آزمایش سوم

# مقدمه

در این آزمایش به یادگیری کار با استپر موتور و همچنین روشی برای هم‌زمان‌سازی کارها در میکروکنترلر و طراحی یک ساعت خواهیم پرداخت. فایل کتابخانه‌های مورد نیاز در این آزمایش در یک فایل Zip در صفحه درس قرار داده شده است.

* ساده‌ترین روش برای هم‌زمان‌سازی کارها در میکروکنترلرها استفاده از حلقه بی‌نهایت و انجام به نوبت کارهاست.:

while(true){

Task1();

Task2();

Task3();

}

* نکته بسیار مهم در هم‌زمان‌سازی این است که هیچ‌کدام از کارها نباید ایجاد مسدودیت کنند! در صورتی که کاری نیاز به مسدودیت داشته باشد، می‌بایست با شرط از انسداد میکرو جلوگیری کرد:

while(true){

if (Task1\_Occurs())

Task1();

if (Task2\_Occurs())

Task2();

if (Task3\_Occurs())

Task3();

}

* برای استفاده از کتابخانه‌ها، هنگام انتخاب Run-Time Env. فقط CMSIS-CORE را انتخاب کنید. تمام فایل‌ها را در محیط ویندوز در کنار فایل main.c خود قرار دهید. فایل‌های .c و .s را در محیط برنامه keil به کنار فایل main.c واقع در Source Group 1 اضافه کنید. (کلیک راست روی فولدر، Add existing Files…)
* فراموش نکنید که در ابتدای برنامه خود تابع SystemInit() را فراخوانی کنید.
* حتما فایل‌های کتابخانه‌ها را مطالعه کنید و با نحوه کار آن‌ها آشنا شوید.

# پیش‌آزمایش

* نحوه راه‌اندازی استپر موتور (Stepper Motor) به چه صورت است؟
* تفاوت Stepper Motor، DC Motor و Servo Motor چیست؟
* تفاوت اجرای موازی (Parallel) و هم‌زمان (Concurrent) کارها (Tasks) چیست؟
* چه قسمت‌هایی در میکروکنترلرها موجب مسدود شدن (Blocking) برنامه می‌شوند؟

# صورت آزمایش

## اتصالات

اتصالات روی بورد در این آزمایش به شرح زیر است:

* اتصال پایه‌های LCD‌ (RS,RW,E) به P0.0..P0.2
* اتصال پایه‌های LCD‌ DB4..DB7 به P0.4..P0.7
* اتصال فازهای Stepper Motor (A,B,A’,B’) به P2.10..P2.13
* اتصال سگمنت‌های 7-Seg (A-B-C-D-E-F-G-dot) به P0.15..P0.22
* اتصال کاتد هر یک از رقم‌ها (چهار رقم دارد) به P0.27..P0.30
  + (توجه کنید که 7-Seg کاتد مشترک است ولی یک NOT سر راهش وجود دارد. پس اگر می‌خواهید عدد را روشن کنید لازم است به مشترک‌ها یک بدهید.)

## قسمت اول

در این قسمت Stepper Motor را با سرعت ثابت و دلخواهی بچرخانید. (این موتور دارای زاویه چرخش ۱۵ درجه است)

برنامه نوشته شده به همراه فیلمی از چرخش یک دور کامل موتور را در گزارش خود قرار دهید.

## قسمت دوم

عقربه Stepper Motor را مانند عقربه ساعت‌شمار ساعت‌ آنالوگ و کلاسیک روی ساعت حضور خود در آزمایشگاه تنظیم کنید. تصویر مقابل ساعت ۴ یا 16 را نشان می‌دهد.

ساعت حضور، برنامه نوشته شده و تصویر موتور را در گزارش خود قرار دهید.

## قسمت سوم

یک شمارنده دو قسمتی طراحی کنید که هر قسمت با فرکانس متفاوتی کار می‌کند. هر قسمت از این شمارنده را در یک خط از LCD نمایش دهید. فرکانس هر قسمت از طریق فرمول زیر که در آن g معادل شماره گروه‌تان است، محاسبه می‌شود:

محاسبات تاخیرها، برنامه نوشته شده و یک فیلم کوتاه (حدود ۱دقیقه) از نمایشگر را در گزارش خود قرار دهید.

## قسمت چهارم (۵۰ درصد امتیازی)

به کمک قسمت‌های قبل و همچنین آزمایش قبل، ساعتی طراحی کنید که هر ساعت آن، یک ثانیه واقعی باشد. این ساعت را همزمان روی Stepper Motor، LCD و 7-Segment به نحوه زیر نمایش دهید. توجه کنید که نیازی به پیاده‌سازی دقیقه‌شمار نیست. همچنین نیازی به پیاده‌سازی ساعت بعدازظهر(PM) نیست؛ یعنی ساعت بعد از ۱۲ مجددا به ۱ برمی‌گردد:

* در LCD ساعت را به همراه «دو نقطه چشمک‌زن میانی» که یک ثانیه روشن و یک ثانیه خاموش است، نمایش دهید.
* در 7-Segment دو رقم سمت راست دقیقه (همیشه صفر) و دو رقم سمت چپ ساعت (متغیر) خواهد بود.
* در Stepper Motor مشابه قسمت دوم همانند یک ساعت آنالوگ ساعت را نمایش دهید.

توجه کنید که هر سه قسمت فوق می‌بایست همزمان اجرا شده و همه یک ساعت را نشان دهند. همچنین چون به دست آوردن مبدأ در Stepper Motor کار دشواری است، مبدأ (ساعت ۱۲) را دلخواه در نظر بگیرید؛ اما گام‌ها (Steps) می‌بایست صحیح و دقیقاً به اندازه زاویه‌ی یک ساعت (۳۰درجه) باشند.

برنامه نوشته شده به همراه یک فیلم کوتاه (حدود ۱دقیقه) از عملکرد تمام بخش‌ها را در گزارش خود قرار دهید.

موفق باشید