* مفهوم PWM و استفاده‌های آن:
  + Pulse Width Modulation: سروو موتور دارای تنها یک سیم کنترلی می‌باشد که باید از طریق آن زاویه و درواقع مقدار چرخش را مشخص نمود. از طرفی از طریق یک سیم تنها می‌توان دو مقدار ۰ و ۱ را انتقال داد، به همین دلیل با کنترل کردن Duty Cycle و یا نسبت زمان ۱ به ۰ بودن سیگنال در یک پریود مقادیر مختلفی را به کنترلر منتقل می‌کند.(تنها استفاده آن در سروو موتور نمی‌باشد و در موارد مختلف مانند تنظیم مقدار نور، تنظیم رنگ led، سرعت موتور و ... برای ایجاد مقادیر مختلف استفاده می‌شود)
* کاربردهای Servo Motor:
  + سروو موتور سرعت و چرخش زیادی ندارد ولی دارای قدرت زیادی می‌باشد، از این رو از آن برای کارهای نیازمند به قدرت بالا مانند آسانسور، سکان، ربات و ... استفاده میگردد.
* توضیح در مورد ورودی آنالوگ و تحلیل آن در آردواینو و تابع مورد استفاده این آزمایش:
  + از این ورودی جهت پیاده سازی و استفاده از PWM بهره برده می‌شود و در صورتی که پین مورد نظر قابلیت PWM را داشته باشد، می‌توان با استفاده از analogWrite دیوتی سایکل و درواقع مقادیر مختلف بین ۰ تا ۲۵۵ را به کنترلر منتقل کرد
* analogRead():
  + برعکس analogWrite عمل میکند و به صورت پیش فرض برای ۱۰ بیت تعریف شده و وضعیت پین را در قالب عددی بین ۰ و ۱۰۲۳ به ما برمیگرداند.
* توابع مورد نیاز از کتابخانه Servo.h:
  + **attach():** شماره پین کنترلی که وظیفه پیاده سازی PWM را بر عهده دارد به عنوان ورودی میگیرد و کارهای مربوطه به آن را هندل می‌کند.
  + **write():** زاویه مورد نظر را به عنوان ورودی گرفته و سروو موتور را بر روی آن زاویه تنظیم میکند.
  + **read():** زاویه فعلی سروو موتور را خروجی می‌دهد.
  + **writeMicroseconds():** همانند دستور write بوده با این تفاوت که ورودی آن بین ۱۰۰۰ تا ۲۰۰۰ می‌باشد و در نتیجه دقت بالاتری به ما می‌دهد(اگر چه ممکن است بخاطر ضعف موتور خطای بیشتری نیز داشته باشد).
  + **readMicroseconds():** زاویه فعلی موتور را برحسب میکروثانیه به ما برمی‌گرداند.