

بخش الکترونیک یادآوری (مفاهیم اولیه، عناصر الکتریکی و ابزار کار)

سرفصل عناوین مورد بحث
الف) مفاهیم پتانسیل الکتریکی جریان سیگنال متناوب و مستقیم اتصال سری و موازی پالس قطار پالس
ب) عناصر الکتریکی و الکترونیکی باطری مقاومت خازن
ج) ابزار ساخت مدار

پتانسیل الکتریکی (V)

انرژی بارهای الکتریکی موجود در یک جسم، پتانسیل الکتریکی نام دارد که می‌تواند بارها را بصورت جریان از عناصر رسانا عبور دهد.

زمین

زمین واقعی سطح پتانسیل صفر می‌باشد. در مدارات الکتریکی جهت مقایسه سطح ولتاژ نقاط مختلف، یک نقطه را بعنوان مرجع پتانسیل صفر فرض می‌کنند و ولتاژ بقیه نقاط را نسبت به آن محاسبه می‌نمایند.



پیش فرض
اگر سطح پتانسیل یک نقطه بیشتر از سطح پتانسیل مرجع باشد علامت آن مثبت و اگر کمتر باشد علامت آن منفی می‌شود.

جریان الکتریکی (I):

به حرکت بارهای الکتریکی در یک رسانا جریان الکتریکی گفته می‌شود.
در صورتیکه دو سر یک عنصر، اختلاف پتانسیل (اختلاف ولتاژ) وجود داشته باشد، بارهای الکتریکی جهت ایجاد تعادل از سمت پتانسیل بیشتر به پتانسیل کمتر حرکت می‌کنند که همان جریان الکتریکی است.
در عمل جهت حرکت الکترون‌ها از قطب منفی به مثبت است اما بصورت قراردادی جهت جریان را از قطب مثبت به منفی در نظر می‌گیریم.

سیگنال

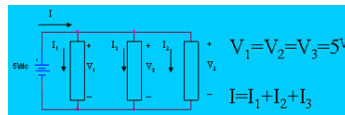
سیگنال‌های الکتریکی ولتاژ یا جریانی هستند که در مدار ایجاد می‌شوند. سیگنال‌ها (معمولاً ولتاژ) انتقال دهنده اطلاعات هستند.

اتصال عناصر

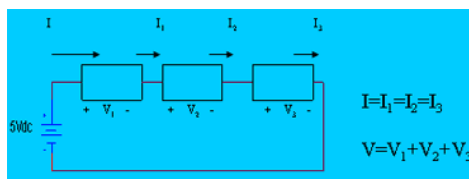
اجزا یک مدار بسته به هدف طراحی بصورت سری، موازی یا ترکیبی بسته می‌شوند.

اتصال موازی:

در اتصال موازی ولتاژ دو سر اجزا با یکدیگر برابر است.



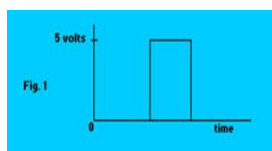
اتصال سری



در اتصال سری جریان عبوری از اجزا با هم برابر است.

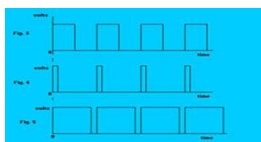
پالس و قطار پالس

پالس به شکل موجي گفته مي شود كه در يك زمان بسيار کوتاه از سطح صفر به يك سطح ولتاژ مشخص مي رسد و براي مدتي در اين سطح ولتاژي باقي مي ماند و سپس به سطح صفر برمي گردد.



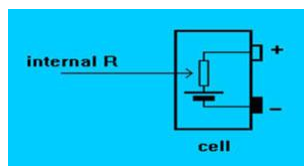
قطار پالس

از تکرار پالسهاي مشابه يك قطار پالس ساخته مي شود.



باطري:

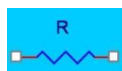
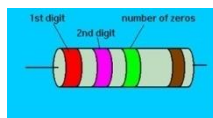
وسيله ايست كه انرژي شيميايي را به انرژي الكتريكي تبديل مي كند و در مدارهاي كم مصرف بعنوان منبع انرژي مورد استفاده قرار مي گيرد.



باطري ها به دو دسته قابل شارژ و يكبار مصرف تقسيم مي شوند.

مقاومت:

جسم هادي دو سري است كه در برابر عبور جريان مقاومت مي كند و ولتاژ دو سر آن متناسب با جريان عبوري از آن مي باشد.

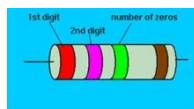


هرچه مقدار مقاومت بيشتر باشد جريان كم تري از آن عبور مي كند.

در الكترونيك مقاومت را با حرف R (ابتدائي كلمه Resistor) نمايش مي دهند.
مقدار ولتاژ دو سر يك مقاومت تنها به مقدار جريان عبوري بستگي دارد. قانون اهم بيانگر اين وابستگي مي باشد:

$$V/R=I$$

واحد اندازه گيري مقاومت اهم Ω مي باشد.



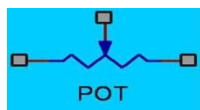
شماره	رنگ
0	سیاه
1	قهوه ای
2	قرمز
3	نارنجی
4	زرد
5	سبز
6	آبی
7	بنفش
8	خاکستری
9	سفید

خطای مقدار مقاومت معمولاً مقدار نامی مقاومتها با مقدار واقعی آن کمی اختلاف دارد. مقدار این خطا را با یک نوار رنگی که از نوارهای مربوط به قرائت مقدار مقاومت فاصله دارد، نشان می‌دهند.

نقره ای $\pm 10\%$
 طلایی $\pm 5\%$
 قرمز $\pm 2\%$
 قهوه ای $\pm 1\%$

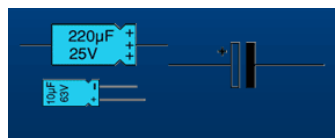
مقاومت متغیر (پتانسیومتر)

نوعی مقاومت الکتریکی است که مقدار آن قابل تنظیم است. پرکاربردترین نوع آن در الکترونیک به Multi turn معروف است.



خازن:

از دو صفحه فلزی که با یک عایق که دی‌الکتریک نامیده می‌شود از یکدیگر جدا شده‌اند تشکیل شده است. در الکترونیک خازن را با حرف C (ابتدای کلمه Capacitor) نشان می‌دهند.



ظرفیت خازن معیاری برای اندازه‌گیری توانایی نگهداری انرژی الکتریکی است. ظرفیت زیاد بدین معنی است که خازن قادر به نگهداری انرژی الکتریکی بیشتری است. واحد اندازه‌گیری ظرفیت فاراد است. یک فاراد واحد بزرگی است و مشخص‌کننده ظرفیت بالایی می‌باشد. بنابراین استفاده

از واحدهای کوچکتر نیز در خازنها مرسوم است .
میکروفاراد μF ، نانوفاراد nF و پیکوفاراد pF واحدهای کوچکتر فاراد هستند .

قرائنت مقدار ظرفیت خازن

در بسیاری از خازن ها با ظرفیت کم ، ظرفیت بر روی خازن نوشته شده ولی هیچ واحد یا مضربی برای آن چاپ نشده و برای دانستن واحد باید به دانش خودتان رجوع کنید . خازن ذخیره کننده انرژی الکتریکی است. بعبارت دیگر بوسیله بار الکتریکی شارژ می شود.
در الکترونیک از خازن برای اتصال بخشهای مختلف یک مدار، تغییر شکل موج ولتاژ و... استفاده می شود.
کاربرد دیگر خازن ها صاف کردن سطح تغییرات ولتاژ مستقیم می باشد . از خازن ها در مدارات بعنوان فیلتر هم استفاده می شود .
زیرا خازن ها به راحتی سیگنالهای غیر مستقیم AC را عبور می دهند ولی مانع عبور سیگنالهای مستقیم DC می شوند .

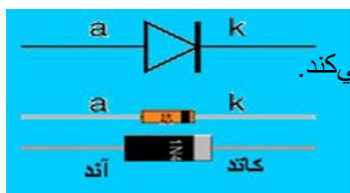
نکاتی در مورد خازنها

در صورت شارژ خازن با ولتاژ بالا حتی در صورت قطع بودن مدار خطر برق گرفتگی در صورت تخلیه بار الکتریکی ذخیره شده وجود دارد. جهت کاهش این خطر معمولاً خازنهایی را که با ولتاژ بالا شارژ کرده اند با یک مقاومت سری می کنند تا جریان تخلیه محدود شود.
جریان مستقیم نمی تواند از خازن عبور نماید.
بعضی از خازنها اصطلاحاً پلاریزه هستند یعنی قطب مثبت و منفی دارند. لذا باید توجه کرد که هنگام استفاده از این خازنها بطور صحیح در مدار قرار داده شوند.

ابزار ساخت و اندازه گیری و قطعات الکترونیک

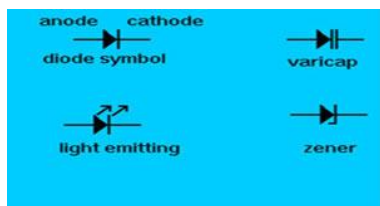
منبع تغذیه ، سیگنال ژنراتور، برد مورد ، مولتی متر ، اسیلوسکوپ، قطعات الکترونیک ، مقاومت ، مقاومت متغیر، خازن

دیود – ترانزیستور – تقویت کننده عملیاتی



دیود
یک عنصر نیمه هادی دو سر است که جریان الکتریکی را یکسو می کند.

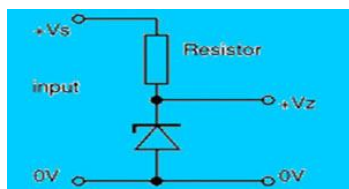
دیودها دارای قطب مثبت (آنود) و منفی (کاتود) هستند و جریان را فقط هنگامیکه از سمت مثبت وارد می شود (بایاس مستقیم) از خود عبور می دهند.
قطب کاتود معمولاً روی بدنه دیود نشانه گذاری شده است.
دیود نوری (LED) و دیود زینر نمونه های دیگری از انواع دیودها هستند.



انواع دیود

دیود زینر

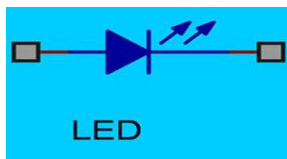
نوعی دیود است که بطور معکوس بایاس می شود و از آن برای تهیه یک ولتاژ ثابت (که معمولاً مقدار آن روی دیود نوشته می شود) استفاده می شود.



دیود نورانی (LED)

به دیودی گفته می شود که هنگام روشن بودن نور متصاعد می کند.

دیودهای نورانی در رنگهای مختلف ساخته شده است.



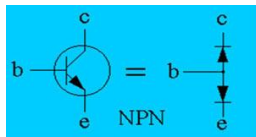
دیود نوری

نوع خاصی از دیود است که هنگامیکه در معرض نور قرار می‌گیرد روشن می‌شود. از این دیود بعنوان حسگر برای تشخیص نور استفاده می‌شود

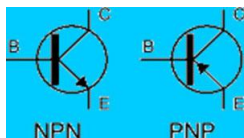


ترانزیستور:

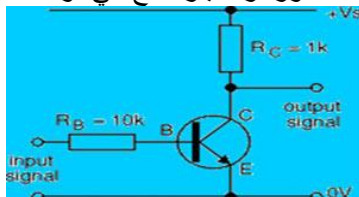
یک عنصر نیمه‌هادی سه پایه است که در مدارها بعنوان تقویت کننده جریان و کلید قطع و وصل مورد استفاده قرار می‌گیرد.



دو نوع اصلی ترانزیستورهای پیوندی NPN و PNP نام دارند.

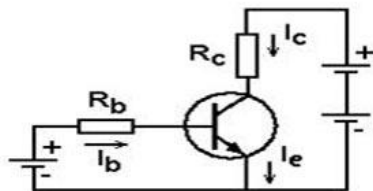


در حالتی که ترانزیستور مثل یک کلید عمل می‌کند، اگر به پایه بیس جریانی وارد شود (ولتاژ بیس مثبت باشد) پایه کلکتور به امیتر اتصال کوتاه می‌شود و اگر ولتاژ بیس صفر یا منفی باشد کلکتور از امیتر قطع می‌شود.



در حالت تقویت کنندگی β برابر جریان عبوری از بیس، از کلکتور و امیتر عبور می‌کند. به β ضریب تقویت جریان ترانزیستور گفته می‌شود که برای هر ترانزیستور عدد مشخص و ثابتی است. β معمولاً بین 50 تا 200 می‌باشد.

روابط زیر در مورد جریان پایه‌های یک ترانزیستور برقرار است:

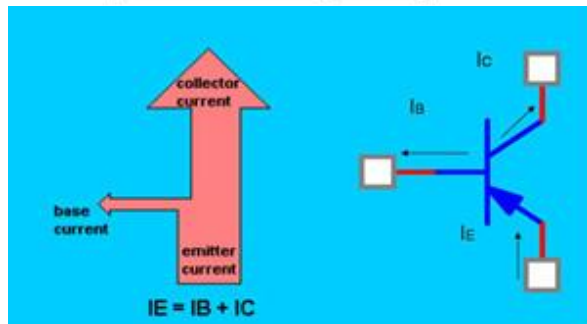


$$I_c = \beta \cdot I_b$$

$$I_e = I_b + I_c$$

$$I_e = (1 + \beta) \cdot I_b$$

رابطه جریانی ترانزیستور

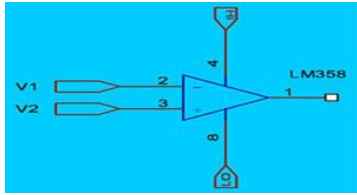
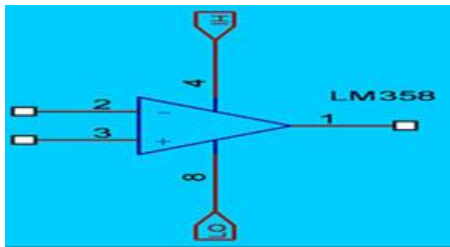


تقویت کننده عملیاتی:

یکی از پرکاربردترین مدارهای مجتمع (IC) است که در مدارها بعنوان تقویت کننده، مقایسه گر و... مورد استفاده قرار می‌گیرد.

در حالت عادی یک Op-Amp پنج پایه دارد:

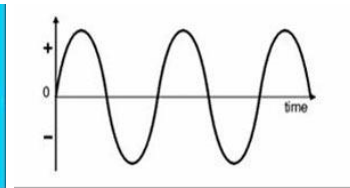
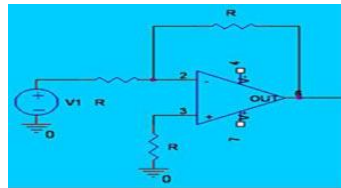
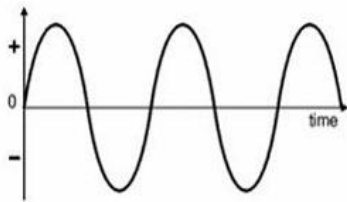
دو پایه ورودی مثبت و منفی - دو پایه تغذیه مثبت و منفی - یک پایه خروجی



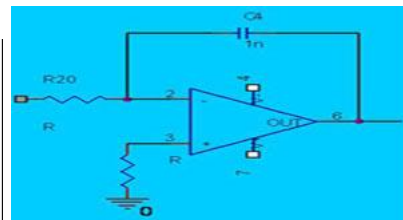
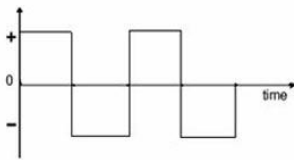
کاربردهای آپ امپ

حالت مقایسه کننده

حالت تقویت کننده (فیدبک منفی)



حالت عملیاتی

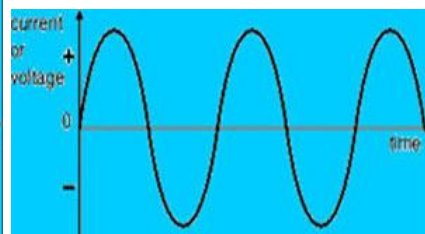


سیگنال متناوب و مستقیم، رگولاتور، رله معمولی و دو طرفه و ...)

سیگنال متناوب و مستقیم - رگولاتور - رله و انواع آن

سیگنال متناوب (AC)

سیگنالهای متناوب در یک مسیر منتشر میشوند و سپس تغییر مسیر می دهند و این عمل دائماً تکرار می شود .



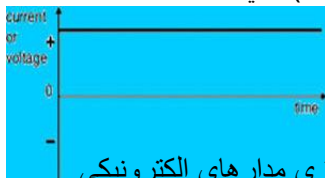
به هر تغییر بین مثبت و منفی ، یک سیکل گفته می شود و واحد آن هرتز می باشد .

در ایران وسایل الکتریکی با فرکانس 50 هرتز کار می کنند .
 سیگنالهای متناوب برای راه اندازی وسایلی از قبیل لامپ ها و گرم کننده ها بکار می روند ولی اکثر مدارهای الکتریکی برای کار نیاز به یک ولتاژ مستقیم دارند

سیگنال مستقیم (DC)

جریان مستقیم همیشه در یک مسیر جاری می شود (همیشه مثبت و یا همیشه منفی است) ولی ممکن است میزان آن کاهش یا افزایش پیدا کند .

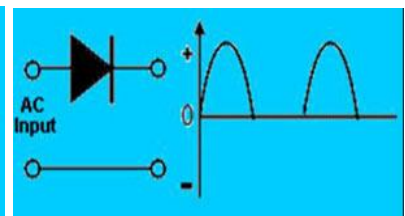
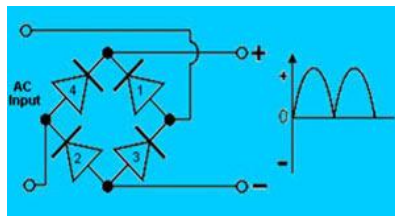
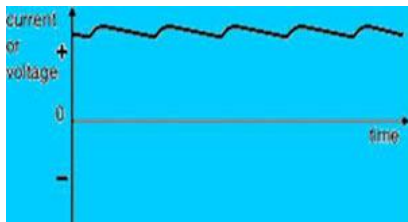
جریان مستقیم همیشه در یک مسیر جاری می شود (همیشه مثبت و یا همیشه منفی است) ولی ممکن است میزان آن کاهش یا افزایش پیدا کند .



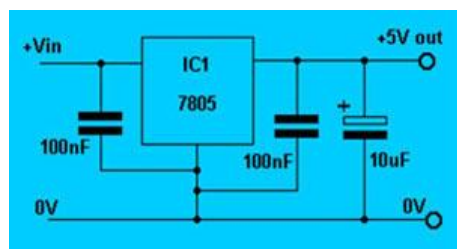
باتری ها و رگولاتورها ولتاژ مستقیم می دهند و این ولتاژ برای تغذیه و راه اندازی مدارهای الکترونیکی مناسب است .

یکسوسازی

برای تهیه سیگنالهای مستقیم از یکسوسازها استفاده می کنند . - یکسوسازی می تواند نیم موج یا تمام موج باشد.

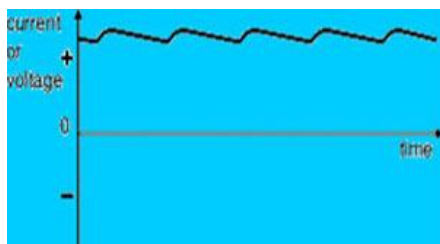


برای تهیه سیگنالهای مستقیم از یکسوسازها استفاده می کنند . یکسوسازی می تواند نیم موج یا تمام موج باشد.



رگولاتور:

یک عنصر سه پایه تثبیت کننده ولتاژ است.
 رگولاتورها در ولتاژهای مختلف قابل تهیه هستند.



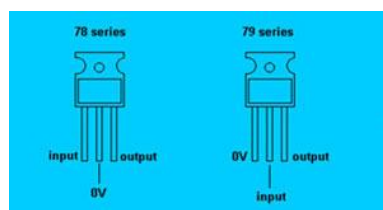
انواع رگولاتورها

رگولاتورهای $\pm 5V$:

7805 و 7905

رگولاتورهای $\pm 12V$:

7812 و 7912

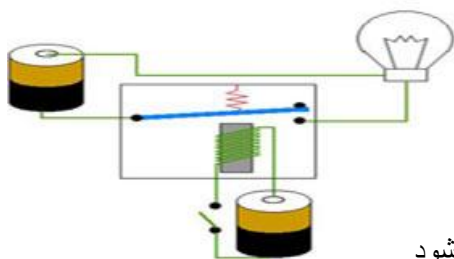


گرماگیر

جهت کاهش دمای رگولاتور معمولاً از گرماگیر (Heat Sink) استفاده میشود.

رله و انواع آن

یک آهنربای الکتریکی است که بعنوان کلید کنترل شونده با جریان عمل میکند.

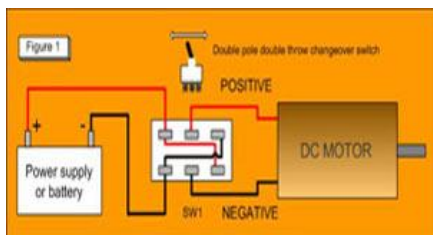


رله دوطرفه

از رله دو طرفه برای کنترل همزمان چرخش به چپ و راست موتور استفاده میشود.

کاربرد رله

کنترل مدارهای ولتاژ بالا توسط سیگنالهای کم ولتاژ
کنترل مدارهای جریان بالا توسط سیگنالهای کم جریان
شناسایی و جداسازی خطا و ایراد در خطوط انتقال و توزیع با باز و بسته کردن کلیدها (رله حفاظتی)
جداسازی مدار کنترل از مدار اصلی
انجام عملیات منطقی (...AND, OR, XOR, NOR, NAND)
ایجاد عملیات تأخیر زمانی



انواع موتور و درایورهای آنها

موتور AC - موتور DC - موتور پله‌ای (Stepper motor)

موتور :

از موتور برای تبدیل انرژی الکتریکی به انرژی مکانیکی استفاده میشود.
انتخاب نوع موتور مورد استفاده به نوع کاربرد و ویژگیهای هر موتور بستگی دارد.

موتور AC :

معمولاً در مدارهایی با مصرف انرژی زیاد و دستگاههای الکتریکی خانگی مورد استفاده قرار می‌گیرد.
این موتورها با جریان متناوب برق کار می‌کنند لذا به آنها موتور AC گفته میشود.
یخچال، جاروبرقی و آبمیوه‌گیری موتور AC دارند.
برای کنترل میزان چرخش موتور از وسیله‌ای بنام شفت انکودر استفاده میشود.

موتور DC:

توان مکانیکی آنها عموماً کمتر از موتورهای AC است. - ساختار ساده‌ای دارند. - بسیاری از اسباب بازیهای برقی با موتور DC کار می‌کنند. - بسیاری از اسباب بازیهای برقی با موتور DC کار می‌کنند - اغلب برای استفاده از موتور DC به مدار راه‌انداز نیاز داریم. - آرمیچر بارزترین نوع موتور DC است. - ایراد موتور DC عدم امکان کنترل دقیق سرعت و چرخش موتور است. - قیمت پایین، تنوع قدرت و سرعت، از جمله مزایای استفاده از موتورهای DC میباشد.

موتور پله‌ای (Stepper motor)

کاربرد اصلی این موتورها در کنترل موقعیت است.

این موتورها ساختار کنترلی ساده‌ای دارند. لذا در ساخت ربات کاربرد زیادی دارند. مطابق با تعداد پالسهای که به یکی از پایه‌های راه‌انداز موتور ارسال میشود موتور به چپ یا راست می‌چرخد.
توان خروجی این موتورها کمتر از دو نوع قبلی است. استفاده از موتور پله‌ای مشکلاتی از جمله وزن زیاد، قیمت بالا و قدرت بسیار کم را بدنبال دارد.
موتور پله‌ای نسبت به دو موتور قبلی دارای حرکت دقیق و حساب شده تری هستند. این موتورها به صورت درجه ای دوران می‌کنند و با درجه های مختلف در بازار موجود هستند.

واژه پله به معنی چرخش به اندازه درجه تعریف شده موتور اطلاق می شود.

$$۲۰۰ * ۱/۸ = ۲۵$$

$$۱۵ * ۲۴ = ۳۶۰$$



چرخش موتور پله ای

موور پله کامل و نیم پله

در حالت عادی میزان چرخش موتور به تعداد پالسهای اعمالی و گام موتور بستگی دارد. هر پالس يك پله موتور را می چرخاند. با تحريك دو فاز مجاور در موتور می توان موتور را به اندازه نیم پله حرکت داد. به این ترتیب تعداد پلهای موتور دو برابر می شود.

راه اندازی موتور پلهای

موتور دیسک سخت يك نمونه موتور پلهای است. تراشه L297 يك راه انداز مناسب برای موتور پلهای است.

مقایسه موتور DC و موتور پلهای

موتور پله ای	موتور DC
استفاده در توانهای پایین	استفاده در توانهای بالا
راه اندازی پیچیده	راه اندازی ساده
بدون نیاز به سامانه کنترلی	کنترل با فید باك
تعمیر و نگهداری ساده و کم	تعمیر و نگهداری مداوم

معرفی حسگرها و انواع آنها

انواع حسگرها - حسگر نوری

حسگر

حسگر يك وسیله الكتريكي است كه تغییرات فیزیکی یا شیمیایی را اندازه گیری می کند و آنرا به سیگنال الكتريكي تبدیل می نماید. - حسگرها معمولاً در طبقه اول مدارات كنترلي قرار می گیرند.

انواع حسگرها

فاصله - رنگ - نور - صدا - حرکت و لرزش - دما - دود

مزایای سیگنالهای الكتريكي

سهولت تقویت كمیتهای الكتريكي - پردازش راحت تر و ارزان تر - انتقال آسان - قابلیت ذخیره و نمایش - دقت بالا سرعت بالا

حسگرهای مورد استفاده در رباتيك

حسگرهای تماسی (Contact) - حسگرهای هم جوارى (Proximity) - حسگرهای دوربرد (Far away)

حسگرهای تماسی

آشکارسازی تماس دو جسم
اندازه گیری نیروها و گشتاورهایی که حین حرکت ربات بین اجزای مختلف آن ایجاد می شود

حسگرهاي همجوارى

آشكارسازى اشيا نژديك به روبات - القاىي - اثر هال - خازنى - اولتراسونيك - نوري

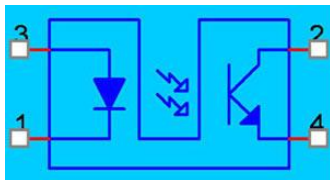
حسگرهاي دوربرد

فاصله سنج (ليزر و اولتراسونيك) - بينايى (دوربين CCD)

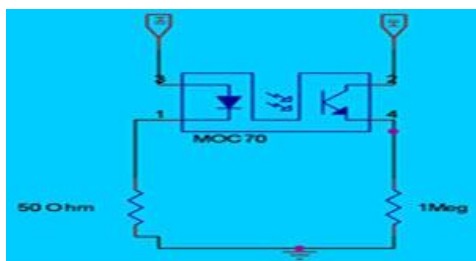
حسگر نوري (گيرنده-فرستنده)



از يك ديود نورانى (فرستنده) و يك ترانزيستور نوري (گيرنده) تشكيل شده است

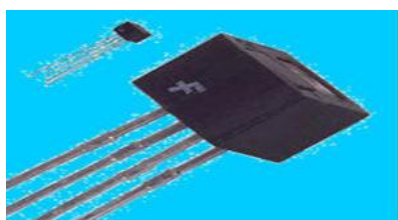


مدار راه انداز حسگر نوري



خروجى حسگر نوري

سطح سفيد = صفر ولت
سطح سياه = پنج ولت

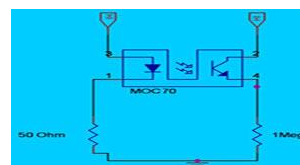
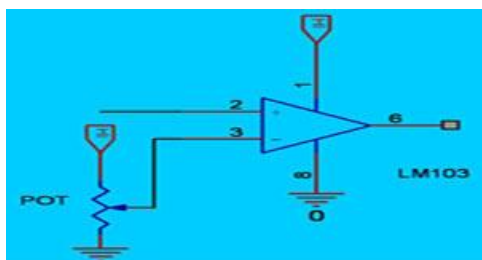


معرفى مدار كنترل ربات مسيرياب

اجزاء مختلف الكترونيك و كنترل ربات - حسگر - مقايسه گر - ميكروكنترلر - رله - موتور

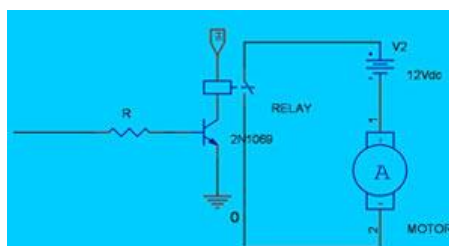
بلوك دياگرام الكترونيك ربات مسيرياب

موتور	<----	رله	<----	كنترلر	<----	مقايسه گر	<----	حس گر نوري
-------	-------	-----	-------	--------	-------	-----------	-------	------------



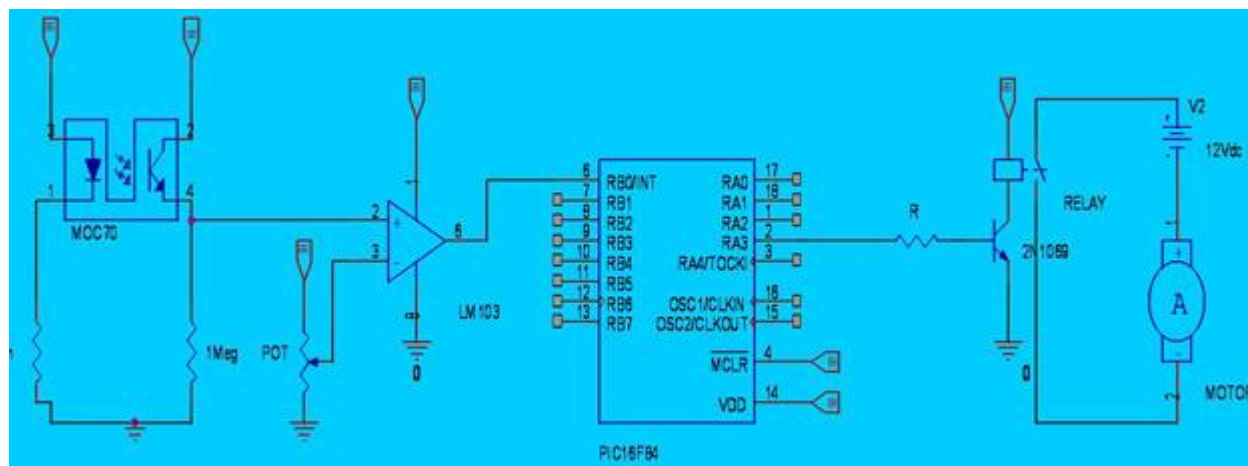
مدار حسگر

مدار مقايسه گر



مدار رله و موتور

موتور	<----	رله	<----	کنترلر	<----	مقایسه گر	<-----	حس گرنوری
-------	-------	-----	-------	--------	-------	-----------	--------	-----------



مبنای دو و مفاهیم آن

رابطه نرم افزار و رباتیک - یادآوری سیستم های شمارش - لزوم طراحی زبان ماشین - تبدیل اعداد به مبنای 2 محاسبات در مبنای 2 - کاربرد مبنای 16 - کدهای اسکي

شباهت ربات با انسان

بدن انسان: مکانیک - مغز و سیستم عصبی: الکترونیک - قوه تعقل: نرم افزار - به وجود آمدن یک نیاز: طراحی زبانی برای درک متقابل میان انسان و کامپیوتر

تشابه سیستم باینری یا مبنای دو با سطح ولتاژ الکتریکی خاموش و روشن

طراحی زبان ماشین با عناصر 0 و 1 در این زبان کلیه داده ها و دستوراتی که انسان به کامپیوتر می دهد، با 0 و 1 شبیه سازی می شوند و کامپیوتر از طریق مدارهای منطقی و الکترونیکی قادر به خواندن و انجام دستورات به این زبان می باشد.

زبان ماشین به منزله زبان مشترک انسان و کامپیوتر

به شرط: الف) تبدیل اطلاعات انسان به زبان 0 و 1 ب) ذخیره 0 و 1 ها و بازیابی اطلاعات در کامپیوتر ها شرط اول:

تبدیل اطلاعات انسان به زبان ماشین یا 0 و 1 - تبدیل اعداد به مبنای دو - تبدیل حروف به کدهای عددی تبدیل کدهای عددی به مبنای دو - طراحی زبانی برای دستورات پردازشی با کلمات کلیدی کد شده در مبنای دو شرط دوم:

ذخیره و بازیابی اطلاعات دودویی انسان و پردازش آنها توسط کامپیوتر

طراحی حافظه های مغناطیسی برای ذخیره دو سطح ولتاژ 0 و 1

طراحی مدارات الکترونیکی برای سیستمهای پردازش و انتقال اطلاعات با دو سطح ولتاژ 0 و 1

تبدیل مبنا ها و محاسبات

تبدیل از دهدهی به دودویی - تبدیل از دودویی به دهدهی - تبدیل دودویی و شانزده شانزدهی - جمع اعداد در مبنای دو متمم 1 و متمم 2 اعداد مبنای 2 - شبیه سازی تفریق با جمع در مبنای 2 - تبدیل از دهدهی به دودویی - تقسیم متوالی بر 2 در هر بار تقسیم باقی مانده را دوباره بر 2 تقسیم می کنیم تا زمانی که خارج قسمت صفر گردد سپس باقیمانده ها بصورت معکوس دنبال هم می آیند.

باقیمانده	خارج قسمت
-----------	-----------

2 / 25	12	1
2 / 12	6	0
2 / 6	3	0
2 / 3	1	1
2 / 1	0	1

تبدیل از دودویی به دهدهی

وزن رقم ها در یک مبنا

در مبناي 10:

یکان: 10^0

دهگان: 10^1

صدگان: 10^2

هزارگان: 10^3 مثال:

$$10^3 \cdot 4 + 10^2 \cdot 3 + 10^1 \cdot 0 + 10^0 \cdot 9 = 4309$$

در مبناي 2:

مشابه روش قبل از سمت راست رقمها عبارتند از

توان دوم عدد 2 از 0 به بالا

برای مثال:

$$110101_2 = 1 \cdot 2^0 + 0 \cdot 2^1 + 1 \cdot 2^2 + 0 \cdot 2^3 + 1 \cdot 2^4 + 1 \cdot 2^5$$

$$110101_2 = 1 + 0 + 4 + 16 + 32$$

$$110101_2 = 53_{10}$$

سیستم شانزده شانزدهی

طبق تعریف این سیستم ارقام 0 تا 15 را می بایستی داشته باشد که برای راحتی ارقام 10 تا 15 را با حروف A تا F انگلیسی نمایش می دهند:

1 2 3 4 5 6 7 8 9

A B C D E F

برای مثال:

$16D_{16}$

$16DA_{16}$

24_{16}

با رعایت وزن ارقام در اینجا نیز داریم:

$$9F5_{16} = 5 \cdot 16_0 + 15 \cdot 16_1 + 9 \cdot 16_2$$

$$9F5_{16} = 5 + 240 + 2304$$

$$9F5_{16} = 2549_{10}$$

استفاده از مبناي دو به جای مبناي شانزده برای راحتی نوشتار
جایگزینی هر چهار رقم مبناي دو با یک رقم مبناي شانزده

$$0000_2 = 0_{16} = 0$$

$$0001_2 = 1_{16} = 1$$

$$0010_2 = 2_{16} = 2$$

$$1101_2 = D_{16} = 13$$

$$1110_2 = E_{16} = 14$$

تبدیل دودویی و شانزده شانزدهی

الف) تبدیل هر چهار رقم مبنای دو به یک رقم مبنای شانزده و برعکس

C <--- 1100

9 <--- 1001

E <--- 1110

1 <--- 0001

$16C9E1 = 1100100111100001_2$

جمع اعداد در مبنای دو

نکته مهم : رقم نقلي

1111
1101
1001
10110

101100

نمایش حروف و کاراکترها در مبنای 2

نمایش استاندارد سیستم اسکی

ASCII

این سیستم در سال 1960 طراحی شد و امروز تمامی چاپگرها و نمایشگرها از آن برای تبادل اطلاعات از طریق کامپیوتر و صفحه کلید استفاده می کنند.

به هر رقم مبنای 10، حروف کوچک و بزرگ، علائم و کاراکترها و... یک کد دودویی اختصاص می دهند که معمولاً در مبنای شانزده نوشته می شود:

7 <--- 37

e <--- 65

E <--- 45

<--- 23

} <--- B7

جمع بندی مطالب این جلسه:

زبان ماشین به عنوان زبان مشترک میان انسان و کامپیوتر طراحی شده و سیستمهای الکترونیکی کامپیوترها، با استفاده از قابلیت تبدیل داده ها و دستورات به 0 و 1 آنها را ذخیره و پردازش می نمایند.
از این جهت نیازمند آشنایی با سیستم دودویی اعداد هستیم.

گیتهای منطقی

لزام استفاده از جبر بول- اصول و قضایای جبر بول - معرفی توابع سوئیچینگ - جدول درستی توابع

پیاده سازی سیستم مبتنی بر 0 و 1 برای کامپیوترهای دیجیتال - استفاده از جبر بول و منطق درست یا نادرست

تعریف جبر بول

سیستمی متشکل از مجموعه ای مانند K با دو یا چند عضو و دو عملگر AND (.) و OR (+) به طوریکه اگر a و b عضو K باشند $a+b$ و $a.b$ نیز عضو مجموعه K باشند.

اصول جبر بول:

الف) عناصر منحصر به فرد 0 و 1 عضو مجموعه K هستند:

$a+0=a$

$a.1=a$

عناصر هماني

(ب) خاصيت جابجايي:

براي هر عضو a و b در K :

$$a+b=b+a$$

$$a.b=b.a$$

(ج) خاصيت شرکت پذيري:

براي هر a, b, c در K :

$$a+(b+c)=(a+b)+c$$

$$a.(b.c)=(a.b).c$$

(د) توزيع پذيري:

براي هر a, b, c در K :

$$a+(b.c)=(a+b).(a+c)$$

$$a.(b+c)=(a.b)+(a.c)$$

(ه) وجود متمم:

براي هر a در K يك عضو منحصر به فرد a^{-} در K وجود دارد به طوريكه:

براي راحتی در نگارش

معمولاً از نوشتن . در عبارتها صرف نظر مي کنند:

$$a.b=ab$$

قضايای جبر بول:

(A) هم ارزي:

$$a+a=a$$

$$a.a=a$$

(B) عضو بي اثر براي . و +

براي عمل +:

$$a+0=a$$

براي عمل .:

$$a.1=a$$

(C) قضيه بازگشت:

=

$$a = a$$

(D) قضيه جذب:

$$a + ab = a$$

$$a(a+b) = a$$

(E) قضيه شبه جذب:

$$a + \overline{ab} = a + b$$

$$a(\overline{a} + b) = ab$$

(G) دمورگان: $\overline{\overline{a+b}} = \overline{a} \cdot \overline{b}$

$$\overline{a.b} = \overline{a} + \overline{b}$$

استفاده از قضایا برای ساده کردن عبارتهای جبر بول

$$\overline{a(b+c)} + \overline{ab} = \overline{ab+ac} + \overline{ab}$$

$$a(b+c) + \overline{ab} = \overline{b+ac}$$

$$a(b+c) + \overline{ab} = \overline{b(ac)}$$

$$a(b+c) + \overline{ab} = \overline{b(\overline{a} + \overline{c})}$$

توابع سوئیچینگ

تابعی از چند متغیر بولی و عملگرهای . و + مانند:

$$f(a, b, c) = ab + ac + bc$$

اگر n متغیر داشته باشیم

هر متغیر دو حالت: 0 و 1

بنابراین: 2 به توان n حالت مختلف از ترکیب ورودی ها داریم

مثال:

a
0
1

a	b
0	0
0	1
1	0
1	1

a	b	c	ab	ac	bc	ab+ac
0	0	0	0	0	0	0
0	0	1	0	0	1	0
0	1	0	0	0	0	0
0	1	1	0	0	1	0
1	0	0	0	0	0	0
1	0	1	0	1	0	1
1	1	0	1	0	0	1
1	1	1	1	1	1	1

جدول درستي

براي هر تابع سوئیچینگ يا مدار منطقي مي توان يك جدول درستي يا جدول مشخصات تعريف كرد كه اين جدول بيان كننده وضعيت مدار خواهد بود.

در جدول درستي تمامي حالتهاي مختلف ورودي هاي تابع را نشان مي دهيم ، سپس به ازاي هر تركيب ورودي بر اساس عملکرد تابع ، خروجي را مشخص مي كنيم. به عبارت ديگر اين جدول بيان كننده عملکرد منطقي تابع و مدار معادل آن است.

فرمهاي متعارف SOP و POS
مینترم ها و ماکسترم ها
گیتهاي منطقي
قطعات الكترونيك گیت ها

فرمهاي متعارف براي نمايش توابع سوئیچینگ :

SOP: جمع جملات کمينه

POS: ضرب جملات بیشينه

SOP جمع حاصلضرب ها

OR كردن عبارتهاي AND شده

$$f(a,b,c) = \overline{a}bc + \overline{b}cd + \overline{a}cd$$

POS ضرب حاصلجمع ها

AND كردن عبارتهاي OR شده

$$f(a,b,c,d) = (\overline{a} + b + c)(\overline{b} + c + d)(a + c + d)$$

جملات کمينه و بیشينه يا :

Maxterm - Minterm

Minterm ها

m

جمله ضرب از همه n متغير يك تابع به صورت متمم يا غيرمتمم در تابع n متغيره 2 به توان n حالت از مینترم ها را داريم
مثال براي 2 متغير:

$$\overline{a}b, \overline{a}\overline{b}, a\overline{b}, ab$$

Maxterm ها

M

جمله جمع از همه n متغیر یک تابع به صورت متمم یا غیرمتمم در تابع n متغیره 2 به توان n حالت از ماکسترم ها را داریم مثال برای 2 متغیر:

$$a+b, a+\bar{b}, \bar{a}+b, \bar{a}+\bar{b}$$

شماره گذاری مینترم ها و ماکسترم ها

تعداد متغیر : n

تعداد مینترم یا ماکسترم : 2 به توان n

mn-1...0m

Mn-1...0M

برای پیدا کردن فرم یک مینترم یا ماکسترم از روی اندیس آن :

(1) عدد اندیس m یا M را به مبنای 2 می بریم

(2) در مینترم به جای 1 خود متغیر و به جای 0 متمم آن را قرار می دهیم

(3) در ماکسترم به جای 0 خود متغیر و به جای 1 متمم آن را قرار می دهیم
مثال:

$$\begin{aligned} M_{13} &\rightarrow 1101 & f(a, b, c, d) : \\ m_5 &\rightarrow 0101 \\ M_{13} &= \bar{a} + \bar{b} + c + d & m_5 &= \bar{a} b \bar{c} d \end{aligned}$$

نمایش استاندارد با مینترم یا ماکسترمهای تابع:

استفاده از SOP یا POS

f(A,B,C) برابر است با:

$$\begin{aligned} &= \bar{A}\bar{B}\bar{C} + \bar{A}B\bar{C} + \bar{A}BC + ABC \\ &= 010 + 110 + 011 + 111 \\ &= m_2 + m_6 + m_3 + m_7 \end{aligned}$$

مدارهای منطقی دیجیتال یا مدارهای سوئیچینگ

ترکیب سری و موازی عناصری به نام گیت

گیت: مسیرهای باز یا بسته سیگنال

به لحاظ ساختار فیزیکی قادرند در طی چند نانو ثانیه روشن یا خاموش شوند

در طراحی الکترونیکی مدارهای منطقی دو استاندارد معروف به کار می رود:

TTL

در این منطق، 5 ولت معادل 1 منطقی می باشد.

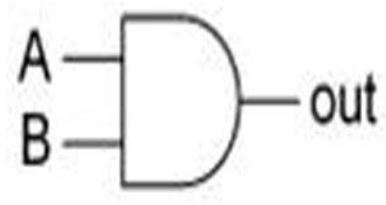
CMOS

در این منطق، 12 ولت معادل 1 منطقی می باشد.

انواع گیت :

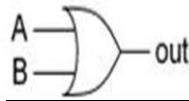
گیت AND :

همانطور که از نامش پیداست مانند "و" رفتار می کند یعنی در صورتی که یکی از ورودیهای آن 0 باشد خروجی آن صفر خواهد بود.



IN	IN	OUT
0	0	0
0	1	0
1	0	0
1	1	1

گیت OR :



IN	IN	OUT
0	0	0
0	1	1
1	0	1
1	1	1

گیت NOT :

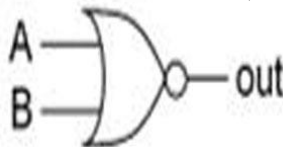
این گیت در ازای ورودی 0 یا 1 معکوس آن را به خروجی می فرستد.

OUT=1 <---- IN=0

OUT=0 <---- IN=1

گیت NOR :

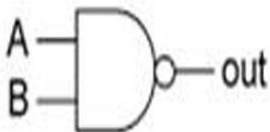
این گیت به عنوان یک المان منطقی ساده، عمل دو گیت OR و NOT را با هم ادغام کرده، در یک گیت نشان می دهد و شامل دو یا چند ورودی و یک خروجی می شود.



IN	IN	OUT
0	0	1
0	1	0
1	0	0
1	1	0

گیت NAND :

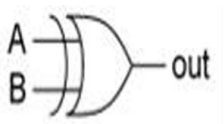
این گیت به عنوان یک المان منطقی ساده، عمل دو تابع AND و NOT را با هم ادغام کرده، و در یک گیت نشان می دهد. این مدار شامل دو یا چند ورودی و یک خروجی است.



IN	IN	OUT
0	0	1
0	1	1
1	0	1
1	1	0

گیت XOR :

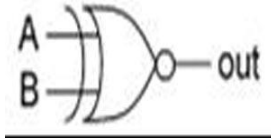
این گیت شامل دو یا چند ورودی و یک خروجی است. در گیت XOR در صورتی خروجی 1 می شود که فقط یکی از ورودیهای 1 باشد.



IN	IN	OUT
0	0	0
0	1	1
1	0	1
1	1	0

گیت XNOR:

این گیت شامل دو یا چند ورودی و یک خروجی است در گیت XNOR در صورتی که خروجی یک می شود که یا هر دو ورودی صفر و یا هر دو ورودی یک باشد.



IN	IN	OUT
0	0	1
0	1	0
1	0	0
1	1	1

قطعات الکترونیکی
گیت های منطقی

AND

OR

NOT

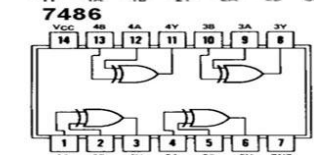
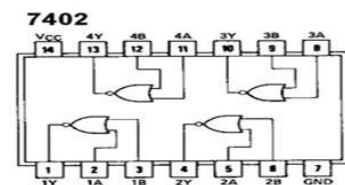
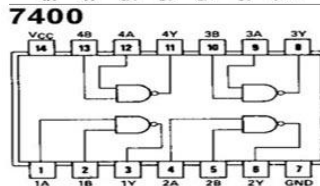
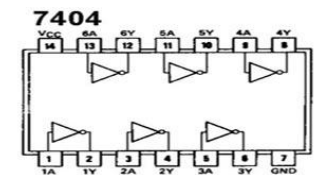
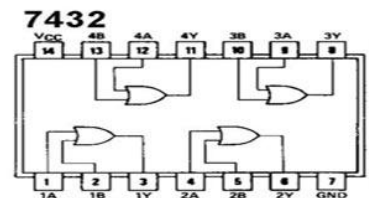
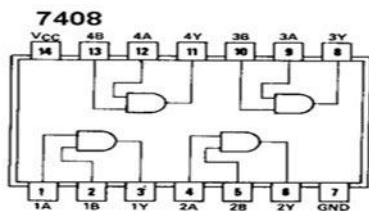
NAND

NOR

XOR

مدارهای ترکیبی

مفهوم مدار منطقی و مدارهای ترکیبی
ایجاد تابع یک مدار منطقی
مثالهایی از مدارهای منطقی



طراحی نیم افزاینده
طراحی مدار رأی گیری

مدار منطقی

هر مدار منطقی شامل ترکیبی از گیت‌های منطقی است .
برای بررسی عملکرد هر مدار منطقی باید رفتار آن به ازای همه ورودی‌های ممکن مدار را بیابیم.

مدارات منطقی ترکیبی

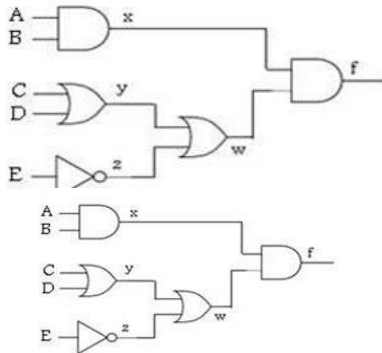
مداراتی هستند که خروجی آنها در هر لحظه فقط به ورودی‌های همان لحظه بستگی دارد. به بیان دیگر یک مدار ترکیبی، مداری است بدون عنصر حافظه‌ای برای ذخیره وضعیت قبلی مدار.

$$F = X.W$$

$$W = Y + Z$$

$$X = A.B$$

$$F = (A.B).(\bar{E} + (C + D))$$

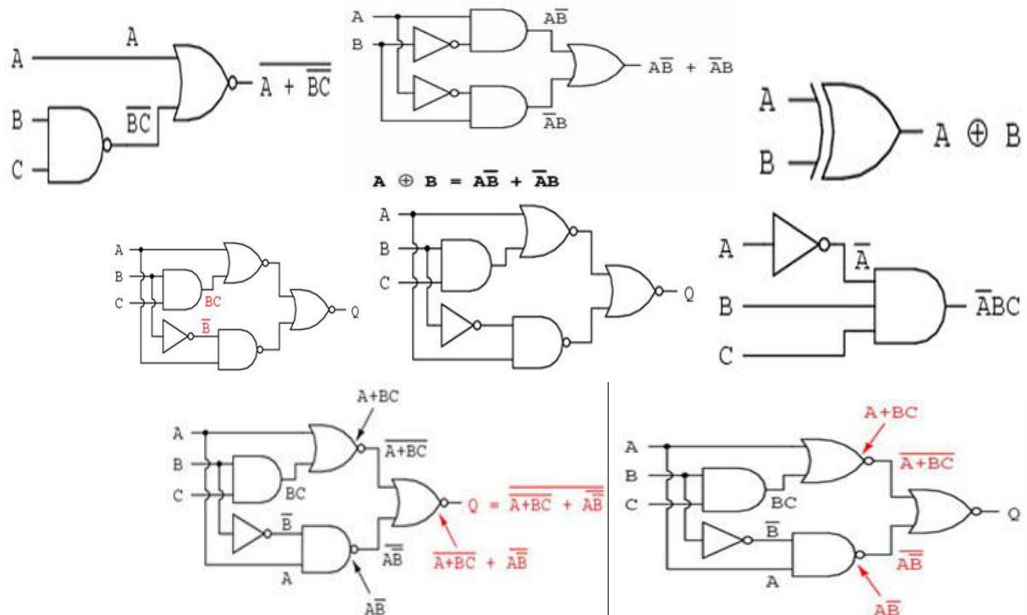


جدول درستی در مدارات ترکیبی

برای هر مدار منطقی می‌توان یک جدول درستی تعریف کرد که تمامی حالت‌های مختلف ورودی‌های مدار را نشان می‌دهد، سپس به ازای هر ترکیب ورودی بر اساس عملکرد مدار خروجی را مشخص می‌کنیم.
جدول درستی بیان‌کننده عملکرد منطقی مدار است.

ایجاد تابع منطقی مدار

برای طراحی یک مدار باید با استفاده از جدول درستی آن یک تابع منطقی بدست آورد.
در ایجاد تابع منطقی از جبر بول استفاده می‌شود.



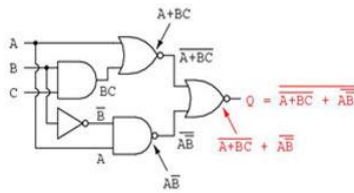
$$A + BC + AB - A + BC.(AB)$$

$$(A + BC)(\bar{A}\bar{B})$$

$$A\bar{A}B + BC\bar{A}B$$

$$\bar{A}B + 0$$

$$\bar{A}B$$



طراحی مدا نیم جمع کننده

Half Adder

برای جمع دو عدد یک رقمی مبنای دو

	x	y	c	s
0	0	0	0	0
1	0	1	0	1
2	1	0	0	1
3	1	1	1	0

X

+ y
c s

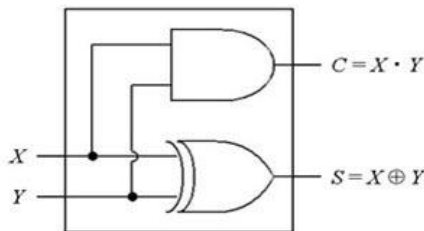
تابع متناظر با S :

$$f(x, y) = m_1 + m_2$$

$$f(x, y) = 01 + 10$$

$$f(x, y) = \bar{x}y + x\bar{y}$$

$$f(x, y) = x \oplus y$$



تابع متناظر با C :

$$xy = 3 f(x, y) = m$$

مثالی دیگر از طراحی یک مدار ترکیبی

یک خانواده 4 نفره برای انجام کاری رأی گیری می کنند. ارزش رأی پدر و مادر 2 برابر رأی فرزندان است. مداری طراحی کنید که هر نفر با فشردن کلیدی رأی دهد و در صورتی که مجموع ارزش آراء حداقل 4 شد ، یک چراغ به نشانه جواب مثبت رای گیری روشن شود.

$$f(A, B, x, y) = \sum m(7, 11, 12, 13, 14, 15)$$

$$f(A, B, x, y) = m_7 + m_{11} + m_{12} + m_{13} + m_{14} + m_{15}$$

$$f(A, B, x, y) = \bar{A}BCD + \bar{A}BC\bar{D} + A\bar{B}CD + A\bar{B}C\bar{D} + AB\bar{C}D + ABC\bar{D}$$

$$+ AB\bar{C}\bar{D} + ABC\bar{D} + ABCD$$

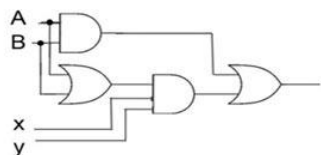
$$f(A, B, x, y) = BCD + ACD + \bar{A}BC + \bar{A}BC$$

$$f(A, B, x, y) = (A + B)CD + \bar{A}B$$

خلاصه جدول

درستی

	A	B	x	y	R	f
7	0	1	1	1	4	1
11	1	0	1	1	4	1
12	1	1	0	0	4	1
13	1	1	0	1	5	1
14	1	1	1	0	5	1
15	1	1	1	1	6	1



معرفی برنامه نویسی کامپیوتر و زبان Basic

نرم افزار و برنامه کامپیوتری

زبانهای برنامه نویسی

حل مسأله و الگوریتم

ساختارهای شرط و تکرار

نرم افزار
تعیین و کنترل عملیاتی که کامپیوتر بایستی انجام دهد

برنامه کامپیوتر
دستورالعمل های قدم به قدم برای اجرا توسط کامپیوتر

انواع نرم افزار

نرم افزارهای سیستم
نرم افزارهای کاربردی
نرم افزارهای سیستم:
برقراری ارتباط بین برنامه های کاربردی ، کاربر ، سخت افزار
برنامه های راه انداز
سیستم عامل
برنامه های کمکی
مترجم های زبان های برنامه نویسی
نرم افزارهای کاربردی :
عمومی : واژه پرداز ها ، مرورگرهای اینترنت و...
تخصصی : گرافیکی ، ایجاد صفحات اینترنتی و...
آموزشی و مراجع

زبانهای برنامه نویسی:

برقراری ارتباط بین کامپیوتر و انسان برای اجرای الگوریتم ها توسط سخت افزار

انواع زبانهای برنامه نویسی:

زبانهای سطح پایین
وابسته به سخت افزار و نزدیک به آن
زبانهای سطح بالا
نزدیک به زبان محاوره ای

مترجم زبان:

نرم افزارهای که برنامه نوشته شده به یک زبان برنامه نویسی را به زبان قابل فهم برای کامپیوتر یعنی زبان ماشین تبدیل می کند.

بخشی از یک برنامه به زبان ماشین:

F018E00FF308E07031C8F07031CEA1000000028288:
D00DF300F2003288D01E83EB810001000232803308:
C008D09FC30031C18288C070318152838100020008:
C0764008D0F15280C181E288C1C222894100030008:
B11000400000002228080083130313831264000800:
F00F43002203C10005000061483160610831201308:
F00F43002203010006000061083160610831201308:

نسل اول: زبان ماشین

وابسته به سخت افزار

0101----< دستور جمع

0001----< اولین عملوند

0010----< دومین عملوند

0011----< آدرس حاصل

نسل دوم : زبان اسمبلی

- استفاده از حروف و اختصارات به جای 0 و 1

- هر دستور مستقیم به یک دستور زبان ماشین تبدیل می شود

- مترجم آن اسمبلر نام دارد

ADD x y R مثال :

نسل سوم: زبان های سطح بالا

...,BASIC,C,C++,Pascal

همه منظوره و برای حل مسائل عمومی

$A=B+C$

نسل چهارم :

زبانهای تولید برنامه های کاربردی برای حل مسائل خاص

SQL برای جستجو در بانک اطلاعاتی و تهیه گزارش

نسل پنجم :

زبانهای طبیعی

زبان انسان

در دست ایجاد

مترجم های زبانهای سطح بالا :

1. کامپایلر ها

تمام برنامه یکجا ترجمه می شود

2. مفسر ها

برنامه خط به خط ترجمه می شود

برنامه نویسی

هدف: حل یک مسأله

تعریف مسأله

طرح روش گام به گام حل

تبدیل به یک زبان

آزمایش

ذخیره سازی

به روز رسانی و نگهداری

تعریف مسأله:

هدف مسأله

ورودی ها

خروجی ها

روش پردازش داده های ورودی

الگوریتم :

مجموعه مراحل لازم برای حل مسأله و دریافت خروجی های مطلوب از ورودی ها به صورت گام به گام

به هر زبانی می توان نوشت

شروع

نمره های یک دانش آموز را بگیر

معدل او را حساب کن

معدل را به ما بگو

پایان

ساختارهای کنترلی

توالی و ترتیب : دستورالعمل ها پشت سر هم

شروع

$A=5$

$$B=A*5$$

پایان

متغیر:

مکانی از حافظه کامپیوتر که مقداری را در آن ذخیره می کنیم. $A=5$

انتخاب :

در مواقعی که نیاز به تصمیم گیری در شرایط مختلف باشد
شروع

عدد ی را از ورودی بگیر

اگر زوج بود چاپ کن

اگر فرد بود در 2 ضرب و چاپ کن

پایان

تکرار و حلقه :

تکرار و حلقه :

نیاز به تکرار یک رشته دستورالعمل

به تعداد مشخص

تا زمانی که وضعیت خاصی برقرار باشد

شروع

$$i=1$$

اگر $i=5$ برو به خط 7

کلمه "Hi" را چاپ کن

$$i=i+1$$

برو به خط 3

پایان

مثال:

الگوریتمی که مجموع اعداد قابل قسمت بر 3 را که بین 0 تا 50 باشند بیابد و نمایش دهد.

شروع

$$A=1$$

اگر $A=51$ برو به خط 9

$$A=A+1$$

$$L=A/3$$

اگر L اعشاری نبود : $B=B+A$

برو به خط 3

حاصل را چاپ کن

پایان

شروع

تا زمانی که سنسور 1 علامت نداده است مستقیم حرکت کن

اگر سنسور فشار داده شد در خلاف جهت قبل حرکت کن

تا زمانی که سنسور 2 فشار داده نشده مستقیم حرکت کن

اگر سنسور فشار داده شد در خلاف جهت قبل حرکت کن

پایان

خلاصه ای از آموزش زبان QBASIC و برنامه نویسی با پورت ها

ویژگی های زبان های برنامه نویسی

ذخیره انواع داده ها و اطلاعات در آدرسهای مشخص حافظه

قابلیت دریافت اطلاعات و نمایش حاصل پردازش

انجام عملیات و محاسبات بر روی داده ها

کنترل ترتیب اجرای برنامه طبق الگوریتم برنامه

انواع داده ها در زبان Basic

انواع داده ها در زبان Basic

داده های ثابت

داده های متغیر

داده های ثابت:

اعداد: 4.5, 673, 5

رشته ها: "673", "Tehran"

داده های متغیر:

عددی: A, Sum

رشته ای: A\$, Name\$

آشنایی با بعضی دستورات زبان Basic

PRINT

PRINT "Hi Hamed"

PRINT "15+1"

PRINT 15+1

PRINT a

اولویت محاسبات ریاضی:

() : 1

^ : 2

/, * : 3

\ : 4

MOD : 5

-, + : 6

مثال:

$(x*c) / 2 + s - a / b$

INPUT

INPUT A

INPUT A,B,C

INPUT Q\$

LET

LET A=5

همواره مقدار سمت راست در متغیر سمت چپ قرار می گیرد

توابع کتابخانه ای

برنامه هایی که از قبل نوشته شده اند و وظیفه خاصی را انجام می دهند

ABS (x) قدر مطلق

SQR(x) ریشه دوم

عملگرهای منطقی

AND

OR

NOT

عبارات شرطي

THEN...شرط... IF

دستور يا دستورات

END IF

INPUT A

THEN 100>IF A

PRINT A

END IF

THEN... شرط ... IF

دستور يادستورات

ELSE

دستور يا دستورات

END IF

حلقه هاي تکرار

GOTO Lable

Sum: INPUT A

B=B+A

GOTO SUM

حلقه :FOR

شمارش تعداد دفعات تکرار

FOR شمارنده=..... TO

دستورات تکرار شدي

NEXT شمارنده

FOR i = 1 TO 10

PRINT " my name is Hamed"

Next i

دو دستور ساده :

CLS : پاک کردن صفحه نمایش در بخش اجرا

END : پایان برنامه

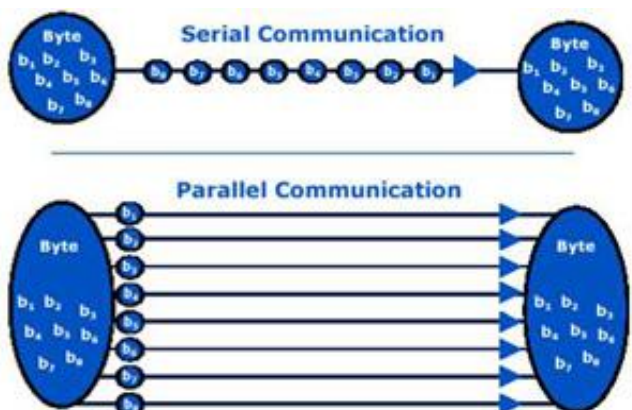
آشنایی با پورت

آشنایی با پورت

پورت يا درگاه محل اتصال وسایل و تجهيزات جانبي به يك کامپيوتر است که در واقع دروازه ورود يا خروج 0 و 1 ها مي باشد .

هر پورت داراي چند پایه يا پين است که هر پين به وسيله یک سیم ، گذرگاہي براي عبور یک واحد داده مي باشد.

نحوه ذخيره شدن داده ها در کامپيوتر:



قابليت نگهداري 0 يا 1 Bit :

Byte : 8 Bit

روشهاي انتقال اطلاعات در کامپيوتر

انتقال سري

انتقال موازي

برخي از انواع پورتها :

پورت موازي يا LPT

پورت سريال يا COM

پورت USB

پورت سريال

داده ها را بصورت سريال (دنبال هم) ارسال و يا دريافت مي کند . در چنين حالي يك بابت از اطلاعات بصورت هشت بيت ويكي پس از ديگري ارسال خواهند گرديد.

مثال : اتصال مودم

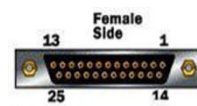


مزيت :

استفاده از يك سيم براي ارسال و دريافت داده

عيب:

سرعت پايين ارسال اطلاعات



پورت موازي

در هر لحظه هشت بيت را از طريق هشت پين جداگانه ارسال يا دريافت مي کند . پورت موازي استاندارد قادر به ارسال 50 تا 100 كيلوبايت در هر ثانيه است.

زمانيكه كامپيوتر اطلاعاتي را براي چاپگر و يا هر وسيله ديگري كه به پورت موازي متصل است ، ارسال مي نمايد ، در هر لحظه هشت بيت ارسال خواهد شد .

پين دوازده :

در صورتيكه چاپگرداراي كاغذ نباشد ، از طريق پين شماره دوازده به كامپيوتر آگاهي لازم داده خواهد شد .

GND

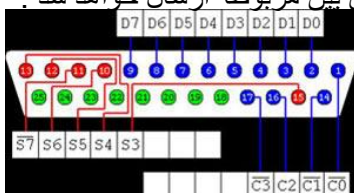
پين هاي شماره هيجده تا بيست و پنج (Ground) زمين هستند و به عنوان مرجع ولتاژ براي پين هاي ديگر استفاده مي شوند.

پين دو تا نه :

حامل داده هستند .

بمنظور مشخص نمودن اينكه يك بيت داراي مقدار يك است ولتاژ پنج ولت از طريق پين مربوطه ارسال خواهد شد .

برروي پيني كه شامل مقدار (داده) صفر است ولتاژي قرار نخواهد گرفت .



آدرس پورت ها

خانه هاي حافظه و پورت هاي انتقال داده در كامپيوترداراي يك آدرس مشخص مي باشند كه قابل دسترسي توسط

برخي زبانهاي برنامه نويسي براي ايجاد تغيير يا خواندن و نوشتن در آنها ، هستند. اين آدرس ها براي هر 8 بيت يا يك بابت مشخص شده اند .

آدرس بابت داده پورت LPT1

H 88dec378

دستورات انتقال داده از پورت در زبان BASIC

OUT آدرس , مقدار

INP (آدرس)

مثال:

H378 , 255 & OUT

H378 &) A = INP(

يك آزمائش برنامه نويسي با پورت LPT

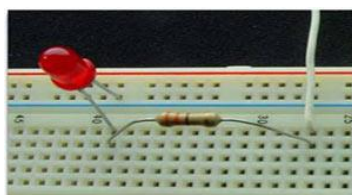
رعائت نکات مهم :

فقط در زمان خاموش بودن كامپيوتر، تجهيزات را به اين پورت متصل يا جدا كنيد . ولتاژهاي ورودي نبايد از 5 ولت

بيشتر شوند و يا از صفر کمتر

پورت موازي در مقابل جريان بيش از حد محافظت نشده

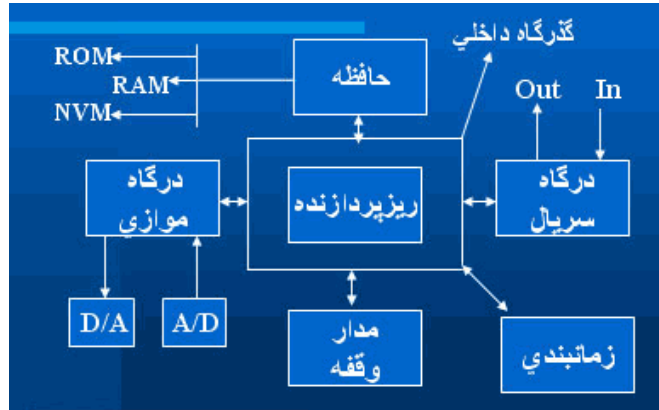
اتصال يك LED به هر 8 پين بابت داده



بنابراین : با امکان فراگیری برنامه نویسی با سخت افزار قادر به کنترل یک مدار الکترونیکی خواهیم بود و به این ترتیب رابطه میان نرم افزار و سخت افزار الکترومکانیکی ربات امکان پذیر خواهد شد.

میکروکنترلر

سیستم های مبتنی بر میکروکنترلر
بلوک دیگرام سیستم مبتنی بر میکروکنترلر
ملزومات سخت افزاری و اجزاء داخلی



سیستمهای مبتنی بر ریزپردازنده ها

ماشین حساب جیبی
ساعت های دیجیتال
دستگاه های خودپرداز
بازی های بصری
کامپیوتر خانگی

بلوک دیگرام سیستم مبتنی بر میکروکنترلر

ملزومات سخت افزاری و اجزاء داخل

ثبات ها (Register)
درگاهها (Port) ورودی خروجی (I/O)
وقفه (Interrupt)
زمان سنج/شمارنده (Timer/counter)
مبدل های D/A & A/D

ثبات ها (Register)

تعریف ثبات :

ثبات ها در حقیقت آرایه ای از سلول ها هستند که مقدار 1 یا 0 می گیرند در حقیقت یک ثبات حاوی یک داده چند بیتی می باشد

انواع ثبات ها

ثبات با امکان بار شدن موازی

شیفت ثبات ها

شیفت ثبات های دوطرفه

شمارنده ها

درگاه های ورودی خروجی (I/O Port)

آدرس دهی درگاه ها

مشخصات عملی درگاه ورودی خروجی

انواع انتقال داده

تعریف درگاه

درگاه در حقیقت کانالی است که از طریق آن میکروکنترلر با دنیای خارج خود ارتباط برقرار می کند

آدرس دهی درگاه ها

آدرس دهی :

نامگذاری هر یک از راه های ارتباطی است که میکروکنترلر توسط آن درگاه ها را از هم تفکیک می کند

مشخصات عملی درگاه ها

درگاه ورودی بافر شده :

درگاهي است كه توسط يك بافر سه وضعيتي كنترل شده و اطلاعات آن فقط در زمان مقتضي به ميكروكنترلر منتقل مي شود

مشخصات عملي درگاه ها

درگاه ورودي قابل قفل شدن :

درگاهي است كه خروجي داده شده توسط ميكروكنترلر را تا زماني كه لازم است حفظ مي كند

انواع روشهاي انتقال داده

انتقال موازي داده

انتقال سري داده

مفهوم وقفه

مفهوم وقفه

وقفه در حقيقت درخواستي است كه توسط يكي از اجزاء داخلي يا خارجي مطرح مي گردد و ميكروكنترلر با توجه به پيش بيني قبلي در زمان مناسب به آن پاسخ مي دهد

انواع وقفه

سخت افزاري:

درخواستي كه توسط يكي از اجزاء سيستم مطرح مي شود مانند يك كليد فشاري

نرم افزاري:

درخواستي كه توسط يكي از قسمت هاي برنامه اجرايي مطرح مي گردد.

زمان سنج / شمارنده

انواع زمان سنجي:

نرم افزاري

سخت افزاري

مبدلهاي D/A & A/D

مفهوم داده آنالوگ و ديگيتال

مبدل آنالوگ به ديگيتال A/D

مبدل ديگيتال به آنالوگ D/A

مفهوم داده آنالوگ و ديگيتال

داده آنالوگ:

داده اي كه مي تواند به طور پيوسته هر مقداري را اختيار كند مانند ولتاژ برق شهر كه در هر تناوب مقداري بين 220- و 200+ را اختيار مي كند

داده ديگيتال: داده اي كه فقط مقادير خاصي را مي تواند داشته باشد مانند نمايشگر كانال تلويزيون كه يك عدد صحيح را هميشه نشان مي دهد

مبدل آنالوگ به ديگيتال A/D

ابزاري است كه با توجه به مقادير ابتدائي و انتهائي داده آنالوگ و ميزان بيتي كه به آن اختصاص داده شده است به هر بازه از مقدار پيوسته ، يك عدد ديگيتال نسبت مي دهد مثال

0 تا 220 ولت با نمايش 8 بيتي:

$$220/256 = 0/85$$

هر يك عدد در مبناي دودويي برابر 0/85 ولت است

$$6.8 = 0.85 * 8 = (00001000)_2$$

مبدل ديگيتال به آنالوگ D/A

ابزاري است كه با توجه به مقادير ابتدائي و انتهائي داده آنالوگ و ميزان بيتي كه به آن اختصاص داده شده است ، به هر بازه از مقدار ديگيتال يك عدد آنالوگ نسبت مي دهد

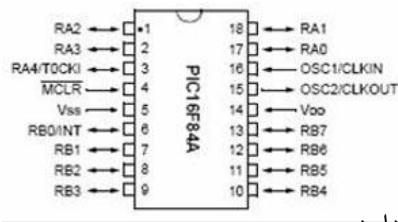
Programmable Interface Controller PIC

انواع PIC
Flash Program Memory
Eprom Program Memory
Rom Program Memory
EEPROM Program Memory

Flash Program Memory

سرعت بالاي خواندن اطلاعات از ROM مزيت اين خانواده محسوب مي شود. بطوريكه سرعت 12 پالس براي اجراي برنامه در ميكروكنترلر 8051 به 4 پالس در PIC رسيده است. اين سرعت براي AVR به يك پالس رسيده است.

PIC16F84A



ويژگيهاي IC16F84A

برنامه ريزي با تنها 35 دستور
اجراي همه دستورات با يك سيكل بجز دستور پرش كه به دو سيكل نياز دارد
سرعت كاري :
فرکانس كاري DC تا 20 MHz (حداكثر سرعت 200 نانوثانيه براي هر سيكل كاري)
حافظه برنامه ريزي 1024 كلمه

68بايت RAM

64بايت EEPROM

طول دستور 14 بيت

وقفه خارجي

زمان سنج 8 بيتي

ويژگيهاي جانبي PIC16F84A

13 پين ورودي/ خروجي با قابليت برنامه ريزي مجزا
تأمين جريان كشي مورد نياز 25 LED ميلي آمپر براي هر پين
حافظه Flash با قابليت برنامه ريزي 10000 بار
EEPROM با قابليت 10.000.000 بار نوشتن
برنامه ريزي آسان

Name	Bit	Buffer Type	Function
RA0	bit0	TTL	Input/output
RA1	bit1	TTL	Input/output
RA2	bit2	TTL	Input/output
RA3	bit3	TTL	Input/output
RA4/T0CKI	bit4	ST	Input/output or external clock input for TMR0. Output is open drain type.

Legend: TTL = TTL input, ST = Schmitt Trigger input

درگاه A

درگاه B

Name	Bit	Buffer Type	I/O Consistency Function
RB0/INT	bit0	TTL/ST ⁽¹⁾	Input/output pin or external interrupt input. Internal software programmable weak pull-up.
RB1	bit1	TTL	Input/output pin. Internal software programmable weak pull-up.
RB2	bit2	TTL	Input/output pin. Internal software programmable weak pull-up.
RB3	bit3	TTL	Input/output pin. Internal software programmable weak pull-up.
RB4	bit4	TTL	Input/output pin (with interrupt-on-change). Internal software programmable weak pull-up.
RB5	bit5	TTL	Input/output pin (with interrupt-on-change). Internal software programmable weak pull-up.
RB6	bit6	TTL/ST ⁽²⁾	Input/output pin (with interrupt-on-change). Internal software programmable weak pull-up. Serial programming clock.
RB7	bit7	TTL/ST ⁽²⁾	Input/output pin (with interrupt-on-change). Internal software programmable weak pull-up. Serial programming data.

Legend: TTL = TTL input, ST = Schmitt Trigger.

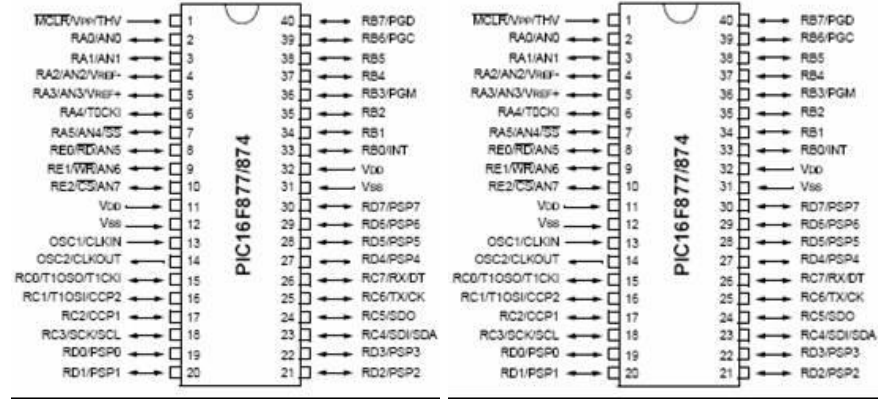
وقفه

وقفه خارجي

زمان سنج

تغيير در منطق ولتاژي پايه هاي درگاه B
اتمام سيكل نوشتن داده در EEPROM

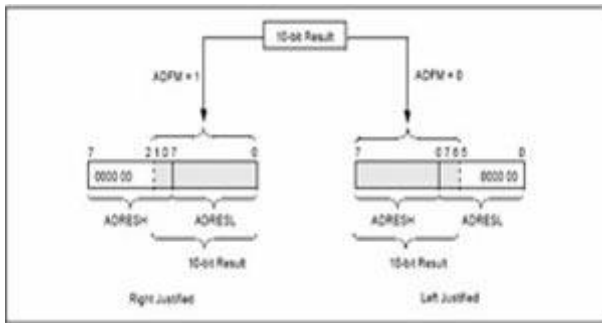
PIC16F877A



PIC16F877A ویژگی‌های

زمان سنج 0 : 8 بیتی
 زمان سنج 1 : 16 بیتی
 زمان سنج 2 : 8 بیتی
 8 کانال میدل آنالوگ به دیجیتال : 10 بیتی
 درگاه موازی
 درگاه سریال

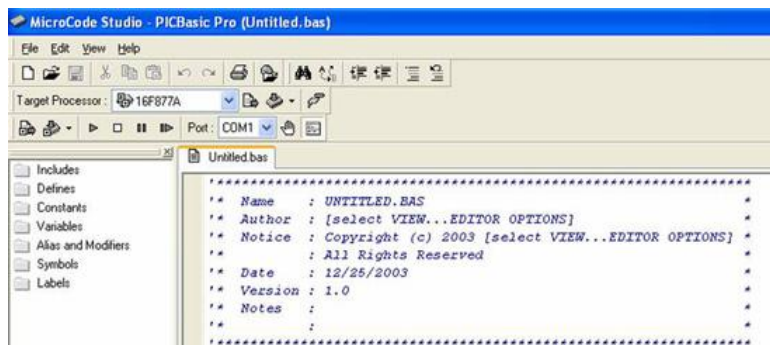
ADCON0 REGISTER



ثبات خروجی A2D

آشنایی با نرم افزار PICBasic

زبان اسمبلی: 37 دستورالعمل
 زبان سطح بالا
 BASIC
 C



محیط نرم افزار

آشنایی با دستورات

نامگذاری خطوط برنامه (Line Labels)
 here:Serout ,N2400,["Hello world!",13,10 Goto here]
 متغیرها (Variables)
 Label VAR Size{.Modifiers}
 dog var byte

cat var bit
w0 var word
Aliases(اسم مستعار)
fido var dog
b0 var w0.byte0
Arrays(آرایه)
Label VAR Size[Number of elements]
sharks var byte[10]

Size	Maximum number of elements
Bit	256
Byte	*96
Word	*48

Constants(مقادیر ثابت)
Label CON Constant expression
mice con 3
Symbols(نشانه)
SYMBOL lion = cat
Numeric Constants(ثابتهای عددی)

100
100%
100\$
Pins
PORTB.1 = 1

No. PIC micro Pins	0-7	8-15
8-pin	GPIO*	GPIO*
18-pin	PORTB	PORTA*
28-pin(except 14 C000)	PORTB	PORTC
28-pin(14C000)	PORTC	PORTD
40-pin	PORTB	PORTC

INCLUDE
با استفاده از این دستور می توان از فایل های منبع که نوشته شده است در برنامه استفاده کرد
DEFINE
بعضی از موارد نظیر پین های مربوط به ارتباط با LCD از طریق دستور های BASIC از پیش مشخص هستند
که این موارد با دستور DEFINE تعیین می شوند
DEFINE LCD_DREG PORTA 'LCD data port

دستورالعمل های ریاضی
W1 = W0 * 1000
W1 = W0 / 1000
3 >> B0 = B0
1 << W1 = W0
SIN & COS
B0 = SQR W1

عملگر های مقایسه

Comparison	Description
------------	-------------

operator	
= or ==	Equal
<> or !=	Not Equal
<	Less than
>	Greater than
<=	Less than or Equal
>=	Greater than or equal

عملگر های منطقی

Logical operator	Description
And or &&	logical and
OR or	logical or
Xor or ^ ^	logical exclusive or
Not And	logical nand
Not Or	logical nor
Not Xor	logical Nxor

BUTTON

Pin,Down,Delay,Rate,BVar,Action,Label

Pin

شماره پینی که سویچ به آن وصل است Down

وضعیت پین وقتی سویچ فشار داده می شود Delay

عددی بین 0 و 256 که تعداد سیکل تکرار را قبل از خواندن مجدد چک می کند

BUTTON

Rate

نرخ تکرار

BVar متغیری به صورت بایت که برای شمارش مورد نیاز است

Action مقدار 0 یا 1 و نمایشگر حالت سویچ برای عملکردن است

Lable مکانی از نرم افزار که در صورت فشردن دکمه باید اجرا شود

COUNT

COUNT Pin,Period,Var

این دستور تعداد پالسهای رخ داده شده در pin را در مدت زمان period که بر حسب میلی ثانیه می باشد در

متغیر var ذخیره می کند

END

این دستور اجرای برنامه را متوقف

می کند

FOR..NEXT

FOR Count = Start TO End {STEP {-} Inc}

{Body}

NEXT {Count}

مثال

FOR i = 1 TO 10

Toggle PORB.1

Pause 500

NEXT i

GOSUB
GOSUB Label

این دستور باعث اجرای يك زیر برنامه می شود که label مشخص می کند البته مکان فعلی ذخیره می شود و بعد از پایان اجرای زیر برنامه با دستور return عمل بازگشت صورت می گیرد

GOTO
GOTO Label

این دستور اجرای برنامه را از جایی که label مشخص می کند پی می گیرد

low & HIGH

HIGH Pin

روی Pin مشخص شده مقدار 1 دیجیتال بارگذاری می کند

LOW Pin

روی Pin مشخص شده مقدار 0 دیجیتال بارگذاری می کند

IF..THEN

IF Comp {AND/OR Comp...} THEN Label

IF Comp {AND/OR Comp...} THEN

Statement...

ELSE

Statement...

ENDIF

If Pin0 = 0 Then pushed

Then old 40 =< If B0

If (B0 = 10) AND (B1 = 20) Then loop

Then 10 <> If B0

B0 = B0 + 1

B1 = B1 - 1

Endif

IF ... THEN... ELSE

If B0 = 20 Then

led = 1

Else

led = 0

Endif

OUTPUT& INPUT

INPUT Pin

OUTPUT Pin

پین مشخص شده را بصورت ورودی یا خروجی تعریف می کند

LCDOUT

LCDOUT Item{,Item...}

LCDOUT \$FE, 1, "Hello"

hello نمایش کلمه

LCDOUT

PAUSEUS & PAUSE

PAUSE Period

در اجرای برنامه وقفه ای به مدت زمان مشخص شده در Period که بر حسب میلی ثانیه می باشد ایجاد می کند

PAUSEUS

Command	Operation
\$FE, 1	Clear display
\$FE, 2	Return home (beginning of first line)
\$FE, \$0C	Cursor off
\$FE, \$0E	Underline cursor on
\$FE, \$0F	Blinking cursor on
\$FE, \$10	Move cursor left one position
\$FE, \$14	Move cursor right one position
\$FE, \$C0	Move cursor to beginning of second line
\$FE, \$94	Move cursor to beginning of third line
\$FE, \$D4	Move cursor to beginning of fourth line

کند

POT Pin,Scale,Var

این عدد باید برای rc کوچک مقدار بالا (256) و برای rc بزرگ مقدار پایین داشته باشد

SERIN Pin,Mode,{Timeout,Label,{[Qual...],{Item...

SEROUT Pin,Mode,[Item{,Item...}]

حالت‌های کاری ارسال سریال

Mode	Mode No.	Baude Rate	State
T2400	0	2400	True
T1200	1	1200	
T9600	2	9600	
T300	3	300	
N2400	4	2400	Inverted
N1200	5	1200	
N9600	6	9600	
N300	7	300	

SERIN 1,N2400,['A'],B0

```
SEROUT 0,N2400,[#B0,10]
```

WHILE..WEND

WHILE Condition

Statement...

WEND

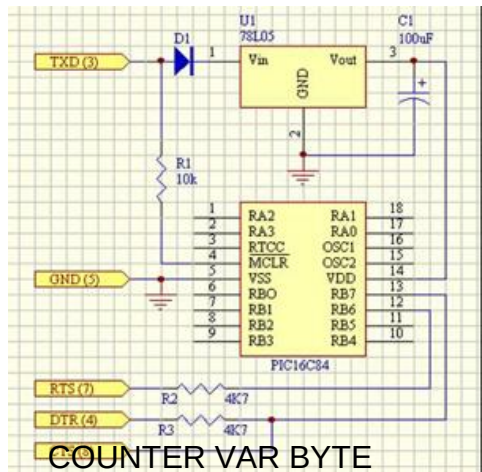
$$i = 1$$

10 => WHILE i

```
Serout 0,N2400,["No:",#i,13,10] i = i + 1
```

WEND

مدار برنامه ریز



COUNTER VAR BYTE

BEGIN

COUNTER =0

PORTB=0

[HOME](#)

BUTTON 9,0,255,1,COUNTER,1,START

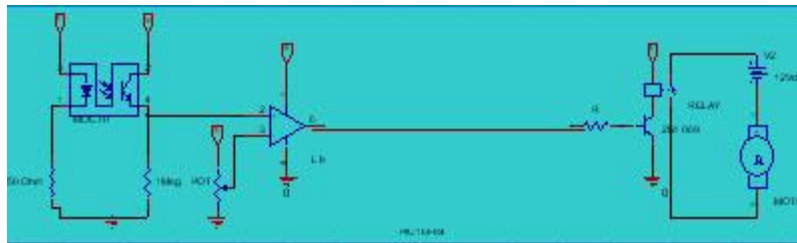
برنامه LED چشمک زن

```
START
WHILE PORTA.1=0
LOW PORTB.4
high poRtB.5
LOW poRtB.6
high poRtB.7
pause 2000

high PORTB.4
low PORTB.5
high PORTB.6
low PORTB.7

pause 2000
WEND
goto BEGIN
END
```

موتور	<----	رله	<----	Driver	<----	مقایسه گر	<----	حس گر
-------	-------	-----	-------	--------	-------	-----------	-------	-------



رسم سه نما

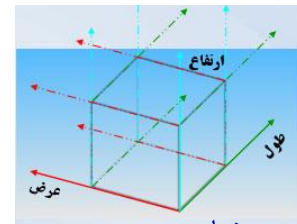
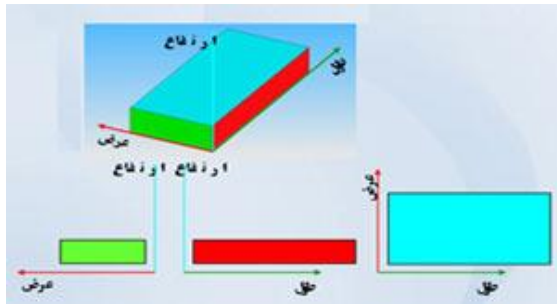
در هر نما چه جزئیاتی نمایش داده می شود؟- کدام نما ها را انتخاب می کنیم؟ - نحوه نمایش سه نما

انتخاب نما ها

در اشکال هندسی چند وجه می تواند وجود داشته باشد؟

انتخاب نما ها

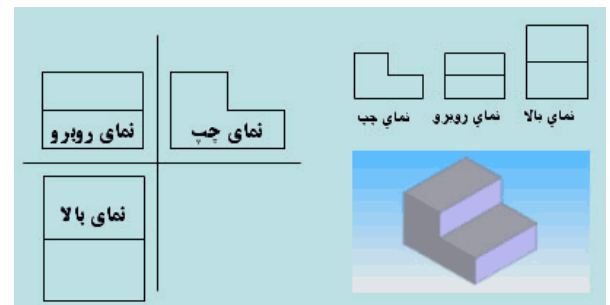
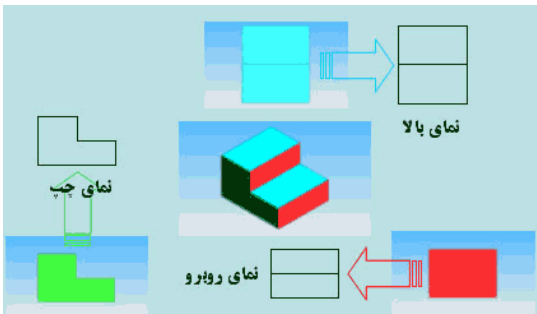
راستاهای اصلی کدام است؟



نحوه نمایش سه نما

نمای روبرو	نمای چپ
نمای بالا	

رسم سه نما - مثال



نمونه نقشه

روش های ساخت- اجزاء مکانیکی ربات -

تعریف

دسته بندی روش ها- روشهای حذفی - ماشین کاری - روش های غیر حذفی - قالب گیری - روش های ترکیبی - فلز کاری

مواد مصرفی

اهمیت مواد مصرفی- انواع مواد مصرفی- فلزات- پلیمرها - سرامیک ها - کامپوزیت ها -

انتخاب روش ساخت

جنس ماده مصرفی - هزینه روش - سهولت دسترسی- دقت ابعادی - حجم تولید - محدودیت ها

روش های مناسب برای نمونه سازی

نمونه سازی- ویژگی های نمونه سازی- روش های مناسب برای نمونه سازی - ماشین کاری- فلز کاری

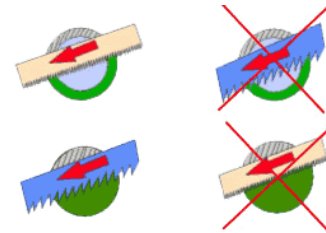
اندازه گیری و اندازه گذاری

خط کش- گونیا- نقاله- کولیس- پرگار- زاویه سنج - سنج

ابزارهاي اندازه کيري
 خط کش- پرگار خارجي- پرگار داخلي - کوليس
 فلز کاري
 ابزار هاي فلز کاري
 ابزار هاي دستي - ابزار هاي ماشيني
 تکنیک هاي فلز کاري
 برش کاري - خم کاري - سوراخ کاري - جوش کاري - لحیم کاري-پرچ کاري
 اشکال مواد مصرفي
 ورق - پروفیل هاي توپر - پروفیل هاي تو خالي
 برش کاري
 تعریف - ابزارهاي برش - اره - قیچی- هوا برش

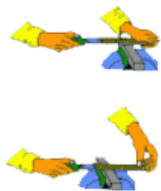
ابزارهاي برش کاري دستي
 گیره- اره دستي - کمان اره- دسته - بدنه - تیغه اره - درشت - متوسط - ریز

انتخاب تیغه مناسب

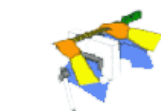


سوهان کاري خشن

سوهان کاري
 تکنیک هاي سوهان کاري



سوهان کاري نرم



پرداخت کاري

خم کاري

تعریف- تکنیک هاي خم کاري - پرس کاري - رول کاري - خم کاري دستي
 سوراخ کاري

آشنائي با مته و ماشين مته- گام هاي سوراخ کاري - علامت گذاري- انتخاب و نصب مته مناسب - تعيين و تنظيم دور -
 ماشين مته- ثابت کردن قطعه کار

اتصالات فلز کاري

اتصالات جذاشدني- پیچ و مهره - اتصالات جذاشدني - پرچ - جوش - لحیم
 حديد کاري

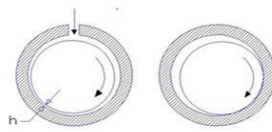
اجزاي مکانيکي و مکانيزمهاي ربات

یاتاقان

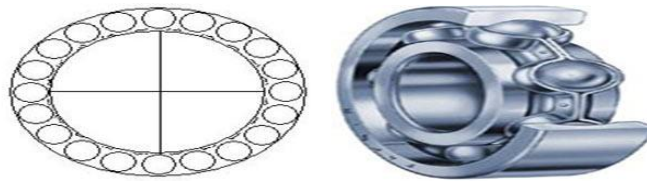
یاتاقان هاي لغزشي- ياتاقان هاي غلتشي: - - بلبرینگ - - رولبرینگ - - ياتاقان هاي غلتک مخروطي
 - ياتاقان هاي کف گرد - - گزینش ياتاقان ها از روي کاتالوگ توليد کننده

یاتاقان هاي لغزشي

سطوح تماس در اين ياتاقان ها توسط قشري از سیال يا ماده چرب کننده از یکدیگر جدا مي شوند



یاتاقان های غلتشی



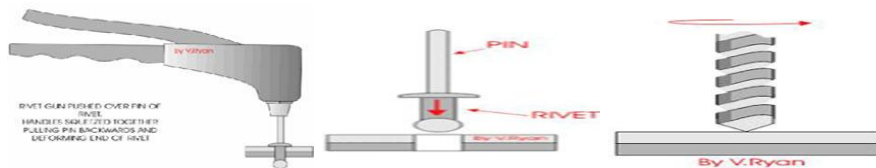
بلبرینگ : معمول ترین نوع یاتاقان های غلتشی که دارای غلتک های کروی هستند .
 رولر برینگ : دارای غلتک های استوانه ای هستند و به علت سطح تماس بیشتر نسبت به بلبرینگ های با همان اندازه بار محوری بیشتری تحمل می کنند .
 تیپر برینگ : دارای غلتک های مخروطی هستند .
 کف گرد : در جایی به کار می روند که بارهای سنگین محوری و ناهمراستایی داشته باشیم .



اتصالات

پیچ و مهره - پرچ - لحیم-جوش - چسب - خار و پین - پیچ و مهره

پرچ



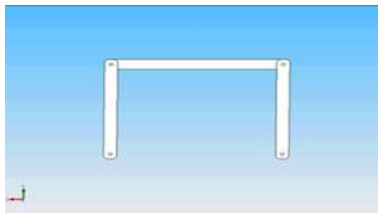
خار

خارهای تخت به عنوان واسطه جهت انتقال گشتاور پیچشی بین میل محور و قطعه ای که روی آن سوار می شود به کار می روند.
 خارهای حلقوی برای تثبیت قطعات روی میل محور به کار می روند.

مکانیزم ها

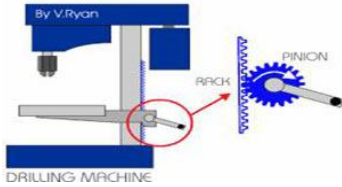
مکانیزم چهار میله ای

این مکانیزم کاربردهای فراوانی دارد و با بررسی آن می توان خواص تعداد زیادی از مکانیزم ها را که معادل این مکانیزم هستند مورد بررسی قرار داد.



مکانیزم چرخ و شانه

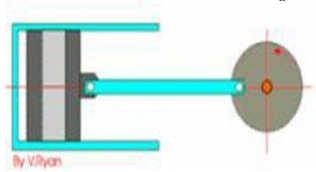
تبدیل حرکت دورانی چرخنده به حرکت مستقیم الخط شانه
اگر α حرکت زاویه ای چرخنده و S حرکت مستقیم الخط شانه و d قطر دایره گام چرخنده باشد:



$$S = (\alpha / 2 \pi) * \pi d = \alpha d / 2$$

مکانیزم لنگ و لغزنده

بارزترین نمونه استفاده از این مکانیزم، موتورهای احتراق داخلی هستند



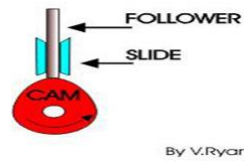
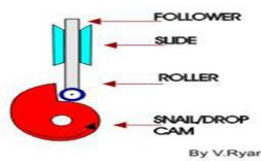
مکانیزم اسکات راسل

مکانیزم بازگشت سریع

در این مکانیزم حرکت چرخشی به یک حرکت رفت و برگشتی متناوب تبدیل می شود. در این حرکت متناوب زمان رفت از زمان بازگشت بیشتر است و حرکت بازگشتی به سرعت صورت می گیرد.

مکانیزم های بادامکی

با یک مکانیزم بادامکی میتوان حرکت چرخشی بادامک را تقریباً به هر نوع حرکت خطی در پیرو تبدیل کرد.



پیچ و مهره

این مکانیزم هم یکی دیگر از مکانیزم های تبدیل حرکت چرخشی به حرکت مستقیم الخط است.

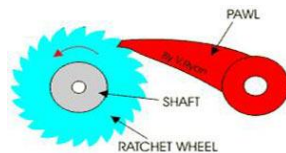
چرخ ضامن دار

چرخ ضامن دار متشکل است از :

1- چرخ

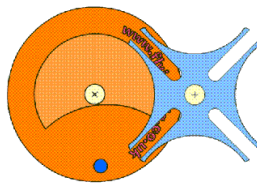
2- گیره

مهمترین ویژگی چرخ ضامن دار این است که گیره فقط اجازه حرکت در یک جهت را به چرخ میدهد



چرخ ژنوا

با استفاده از این مکانیزم می توان یک حرکت چرخشی پیوسته را به حرکت چرخشی منقطع تبدیل کرد.



چرخ دنده و گیربکس
صاف- مارپیچ- مخروطی - حلزونی - گیربکس - گیربکس سیاره ای - انتخاب الکتروموتور مناسب



چرخ دنده ساده
دایره گام (Pitch Circle)
گام دایره ای (p)

$$P = \pi d / N$$

مدول (m)

$$m = d \text{ (mm)} / N$$

نسبت انتقال سرعت



$(W_A / W_B) = (N_A / N_B)$
چرخ دنده های مارپیچ و مخروطی
درگیری تدریجی دنده ها و حرکت نرم
حلزون و چرخ حلزون



محورهای انتقال قدرت متناظرا برهم عمود هستند.
امکان انتقال قدرت با نسبتهای بزرگ میسر است.
حرکت به طور یکطرفه از حلزون به چرخ حلزون منتقل می شود.
نسبت انتقال سرعت حلزون به چرخ حلزون

$$\frac{W_B}{W_A} = \frac{N_A}{N_B} \Rightarrow \frac{W_B}{W_A} = \frac{18}{36} \Rightarrow (W_G / W_W) = (N_W / N_G)$$

$$W_B = 450 \text{ r/min}$$

حلزون معمولا یکراهه است.

$$W_C = W_B = 450 \text{ r/min}$$

گیربکس

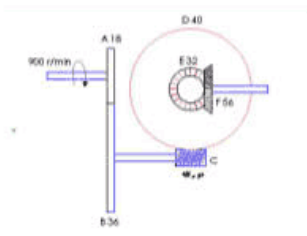
$$\frac{W_D}{W_C} = \frac{N_C}{N_D} \Rightarrow \frac{W_D}{W_C} = \frac{2}{40} \Rightarrow$$

$$W_D = 22.5 \text{ r/min}$$

$$W_E = W_D = 22.5 \text{ r/min}$$

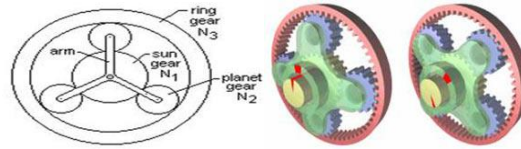
$$\frac{W_F}{W_E} = \frac{32}{56} \Rightarrow W_F = 12.86 \text{ r/min}$$

گیربکس سیاره ای



در این گیربکس ها محور یک یا بیش از یک چرخ دنده نسبت به تکیه گاه حرکت می کند .
معمولا ورودی به چرخ دنده خورشیدی داده می شود ، خروجی از بازو گرفته می شود و چرخ دنده حلقه ای ثابت است .

محور چرخ دنده خورشیدی ثابت و محور چرخ دنده های سیاره ای متحرک است .



گیربکس سیاره ای

با سري کردن چند دستگاه چرخ دنده سیاره ای میتوان به گیربکسي با نسبت تبدیل بالا و اندازه مناسب دست یافت.

انتخاب الکتروموتور مناسب

سرعت، توان و گشتاور خروجي

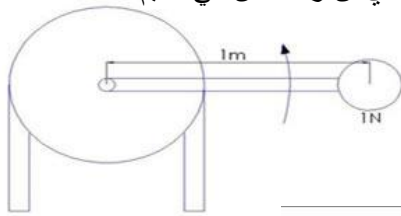
گشتاور معياري است جهت سنجش قدرت گردشي موتور

$$T = Fd \sin a$$

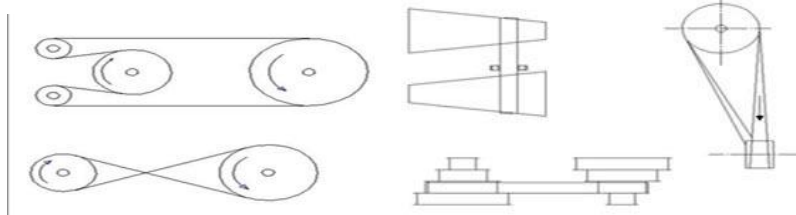
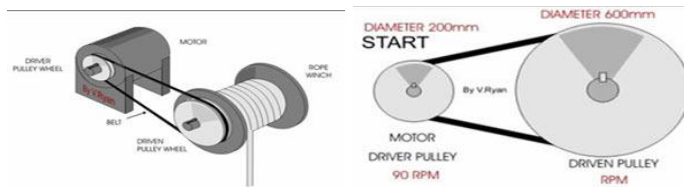
با استفاده از گیربکس گشتاور الکتروموتور را افزایش و سرعت زاویه ای آن را کاهش مي دهيم .

توان

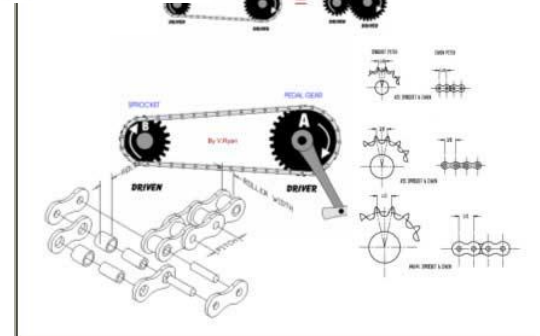
$$P = T\omega$$



پولي و تسمه



زنجر و چرخ زنجر

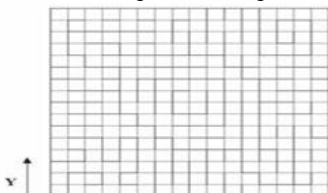


ربات میکرو موس

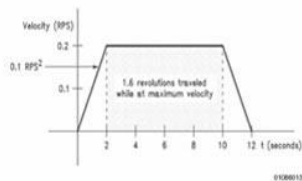
موش هوشمند ربات خودکار و کنترل شده ایست که قادر است مسیری خود را درون يك ماز ناشناخته پیدا کند و به هدف دست یابد.

ماز (Maze) :

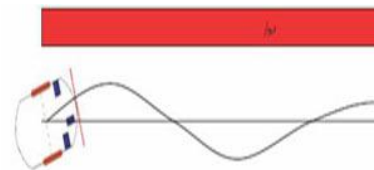
یک جدول 16*16 خانه که شامل تعدادي دیوار می باشد و هدف، رسیدن از یکی از خانه های گوشه جدول به خانه وسط می باشد.



نخستین موش هوشمند در 1977 و در ایالات متحده ساخته شد. هم اکنون بسیاری از دانشگاه‌های معتبر جهان همچون MIT و UL (دانشگاه لندن) و ... دارای تیم هستند.

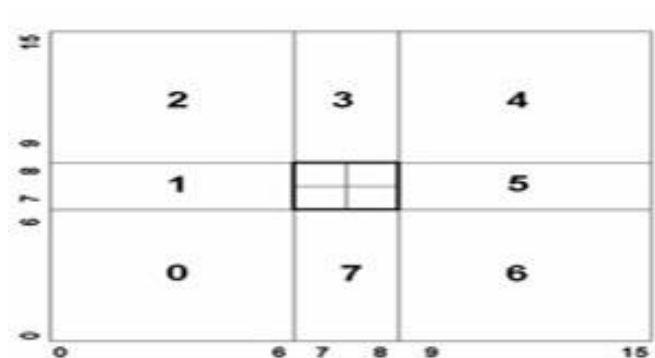


قسمت‌های مختلف یک ربات میکروموس:
مکانیک (شامل شاسی، چرخ‌ها و موتور‌ها)
کنترل (شامل کنترل حرکت و آنالیز داده سنسورها)



سخت افزار

الگوریتم: شامل تصمیم‌گیری و حل ماز



مکانیک

یک ربات میکروموس رباتی است کوچک (تقریباً ۱۰*۱۲ سانتیمتر) که بتواند با شتاب بالا (حدود ۵g) و سرعت بالا (حدود 2.5 m/s) یک ماز را طی کند.
با توجه به طول زیاد مسیر و تعداد زیاد دور زدن‌ها (ماز 16*16)، دقت حرکت ربات باید به حدی باشد که کنترل موقعیت آن در ماز امکان‌پذیر باشد.

ATmega32 یک میکروکنترلر 8 بیت CMOS کم مصرف است که بر پایه ساختار RISC بنا شده است.