

## به نام خدا

### بخش اول

در محیط نرم افزار Keil uVision برنامه ای به زبان اسمبلی برای میکروکنترلر STM32F407VGT6 شرکت STMicoelectornic بنویسید که فرآیند تشخیص لبه را در یک تصویر انجام دهد.

به طور کلی به هرگونه تغییر ناگهانی در رنگ یا میزان روشنایی تصویر، یک لبه گفته می شود. لبه های هر تصویر، مرز یک جسم با سایر اجسام یا مرز جسم با محیط را نشان می دهد و مبنای بسیاری از الگوریتم های پردازش پیچیده تر می باشد. هر تصویر از کنارهم قرار گرفتن تعداد زیادی پیکسل های رنگ تشکیل می شود که مقدار هر پیکسل، میزان و نوع رنگ آن را مشخص می کند. در این پروژه تصویر به صورت سطح خاکستری فرض شده است. یعنی تصویری که در آن رنگ پیکسل ها تنها سیاه یا سفید یا مقداری بین دو است. ( تصویر شامل رنگ های دیگر نمی شود.) مقادیر پیکسل ها در این تصاویر به صورت ۸ بیتی و بین صفر تا ۲۵۵ است. در زیر نمونه هایی از تصاویر سطح خاکستری دیده می شود.



فرآیند تشخیص لبه به این صورت است که مقادیر پیکسل های مجاور با هم مقایسه می شوند و در صورتیکه اختلاف آن ها از یک حد آستانه بیشتر باشد به عنوان لبه در نظر گرفته می شوند. تصویری که در آن لبه ها معین

## پروژه پایانی درس معماری کامپیوتر

شده‌اند، تصویری است با دو مقدار، صفر و یک (سفید و سیاه- تصویر باینری). در صورتی که لبه تشخیص داده شده باشد، سیاه (مقدار صفر) و در صورتی که لبه‌ای تشخیص داده نشده باشد، رنگ سفید (مقدار یک) یا برعکس. مقدار حد آستانه یک مقدار تجربی است و می‌توانید به دلخواه مقدار معقولی برای آن انتخاب کنید.

نمونه‌ای از تشخیص لبه در تصویر در شکل زیر دیده می‌شود.



مقایسه پیکسل‌ها می‌بایست هم به صورت افقی و هم به صورت عمودی انجام شود. مقادیر پیکسل‌ها در بخشی از حافظه ذخیره می‌شوند و در صورتی که لبه‌ای تشخیص داده شود مقدار پیکسل متناظر با صفر و در صورتیکه لبه‌ای تشخیص داده نشود، مقدار آن با ۱ جایگزین می‌شود. (یا برعکس)

• در محیط Keil uVision میکروکنترلر مورد نظر را STM32F407VGT6 انتخاب کنید.

### بخش دوم

در این بخش، الگوریتم تشخیص لبه می‌بایست با یک شدن پین صفرم پورت GPIOA شروع به کار کند و در انتها پین اول پورت B را یک کند. (GPIO: General Purpose Input/Output)

- برای انجام بخش دوم، ابتدا باید کلاک باس‌های مورد استفاده را فعال کنید. برای کار با پورت‌ها، باید ابتدا کلاک پورت‌ها را از رجیسترهای RCC (Reset & Clock Controller) فعال کنید.
- سپس مود پین‌ها را تنظیم کنید.

- PA0 در مود خروجی، با سرعت 2MHz، به صورت Push Pull، بدون Pull up/Pull down

- PB1 در مود ورودی، با سرعت 2MHz، بدون Pull up/Pull down

برای مقدار دهی مناسب رجیسترها باید از دیتا شیت میکروکنترلر در ضمیمه استفاده کنید. (بخش RCC و GPIO)

## پروژه پایانی درس معماری کامپیوتر

- داده‌های تصویر در فایل ضمیمه txt قرار داده شده است. در این فایل داده‌ها به صورت ماتریسی ذخیره شده‌اند. این داده‌ها به صورت یک ماتریس ۱۵ در ۱۵ است که شامل بخشی از تصویر با ابعاد ۱۵ در ۱۵ پیکسل است. (دقت کنید که ترتیب قرار گیری پیکسل‌ها به صورت افقی و عمودی اهمیت دارد!)

- دانشجویانی که تمایل دارند پروژه را انجام دهند می‌بایست حتماً در کوییز مربوط به پروژه (مباحث کوییز: اسمبلی ARM) شرکت کنند. منظور شدن نمره پروژه منوط به شرکت در کوییز پروژه می‌باشد. موعد تحویل پروژه متعاقباً اعلام می‌شود. آن دسته از دانشجویانی که پروژه را در موعد مقرر تحویل ندهند یا به هر دلیلی در کوییز پروژه شرکت نکنند، نمره پروژه برای آن‌ها منظور نخواهد شد.

موفق باشید 😊