

انواع موتور

موتور DC

○ با جاروبک

یک سیم مثبت و یک سیم منفی دارند. با مقدار ولتاژ ورودی میتوان سرعت چرخش را کنترل کرد. این موتور دو آهن ربا دارد. آهن ربا بیرونی دائم و آهن ربا داخلی موقت است به طوری که با جریان برق خاصیت آهنربایی پیدا میکند. بدنه اصلی موتور به 3 آهن ربا موقت متصل است که به طور متوالی قطب هایش عوض میشود. (جهت)

عیب این موتور وجود Brush هایی در آن است که به مرور زمان توسط استحاکاک از بین میرود.

○ بی جاروبک

سه سیم یا بیشتر دارند و برای کنترل سرعت نیاز به Speed Controller دارند. این موتور ها برعکس موتور قبلی آهن ربا دائی در وسط و آهن ربا های موقت در اطراف هستند. این باعث میشود دیگر نیازی به Brush ندارد (چون سیمی به آن وصل نیست).

در این موتور باید به ترتیب خاصی یکی از آهن ربا های موقت را خاموش کرده و بقیه را مخالف هم قرار دهیم تا باعث چرخش موتور شود. (پس جریان باید موج مربعی باشد که در 3 سیکل تکرار میشود)

استپر موتور (باز هم DC)

پنج یا شش سیم دارند که یکی همیشه برای زمین است و برای کنترل سرعت نیاز به میکرو کنترلر هستند. با استفاده از Chips میتوان 360 درجه را به استپ های کوچک تر تقسیم کرد.

از قابلیت های این موتور آن است که میتوان آن را در درجه ای خاص متوقف کرد به طوری که ثابت بایستد. از معایب آن، در صورت ثابت ایستادن در حال استفاده جریان است (یعنی با جریان موتور ثابت ایستاده) پس پرمصرف تر و گران تر است.

سرو موتور (باز هم DC)

سه سیم دارند (سر مثبت و زمین و سیگنال) و برای کنترل سرعت نیاز به میکرو کنترلر هستند. به طوری که منبع DC بسیار سریع اما ضعیف است و چرخ دنده ها باعث قوی تر در ازای کاهش سرعت است. این سیستم با feedback همراه است یعنی موقعیت موتور را فهمیده و به چه سمتی باید حرکت کند، پس چند میلی ثانیه و در چه جهتی بهش ولتاژ دهد. (با استفاده از پتانسیومتر)

از معایب این موتور آن است که نمیتواند 360 درجه بچرخد چون پتانسیو متر نمیتواند 360 درجه بچرخد.

کاربرد؟ اینکه اکثر در زاویه 0 تا 180 درجه بچرخد، مثلاً بستن پیچ اما موتور هایی هستند که 360 درجه هم بچرخند. در این موتور ها بجای دانستن موقعیت، سرعت را اندازه میگیرند.

موتور AC

اگر سه فاز باشد به سه سیم فاز نیاز دارد و یک سیم نول. اگر تک فاز باشد فقط یک فاز و یک نول. طراحی این موتور ها DC بی جاروبک مشابه است با این تفاوت که جای آهنربای دائم وسط، یک شبکه فلزی دارند که جریان آهنربا های بیرونی یک میدان الکتریکی ایجاد بشه که در وسط شبکه یک میدان مغناطیسی القا کند و باعث چرخش شود.

PWM چیست (Pulse Width Modulation)

ما در موتور سروو یک سیم سیگنال برای انتقال اطلاعات داریم. حال اگر بخواهیم یک عدد از برد به موتور بدهیم، نمیتوان با استفاده از محور عمودی اینکار را انجام داد زیرا فقط دو مقدار 0 یا 1 (0 یا 5 ولت) داریم. اما میتوان بر روی محور افقی (زمان) بازه های مشخصی را تعیین کرده و برای مثال گفت 70% آن 1 بوده و 30% باقی 0 باشد. با اینکه چقدر از آن زمان یک است میتوان عدد را تشخیص داد. (سیکل کاری – چقدر از زمان یک است)

کاربرد های آن؟ پاس دادن عدد به Device

- بازر های منفعل (صدا ها در فرکانس های مختلف – با استفاده از تغییرات فرکانس صدا تولید میکنند)
- سرعت موتور (چقدر سریع؟)
- نور چراغ (چقدر روشنایی؟ یا چه طیف رنگ هایی؟)

کتابخانه Servo

Attach() برای اختصاص دادن یک پین به موتور سروو است. (پهنای پالس به میکرو ثانیه هم جز آرگومان های انتخابی است)

Detach() در عکس Attach() است و پین را آزاد میکند.

Write() یک عدد بین 360 درجه میگیرد و بسته به نوع موتور سرعت یا زاویه را ست میکند.

Read() برعکس Write() زاویه فعلی را خروجی میدهد.

WriteMicroSecond() یک عدد در واحد میکرو ثانیه گرفته و 1500 باشد یعنی وسط بایستد، 2000 و 1000 به طرفین میچسبد.

ReadMicroSecond() مانند تابع بالایی صرفاً دقت زیاد میشود موقع خواندن.

Analog to Digital Convertor (ADC)

این قابلیت در برخی پین های میکرو وجود دارد. ولتاژ کوچک تر از 2 باشد 0 میشود (بین 0 تا 3 ولت 0 محسوب میشود) و این دستگاه موج را پلکانی میکند تا دقت افزایش پیدا کند به طوری که ارتفاع سیگنال که 2- تا 2 ولت است به 1024 عدد مپ میشود. با یک تناسب میتوان فهمید دقیقاً چند ولتاژ داریم.

در سورت استفاده از پتانسومتر که یک مقاومت متغیر است، ولتاژ تولیدی میتواند هر عددی بین 0 تا 5 باشد که اگر به این بین بدهیم ولتاژ مپ میشود به 1024 عدد (Scale میشود) که میتوان از آن به عنوان ورودی استفاده کرد. پس دستور map صرفاً scale عدد را عوض میکنیم. (این دقت ایجاد شده Resolution نام دارد)

AnalogRead() مقدار خوانده شده توسط ADC به کمک این دستور انجام میشود. خروجی این دستور عددی بین 0 تا 1023 است.