW خروجی دهند و برخی از آنها قادر به مدولاسیون پهنای پالس (PWM) هستند. پین های "Analog In" می توانند سطح ولتاژ را از سنسورها بخوانند تا به عنوان ورودی آنالوگ تفسیر شوند.

2. LCD (LCD1):

از Arduino به متصل است و احتمالاً برای نمایش وضعیت بارها یا سایر اطلاعات استفاده می شود. این نمایشگر به برای نمایش داده هایی مانند وضعیت سیستم یا اندازه گیری استفاده می شود. متصل شده است. D9 تا D2 طریق پین های دیجیتال

برای حالت های (E، RW، RS) و خطوط کنترل (VEE)، تنظیم کنتراست (VDD، VSS) همچنین شامل اتصال برق داده/دستور و عملیات خواندن/نوشتن است.

3. پوتانسیومتر (RV1):

این یک مقاومت متغیر است که احتمالاً برای تنظیم پارامتر کنترلی مانند سطح آستانه برای تغییر رله ها استفاده می شود.

4. ULN2003 (U1):

این یک آرایه هفت کاناله Darlington است که برای تقویت جریان از پین های خروجی Arduino تا سطح مورد نیاز برای هدایت رله ها استفاده می شود. - هر کانال یک خروجی دیجیتالی Arduino را به یک سیم پیچ رله متناظر متصل می کند. این به عنوان یک تقویت کننده جریان عمل می کند، به طوری که خروجی های جریان پایین آردوینو می توانند سیم پیچ های رله با جریان بالاتر را کنترل کنند. پین COM به ولتاژ منبع تغذیه مورد نیاز برای رله ها متصل است.

5. رله ها (RL1، RL2، RL3):

اینها سوئیچ های الکتریکی هستند که به Arduino اجازه می دهند بارهای AC با توان بالا را با استفاده از یک Arduino بتوان پایین کنترل کنند. هر رله به یک خروجی از ULN2003 متصل است. هر رله دارای یک سیم پیچ (بخش آهنربای الکتریکی) است که توسط ULN2003 کنترل می شود. هنگامی که سیم پیچ فعال می شود، تماس ها را می بندد، اجازه می دهد جریان از منبع AC به LOAD جریان یابد. از رله ها استفاده می شود زیرا آردوینو نمی تواند به طور مستقیم بارهای AC ولتاژ بالا را تغییر دهد.

6. AC بارهای:

اینها دستگاه ها یا سیستم هایی هستند که توسط رله ها کنترل می شوند. آنها توسط یک منبع تغذیه AC که طبق نمودار VAC 220 با فرکانس 50 هرتز است، تغذیه می شوند.

7. AC منبع تغذیه:

منبع تغذیه بارها است، نه برای Arduino یا ULN2003

نمودار نشان می دهد که وقتی Arduino یک سیگنال را خروجی می دهد، از طریق ULN2003 عبور می کند که سپس یکی از رله ها را فعال می کند. هنگامی که یک رله فعال شود، مداری را که اجازه می دهد جریان AC به بار مربوطه جریان یابد، می بندد و آن را روشن می کند.

در عملیات واقعی ، کاربر ممکن است پوتانسیومتر (RV1) را بچرخانید تا سطح مورد نظر خود را تنظیم کند. Arduino این سطح را از A0 می خواند ، آن را پردازش می کند و بر اساس منطق برنامه خود ، ممکن است سیگنالی را از یکی از پین های دیجیتال خود به ULN2003 ارسال کند. ULN2003 این سیگنال را تقویت می کند تا سیم پیچ رله را هدایت کند که مدار را می بندد و اجازه می دهد انرژی AC به بار متصل جریان یابد و آن را روشن یا خاموش کند. وضعیت عملیات را می توان روی صفحه نمایشگر LCD مانند "بار 1 روشن" یا "بار 1 خاموش" نمایش داد.