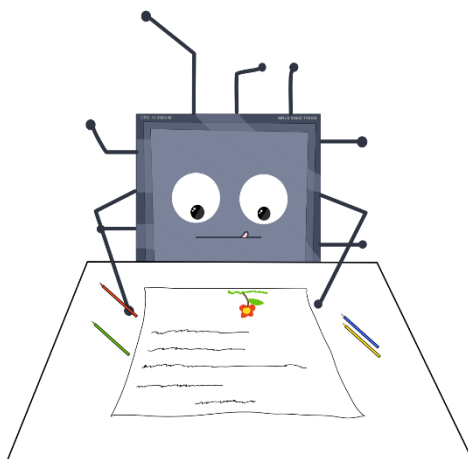




## Final Project

- ◀ مهلت ارسال پروژه ساعت ۲۳:۵۹ چهارشنبه ۲۶ دی می باشد.
  - ◀ پروژه به صورت شبیه سازی و با استفاده از نرم افزار Proteus است.
  - ◀ پروژه بایستی فردی انجام شود.
  - ◀ سوالات خود را می توانید از طریق تلگرام از آقایان **فربد بیجاری (@farbodbij)** و **رائین سلطانی (@raeinsoltani)** بپرسید.
  - ◀ فایل های پروژه شامل کد و فایل های پروتئوس را تنها با قالب **Project\_StudentNumber.zip** در کورسز بارگذاری کنید.
- نمونه: Project\_40031121.zip





## شرح کلی پروژه:

می‌خواهیم با استفاده از Arduino و ماتریس LED به عنوان صفحه نمایش، دستگاهی طراحی کنیم که با آن بازی [Space Invaders](#) را بتوان انجام داد.

## اهداف پروژه:

- کنترل ماتریس LED با Arduino
- برنامه نویسی بازی Space Invaders در Arduino و کنترل و نمایش آن در ماتریس LED در پروتئوس

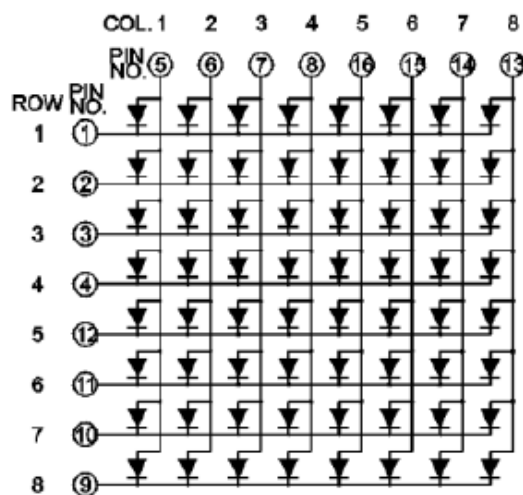
## ابزارهای مورد نیاز:

- Arduino Uno
- (MATRIX-8X8-BLUE) 8×8 LED Matrix
- 4 Buttons



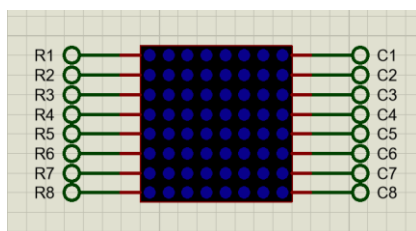
## آشنایی با ماتریس LED:

ماتریس LED شامل تعدادی LED است که به صورت سطری و ستونی به هم وصل شده‌اند.



Anodic Column/Cathodic Row

برای روشن کردن هر یک از LED ها باید پایانه‌ی مثبت آن را یک منطقی و پایانه‌ی منفی آن را صفر منطقی کنیم تا جریان از آن عبور پیدا کند. با کمی دقت در مدار می‌بینیم که نمی‌توان کل ماتریس را در یک لحظه کنترل کرد. برای مثال در شکل بالا فرض کنید دو LED در مختصات (۳,۱) و (۴,۳) را روشن می‌کنیم. می‌بینیم که همزمان با آن (۳,۳) و (۴,۱) نیز ناخواسته روشن می‌شوند. در این حالت برای رفع این مشکل، باید از تکنیک Multiplexing استفاده کنیم. در این تکنیک ما در هر لحظه فقط یکی از سطرها را کنترل می‌کنیم و بقیه خاموش هستند. به این صورت که سطر مورد نظر مقدار صفر منطقی می‌گیرد و ستون‌های متناظر با LED هایی از آن سطر که می‌خواهیم روشن کنیم یک می‌شوند. بعد از چند لحظه آن سطر را یک می‌کنیم و این کار را برای سطر بعدی تکرار می‌کنیم. در واقع در هر لحظه فقط یک سطر روشن است اما از آنجایی که این کار را با سرعت بسیار زیاد انجام می‌دهیم این خطای دید بوجود می‌آید که همه‌ی سطرها همزمان روشن هستند.

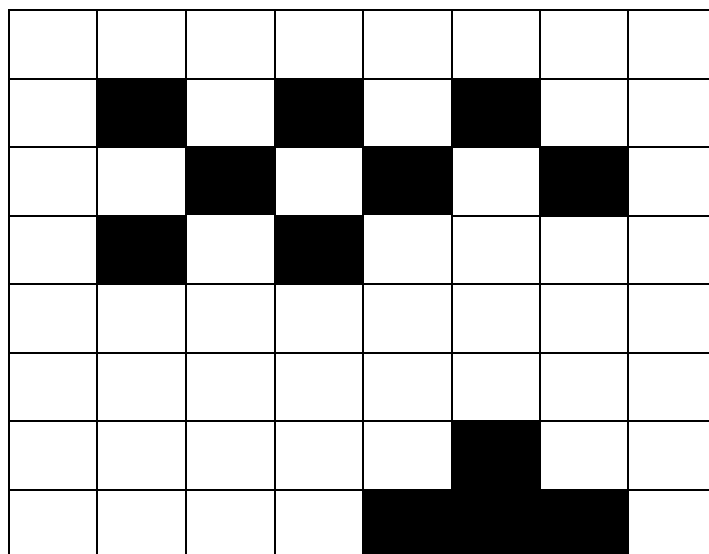


پین‌های ماتریس LED در پروتئوس به صورت زیر هستند.



## جزئیات پیاده‌سازی:

◀ مختارید به هر صورت که مایل بودید موجودات فضایی بازی Space Invaders را پیاده‌سازی کنید و آن را روی ماتریس LED نمایش دهید. برای ساده ترین حالت قابل قبول نیازمند ۸ مستطیل ۱×۱ که به شکل تصادفی در نیمه بالا صفحه ایجاد شدند هستید.



- ◀ کنترل سفینه باید با استفاده از ۴ دکمه (چهار جهت اصلی) صورت گیرد.
- ◀ سفینه با استفاده از کلید دلخواه پنجمی، پرتابه ای را رو به بالا شلیک می‌کند و در صورت برخورد خانه مورد اصابت از صفحه بازی حذف می‌شود.
- ◀ در صورت از بین رفتن تمامی خانه ها صفحه بازی ریست می‌شود.
- ◀ نیازی به نگه داشتن و نمایش امتیاز نیست.



◀ در طراحی معماری نرم افزار نیز مختارید اما برای سادگی کار، پیشنهاد ما این است:

- با اینکه به دلیل استفاده از Multiplexing در تمام لحظات نیاز است صفحه نمایش آپدیت بشود اما وضعیت موجودات فضایی، سفینه و پرتابه (اعم از مکان، سرعت حرکت و ...) نیازی به آپدیت شدن مداوم ندارد و می توانید با استفاده از یک وقفه تایمر این کار را (برای مثال) هر ۰.۵ ثانیه یک بار انجام دهید. این فاصله ۰.۵ ثانیه ای را می توانید خودتان به گونه ای تنظیم کنید که بازی روان تر اجرا شود و آن را tick می نامیم.
- همان طور که میدانید آردوینو بر پایه زبان برنامه نویسی C++ است و امکان استفاده از مفاهیم شی گرای در آن مهیاست. دستورات این بخش شباهت زیادی به زبان برنامه نویسی Java دارد. برای مطالعه بیشتر میتوانید به [این لینک](#) مراجعه کنید.
- مستقیماً کار نمایش روی ماتریس LED را انجام ندهید. یک آرایه boolean با اندازه ۸×۸ داشته باشید که مقدار ۱ به معنای روشن بودن LED در آن مختصات و ۰ به معنای خاموش بودن آن است. در تابعی در کلاس پدر که وظیفه نمایش دادن مستطیل را دارد صرفاً این آرایه را تغییر دهید و کاری به ماتریس LED نداشته باشید. قسمت مجزایی از کد که قرار است ماتریس LED را کنترل کند طبق این آرایه عمل میکند و این آرایه را روی ماتریس نمایش میدهد.

◀ مجازید از کتابخانه های مختلف استفاده کنید، اما نباید از توابع آماده برای حرکت دادن موجودات فضایی، سفینه و پرتابه استفاده کنید. استفاده شما از کتابخانه ها باید در حد نشان دادن پیکسل ها روی صفحه نمایش باشد.

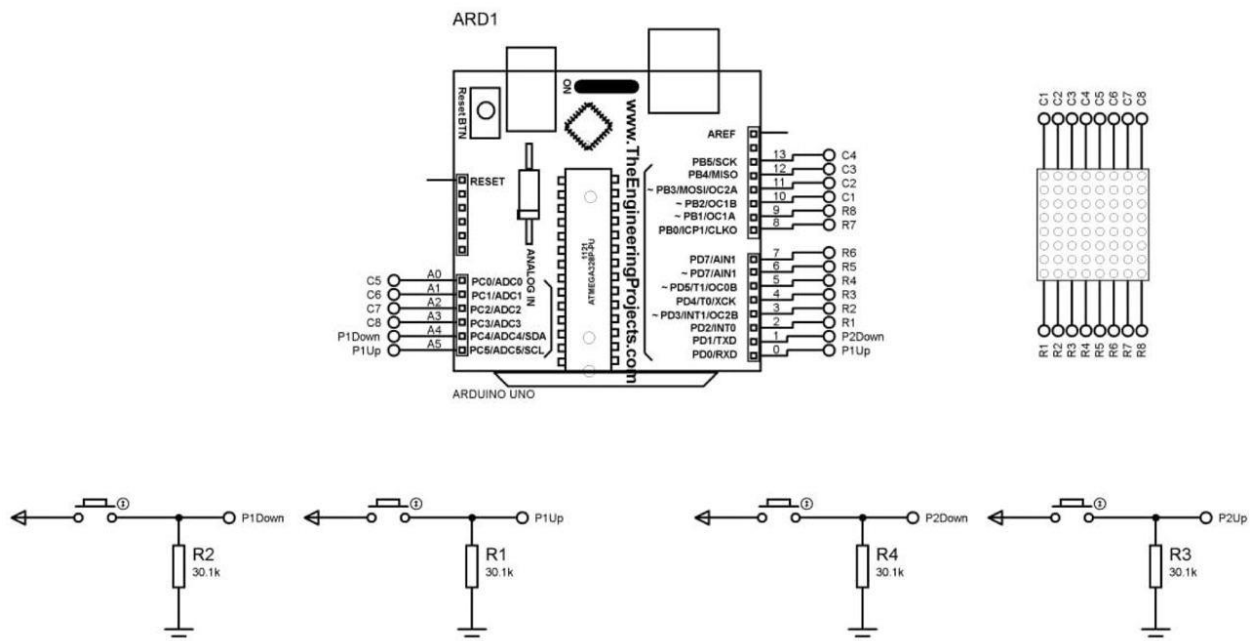


◀ مواردی که شامل **نمره امتیازی** خواهند بود:

- سیم کشی منظم (برای اینکار میتوانید از یکی از دو روشی که در [این لینک](#) توضیح داده شده است، استفاده کنید)
- استفاده از ماتریس RGB به جای ماتریس LED تک رنگ و کنار هم قرار دادن چند ماتریس RGB برای بزرگ کردن زمین بازی (مثلا ۱۶X۱۶)
- برای وصل کردن صفحه‌ها به هم نیاز دارید از چیپ MAX۷۲۱۹ استفاده کنید:
  - چیپ MAX۷۲۱۹ یک آی‌سی کنترلر است که وظیفه کنترل و مالتی‌پلکسینگ صفحه نمایش را برای ما انجام می‌دهد. علاوه بر این می‌توانیم با استفاده از این چیپ چند صفحه نمایش را به همدیگر متصل کنیم و همه را با استفاده از ۳ پین SPI کنترل کنیم. در این پروژه می‌خواهیم با استفاده از این چیپ یک تابلوی روان بسازیم. برای اطلاعات بیشتر در مورد این چیپ می‌توانید از [این لینک](#) استفاده کنید.
- پیاده سازی پرتاب تیر از موجودات به فضایی به سمت پایین. در صورت برخورد موفقیت به سفینه صفحه بازی ریست می‌شود. (برای اطلاعات بیشتر می‌توانید از [این لینک](#) استفاده کنید.)

## نرم افزار شبیه سازی Proteus:

این نرم افزار مانند نرم افزار ORCAD که در آزمایشگاه مدارهای الکتریکی با آن آشنا شدید، امکان شبیه سازی مدارهای الکتریکی را فراهم می کند. افزون بر آن، شبیه سازی کارکرد خانواده‌هایی از میکروکنترلرها در مدار را فراهم می‌کند. از این رو، میتوان از این نرم افزار برای شبیه‌سازی آزمایش‌های درس ریزپردازنده استفاده کرد.



موفق باشید