

وزارت علوم، تحقیقات و فناوری



موسسه آموزش عالی جهاد دانشگاهی خوزستان

پایان نامه جهت اخذ مدرک کارشناسی

رشته :

مهندسی تکنولوژی ارتباطات و فناوری اطلاعات - مدیریت ICT

عنوان پروژه :

طراحی و ساخت کنترل جهت و دور موتور DC از راه دور با امواج رادیویی

نگارش :

علیرضا چاروسایی

زمستان ۱۳۹۱

وزارت علوم، تحقیقات و فناوری



موسسه آموزش عالی جهاد دانشگاهی خوزستان

صورتجلسه دفاع از پایان نامه جهت اخذ مدرک کارشناسی

جلسه دفاع از پایان نامه (پروژه) کارشناسی آقای/خانم..... دانشجوی  
رشته..... به شماره دانشجویی..... تحت  
عنوان..... با حضور هیات داوران در  
روز..... مورخ...../...../..... برگزار شد و هیات داوران پس از بررسی، پایان نامه مذکور  
را شایسته نمره (به عدد)..... (به حروف)..... با درجه.....  
تشخیص داد.

اعضاء هیات داوران:

نام و نام خانوادگی	سمت	امضاء
	استاد راهنما	
	استاد مشاور	
	استاد داور	
	استاد داور	
	نماینده گروه	
	مدیر گروه	

محل مهر گروه



«تعهد اصالت اثر»

اینجانب **علیرضا چاروسایی** رشته مدیریت فناوری اطلاعات و ارتباطات در مقطع کارشناسی به شماره دانشجویی **تأیید می نمایم** که کلیه مطالب و نتایج مندرج در این پایان نامه حاصل فعالیت پژوهشی و عملی اینجانب بوده و به آثار دیگران که در این نوشته از آن استفاده شده، مطابق مقررات ارجاع گردیده است. این پایان نامه قبلاً برای احراز هیچ مدرکی ارائه نشده است و کلیه حقوق مادی و معنوی این اثر متعلق به موسسه آموزش عالی جهاد دانشگاهی خوزستان می باشد.

بدینوسیله متعهد می شوم در صورتی که خلاف این موضوع ثابت گردد، کلیه عواقب آن منحصراً متوجه اینجانب خواهد بود و دیگران در این خصوص هیچگونه مسئولیتی نخواهند داشت.

وزارت علوم، تحقیقات و فناوری



موسسه آموزش عالی جهاد دانشگاهی خوزستان

«حق انتشار»

تمامی حقوق مادی و معنوی این پایان نامه تحصیلی متعلق به موسسه آموزش عالی جهاد دانشگاهی خوزستان است و هر گونه نقل مطلب تنها با ذکر نام موسسه آموزش عالی جهاد دانشگاهی خوزستان، نام استاد راهنما و نگارنده مجاز است.

دانشجویان در صورتی می توانند نسبت به چاپ مقاله مستخرج از پایان نامه خود اقدام کنند که مقاله مورد تأیید استاد راهنما قرار گرفته باشد. همچنین در صورت ارائه و یا چاپ مقاله مستخرج از این پایان نامه، ذکر نام استاد راهنما و تعلق سازمانی به موسسه آموزش عالی جهاد دانشگاهی خوزستان ضروری است.

عدم رعایت موارد فوق موجب پیگرد قانونی است.

امضا و تاریخ

تقدیم

و بعد از مدتها، پس از پیمودن راههای فراوان که با حضور شیرین اساتید عزیزم، با راهنماییها و دغدگه‌های فراوانشان و شیطنتهای زیبای آن دوران، نگاههای پدر و مادرم، با چشموهای پر از برق شوق، و زیبایی حضور دوستانم در کنارم، که خستگیهای این راه را به امید و روشنی راه تبدیل کرده و امیدوارم بتوانم در آینده‌ی نزدیک جوابگوی این همه محبت آنها را باشم. اکنون، با احترام فراوان برای این همه تلاش این عزیزان برای موفقیت من ، این پایان نامه را به پدر و مادرم، تقدیم میکنم.

سپاسگذاری

پروردگارا

آن دو فرشته ای که از خواسته هایشان گذشتند، سختی ها را به جان خریدند و خود را سپر  
بلای مشکلات و ناملایمات کردند تا من به جایگاهی که اکنون در آن ایستاده ام برسم اما حال  
نه میتوانم موهایشان را که در راه عزت من سفید شد، سیاه کنم و نه برای دستهای پینه بسته  
شان که ثمره تلاش برای افتخار من است، مرهمی دارم . پس توفیقم ده که هر لحظه شکر  
گزارشان باشم و ثانیه های عمرم را در عصای دست بودنشان بگذرانم.

## چکیده

در راستای پیشرفت روز افزون تکنولوژی برای راحتی کاربران پیشرفت های زیادی صورت گرفته است. یکی از این موارد، کنترل وسایل برقی از راه دور می باشد. گاهی اوقات هم فاصله خط فرمان تا دستگاه های خیلی زیاد است و برای افزایش سرعت کار یا کم کردن هزینه های نیاز به ارتباط بی سیم است. دلیل اصلی هر چه که باشد استفاده از این تکنولوژی رو به افزایش است.

در این پروژه یک مدل عملی از کنترل موتور DC با ارتباط بیسیم ارائه شده است. در این پروژه می توان با استفاده از ارتباط رادیویی یک موتور DC را چپ گرد و راست گرد کرده و سرعت آن را کم و زیاد کرد. درایور استفاده شده برای راه اندازی موتور الکتریکی از قسمت کنترلر کاملاً ایزوله است. و هیچ نویزی از موتور الکتریکی نمی تواند بر روی عمل کرد بخش کنترلر تاثیر بگذارد.



## فهرست مطالب

عنوان.....	صفحه
مقدمه.....	۱
فصل اول.....	۲
۱-۱ نحوه راه اندازی پروژه.....	۲
۲-۱ اجزای پروژه.....	۷
۱-۲-۱ لیست قطعات به کار رفته در این پروژه.....	۸
فصل دوم.....	۱۲
۱-۲ آشنایی با رگولاتورها.....	۱۲
۲-۲ شماتیک تغذیه مدار.....	۱۳
فصل سوم.....	۱۹
۱-۳ میکروکنترلر چیست؟.....	۱۹
۱-۳-۱ آشنایی با میکروکنترلر AVR.....	۲۳
۱-۳-۲ آشنایی با ATmega۱۶.....	۲۴
۲-۳ شماتیک قسمت کنترلر.....	۲۸
۱-۲-۳ کانکتور جهت پروگرام کردن میکرو.....	۲۸
۲-۲-۳ کریستال خارجی.....	۳۰
۳-۲-۳ کلید جهت ریست کردن میکروکنترلر.....	۳۱
فصل چهارم.....	۳۲
۱-۴ آشنایی با LCD کاراکتری.....	۳۲
۲-۴ شماتیک مدار مانیتورینگ.....	۳۶

۳۹	فصل پنجم
۴۰	۱-۵ شماتیک درایور موتور
۵۰	فصل ششم
۵۰	۱-۶ آشنایی با ماژول رادیویی ASK
۵۳	۲-۶ آشنایی با تراشه های انکدر و دیکدر
۶۲	۳-۶ شماتیک گیرنده رادیویی
۶۵	فصل هفتم
۶۵	۱-۷ شماتیک برد فرستنده
۷۱	فصل هشتم
۷۱	۱-۸ استاندارد طراحی مدار چاپی
۷۴	۲-۸ روش های فیبر مدار چاپی
۷۴	۱-۲-۸ روش مازیک
۷۵	۲-۲-۸ روش لتراست
۷۶	۳-۲-۸ روش لامینت
۷۷	۴-۲-۸ روش اسپری پزتیو
۸۰	۵-۲-۸ روش تهیه مدار چاپی با اتو
۸۴	۳-۸ روش تهیه اسید مدار چاپی
۸۶	۴-۸ شکل PCB پروژه
۸۷	۵-۸ برد پروژه
۸۸	فصل نهم
۸۸	۱-۹ آشنایی با کامپایلرهای میکروکنترلر AVR

۹۰	..... Bascom AVR محیط کامپایلر ۲-۹
۹۲	..... فصل دهم
۹۲	..... ۱-۱۰ برنامه میکروکنترلر
۹۵	..... ۱-۱-۱۰ توضیحات برنامه میکروکنترلر
۱۰۵	..... منابع

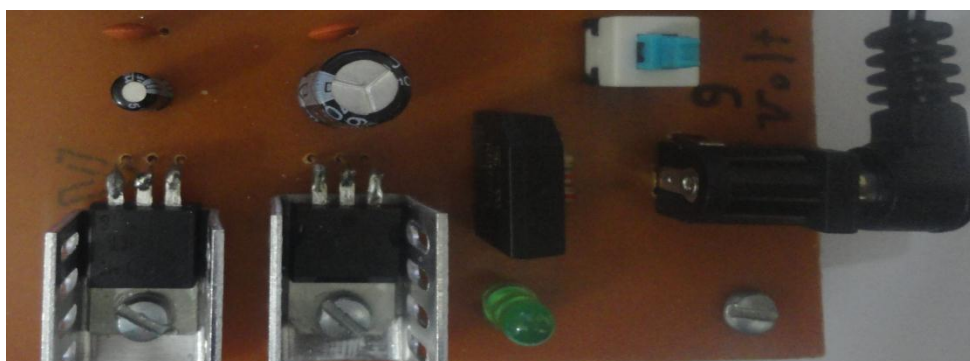
با استفاده از این پروژه چپ گرد ، راست گرد و سرعت گردش موتور DC از راه دور قابل کنترل است. توجه داشته باشید که در این پروژه از فیدبک استفاده نشده است و در صورتی که موتور تحت بار قرار بگیرد ، دور آن کم خواهد شد و پروژه توانایی تشخیص این مسئله را ندارد. در این پروژه از یک موتور الکتریکی کوچک ۱۲ ولت DC استفاده شده است. این پروژه قادر به کنترل هر موتور DC تا ۲۴ ولت و مصرف ۲ آمپر می باشد. برای کنترل این پروژه از میکروکنترلر AVR به نام ATmega۱۶ استفاده شده است. جهت مانیتورینگ چپ گرد و راست گرد موتور و هم چنین سرعت موتور از LCD کاراکتری ۱۶\*۲ استفاده شده است. برای ارتباط رادیویی بین ریموت و برد اصلی از امواج رادیویی ASK استفاده شده است.

## فصل اول

### آشنایی با پروژه

#### ۱-۱ نحوه راه اندازی پروژه

برای استفاده از این پروژه، نیاز به ۲ عدد آداپتور ۱۲ ولت DC یا AC دارید. یکی از آداپتورها برای تغذیه مدار الکترونیکی پروژه و دیگری برای تغذیه موتور الکتریکی می باشد. برای تغذیه مدار پروژ یک آداپتور ۱۲ ولت تا  $300\text{mA}$  نیز کافی است. این آداپتور باید به جک سایه رنگی که در شکل مشخص شده است متصل شود:



شکل ۱-۱ تغذیه مدار

برای تغذیه موتور الکتریکی می توان از یک آداپتور ۱۲ ولت  $1\text{A}$  استفاده کرد. انتخاب این آداپتور باید متناسب با موتور الکتریکی مورد استفاده باشد.  
توجه:

چنانچه از یک موتور ۶ ولت استفاده میکنید، حتما از یک آداپتور ۶ ولت استفاده نمایید.

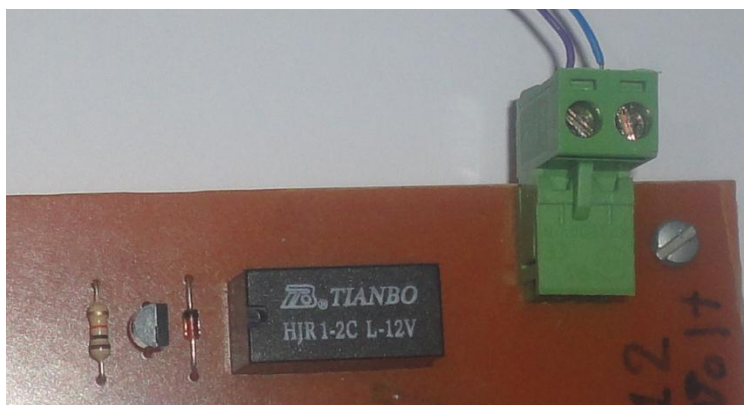
محل اتصال این آداپتور در شکل زیر مشخص شده است:



شکل ۲-۱ تغذیه موتور الکتریکی

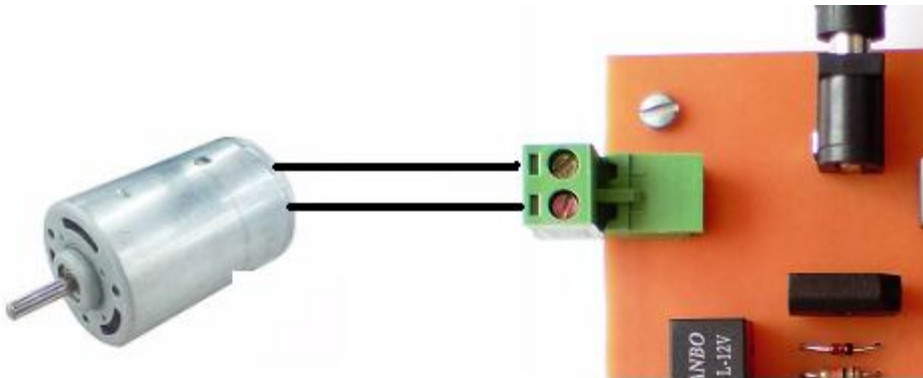
موتور الکتریکی استفاده شده در این پروژه از نوع DC می باشد. ولتاژ کاری این موتور باید متناسب با آداپتوری باشد که برای راه اندازی استفاده می شود.

توجه: به هیچ وجه از یک آداپتور ولتاژ بالا برای راه اندازی یک موتور با ولتاژ کمتر استفاده نشود. محل اتصال موتور الکتریکی، سبز رنگی است که در شکل زیر مشخص شده است:



شکل ۳-۱ ترمینال فونیکس

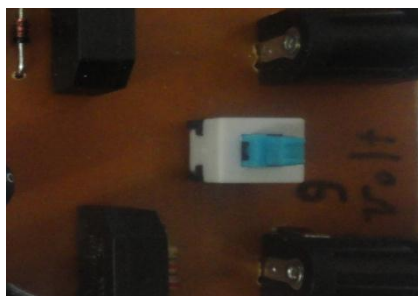
دو سر موتور را با استفاده از سیم به ترمینال متصل کنید.



شکل ۴-۱ ترمینال فونیکس

سیم را در شیار ترمینال قرار داده و با استفاده از پیچ ها، آن را محکم کنید.

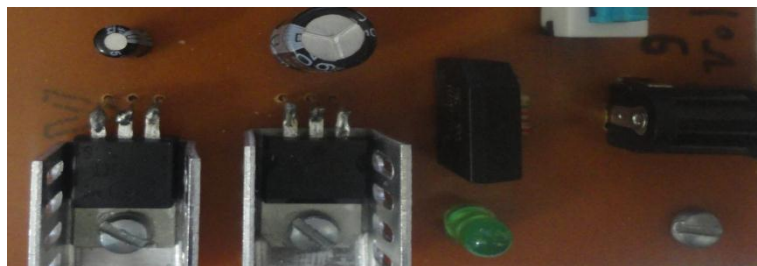
جهت روشن شدن برد باید کلید کوچک ON/OFF را فشار دهید:



شکل ۵-۱ کلید ON/OFF

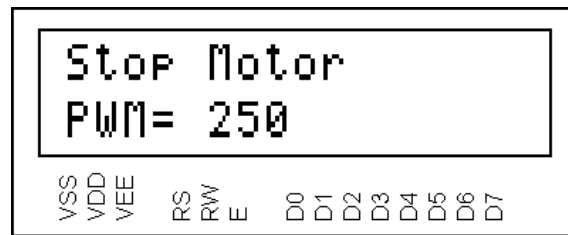
در صورت اتصال درست آداپتور ۱۲ ولتی که جهت تغذیه مدار میباشد LED سبز رنگ کنار

کلید ، روشن خواهد شد



شکل ۶-۱ اتصال درست آداپتور

پس از روشن شدن پروژه متن زیر بر روی LCD نمایش داده می شود.



شکل ۷-۱ LCD

در صورتی که متن بالا را مشاهده نکردید باید با یک عدد پیچ گوشتی کوچک ، پتانسیومتر سمت چپ LCD را بچرخانید.



شکل ۸-۱ پتانسیومتر LCD

این پتانسیومتر به منظور تنظیم وضوح تصویر LCD استفاده می شود.

پتانسیومتر سمت راست برای تنظیم شدت نور LCD است.

تنظیم این پتانسیومتر برای بیشترین شدت نور ، باعث افزایش مصرف پروژه می شود.

جهت کنترل موتور باید از کلید های روی برد فرستنده استفاده شود.



راهنمای استفاده از این کلید ها از قرار زیر است:



شکل ۱-۹ کلید های فرستنده

با فشردن کلید **Left\_Key** موتور چپ گرد خواهد چرخید.

با فشردن کلید **Right\_Key** موتور راست گرد خواهد چرخید.

با فشردن کلید **Stop** موتور از حرکت خواهد ایستاد.

با فشردن کلید **UP\_Key** مقدار **PWM** افزایش می یابد. با افزایش **PWM** دور موتور

بیشتر می شود.

با فشردن کلید **Down\_Key** مقدار **PWM** کاهش می یابد. با کاهش **PWM** دور موتور

کم می شود.

نکته: افزایش مقدار **PWM** باعث افزایش ولتاژ دو سر موتور میشود. و به همین دلیل دور

موتور بیشتر میشود، جهت کم کردن دور موتور باید با کم کردن **PWM** ولتاژ دوسر موتور را

کم کرد.

در تمامی مدت، جهت گردش موتور و مقدار **PWM** بر روی **LCD** قابل مشاهده است:

```
Rotate Left  
PWM= 125
```

مقدار PWM تنظیم شده در حافظه EEPROM میکروکنترلر ذخیره می شود، اطلاعات ذخیره

شده در EEPROM با قطع تغذیه میکروکنترلر از بین نمیرود.



از دست زدن به زیر برد فرستنده جدا خود داری شود.



از تماس زیر برد اصلی با سطوح فلزی جدا خود داری شود.



از قرار دادن برد بر روی فرش یا هر سطح پرزدار خود داری کنید. زیرا الکتریسیه

ساکن روی سطوح پشمی ممکن است به برد آسیب برساند.

## ۲-۱ اجزای پروژه

سخت افزار این پروژه از قسمت های زیر تشکیل شده است:

۱. برد الکترونیکی گیرنده

• تغذیه

• کنترلر

• مانیتورینگ

• درایور ( راه انداز موتور )

• گیرنده رادیویی

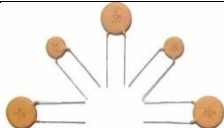
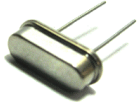


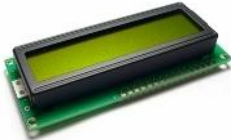








۲. برد الکترونیکی فرستنده



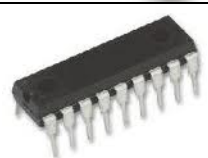
نرم افزار:

- برنامه میکروکنترلر

### ۱-۲-۱ لیست کامل قطعات به کار رفته در این پروژه

تعداد	تصویر	نام قطعه
۱		برد مسی خام
۲		جک آداپتوری مادگی (رو بردی)
۲		پل دیود ۲ آمپر / KBP210 / شانه ای
۱		تک سوئیچ کوچک / ۶ پایه on off /
۲		LED زرد ۵ میلی مات
۱		LED سبز ۵ میل مات
۱		رگولاتور ۷۸۰۹
۱		رگولاتور ۷۸۰۵
۳		هیتسینک ۱.۵*۲
۱		خازن الکتrolیت ۴۷۰Uf/۱۶V
۱		خازن الکتrolیت ۱uf/۵۰V
۲		خازن عدسی ۱۰۰nF

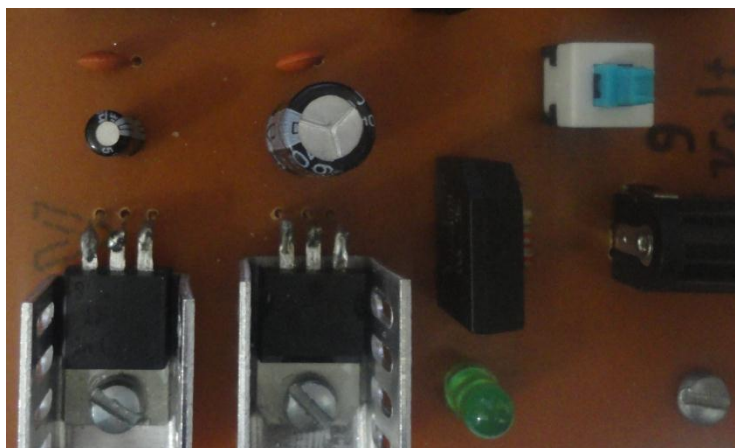
۲		خازن عدسی ۲۷pF
۱		کریستال ۱۱.۰۵۹۲MHz
۶		تک سوئیچ ۲ پایه / ۱.۵
۲		پتانسیومتر خوابیده ۲K اهم
۱		LCD کاراکتری ۲*۱۶
۱		پین هدر مادگی ۱۶ پین
۱		پین هدر نری ۱۶ پین
۱		پین هدر مادگی ۶ پین
۱		رله مخابراتی دو کنتاكت ۱۲ ولت
۲		۱N۴۱۴۸
۱		BC۳۲۷
۱		TLP۵۲۱
۱		ترمینال فونیکس ۲ پایه / رایت

۳		مقاومت ۱۰K اهم
۶		مقاومت ۴.۷K اهم
۲		مقاومت ۲.۲K اهم
۱		مقاومت ۲۲۰K اهم
۱		مقاومت ۱M اهم
۲		مقاومت ۳۳۰ اهم
۱		ATmega۱۶
۱		سوکت زیر IC ، ۴۰ پین
۱		PT۲۲۶۲
۱		PT۲۲۷۲-L۴
۲		سوکت زیر IC ۱۸PIN
۱		ماژول گیرنده رادیویی RWS-۳۱۵
۱		ماژول فرستنده رادیویی TWS-۳۱۵

۲		آنتن تلسکوپی
۱		باتری ریموتی ۱۲V / ۲۳A
۲		آداپتور ۱۲V
۱۱		پیچ ۱CM
۴		پیچ ۲CM
۲۳		مهره

## فصل دوم

### تغذیه مدار



شکل ۱-۲ تغذیه مدار

#### ۱-۲ آشنایی با رگولاتورها

رگولاتورها در یک دسته بندی کلی به ۳ بخش زیر تقسیم میشوند.

۱. رگولاتورهای ولتاژ خروجی ثابت مثبت: خروجی آنها یک عدد ثابت و غیر قابل تغییر + می باشد که نام گذاری آنها هم به صورت ۷۸XX یا L۷۸XX یا M۷۸XX می باشد. ۲. رقم سمت راست که به صورت XX نشان داده شده نشان دهنده ی ولتاژ خروجی است. مثلاً ولتاژ خروجی رگولاتور ۷۸۰۵، ۵ ولت می باشد L یا M هم نشان دهنده ی حداکثر جریان دهی آن است ( L تا ۱ آمپر و M تا ۱.۵ آمپر).

۲. رگولاتورهای ولتاژ خروجی ثابت منفی: خروجی آنها یک عدد ثابت منفی و غیر قابل تغییر می باشد که نامگذاری آنها به صورت ۷۹XX می باشد.

۳. رگولاتورهای ولتاژ خروجی متغیر: به وسیله ی این رگولاتورها می توان ولتاژ خروجی را کنترل کرد. معروف ترین و پر کاربردترین نوع خروجی + آنها LM۳۱۷ و LM۱۳۸ و LM۳۳۸ و خروجی - آنها LM۳۳۷ می باشد. این قطعه برای ره اندازی نیاز به یک مدار جانبی مختصر دارد.

این رگولاتورها ۳ پایه دارند. مثبت +، خروجی، زمین یا -

قطب - منبع تغذیه را زمین نیز می گوئیم (Gnd)

در رگولاتورهای سری ۷۸XX ولتاژ ورودی باید حداقل ۲.۳ ولت بیشتر از خروجی آنها باشد. حداقل ولتاژ ورودی و همچنین ولتاژ خروجی آنها در جدول زیر آمده است:

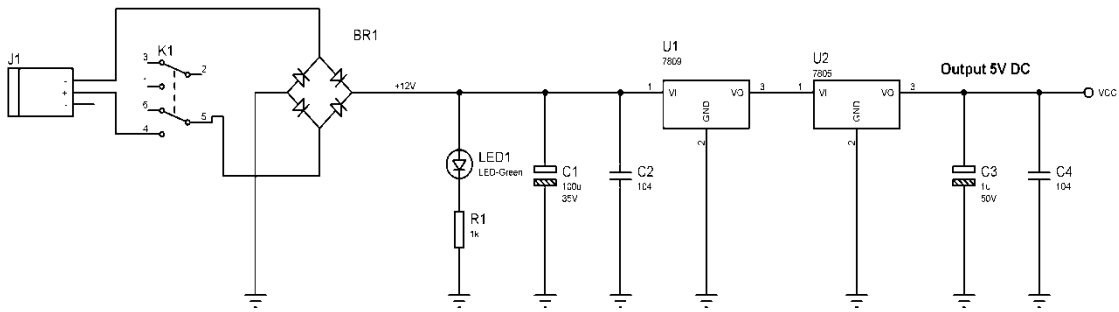
ولتاژ خروجی	حداقل ولتاژ ورودی	شماره مدل
۵	۷.۳	۷۸۰۵
۸	۱۰.۵	۷۸۰۸
۹	۱۱.۵	۷۸۰۹
۱۲	۱۴.۶	۷۸۱۲
۱۸	۲۱	۷۸۱۸
۲۴	۲۷.۱	۷۸۲۴

## ۲-۲ شماتیک تغذیه مدار

تغذیه داخلی مدار +۵ ولت DC است برای این منظور ما نیاز به یک مدار داریم تا ولتاژ ۱۲ ولت را به ۵ ولت یکسو شده تبدیل کند.

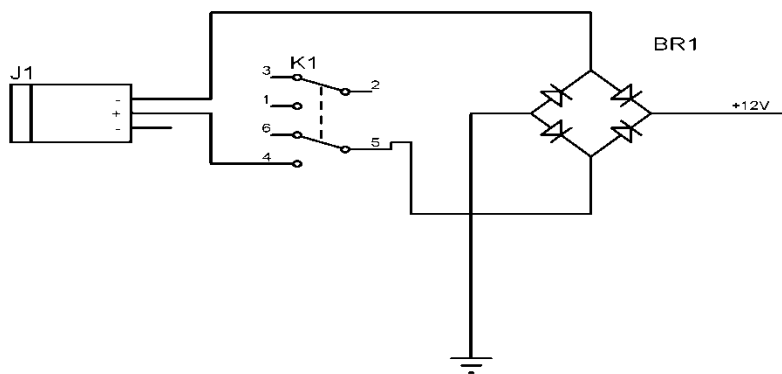


مدار این قسمت از قرار زیر است:



شکل ۲-۲ شماتیک تغذیه مدار

قسمت اول مدار تغذیه:



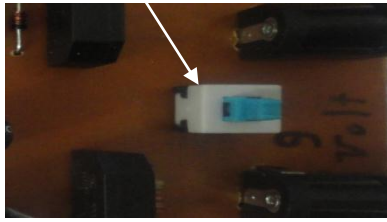
شکل ۲-۳ شماتیک تغذیه مدار

J1 : جک مشکی رنگ جهت اتصال آداپتور. برای این پروژه از یک آداپتور ۱۲ ولت DC استفاده شده است



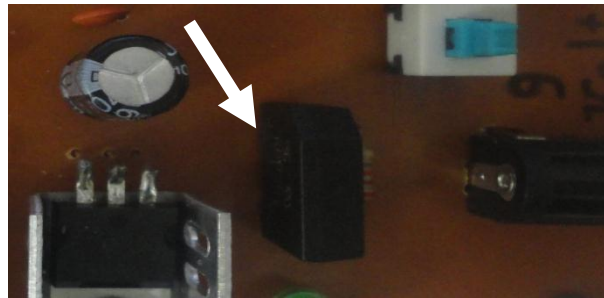
شکل ۲-۴ اتصال آداپتور

K1: کلید کوچک ON/OFF جهت روشن و خاموش کردن مدار.



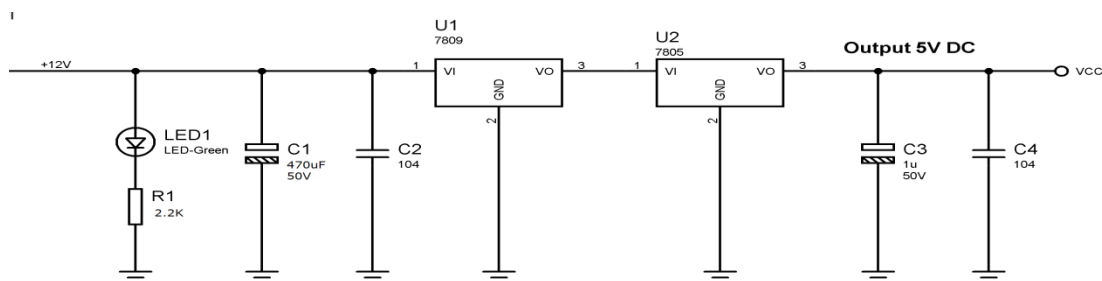
شکل ۲-۵ کلید ON/OFF

BR1: پل دیودی ۲ آمپر جهت یکسو کردن ولتاژ ورودی. این پل دیودی مدار را از اتصال غلط پلاریته تغذیه جلوگیری میکند. با وجود این پل دیودی، می توان از تغذیه AC نیز برای این پروژه استفاده کرد.



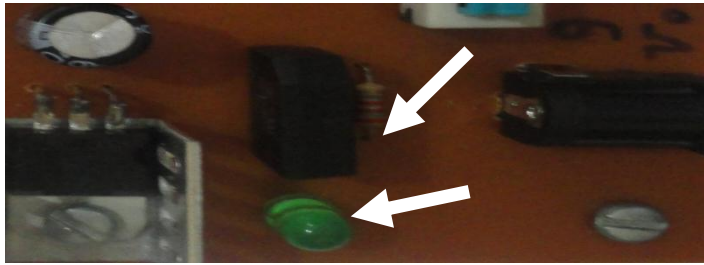
شکل ۲-۶ پل دیودی

قسمت دوم مدار تغذیه:



شکل ۲-۷ شماتیک تغذیه مدار

R1 و LED1 : یک عدد مقاومت و LED سبز رنگ جهت نمایش وصل شدن ولتاژ به مدار.



شکل ۸-۲ جهت نمایش وصل شدن ولتاژ

C1 : خازن الکترولیت برای حذف نویز از مدار . رنج این خازن به تجربه ۴۷۰uf / ۵۰v

انتخاب شد.



شکل ۹-۲ حذف نویز از مدار

C2 : خازن عدسی کوچک ۱۰۰nf جهت حذف نویز های ریز مدار.

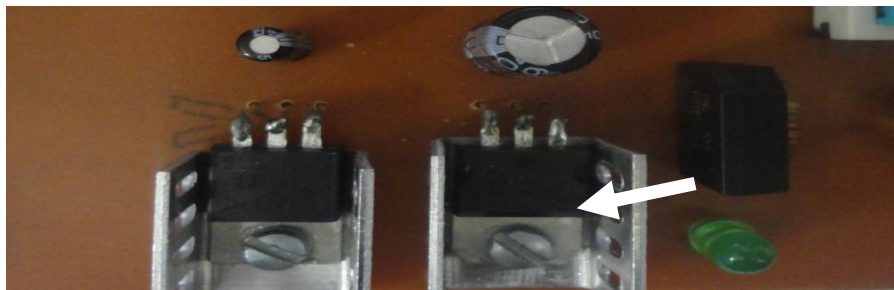


شکل ۱۰-۲ حذف نویز از مدار

۷۸۰۹ : این آی سی یک رگلاتور می باشد، و وظیفه آن در مدار تغذیه بسیار مهم است. این

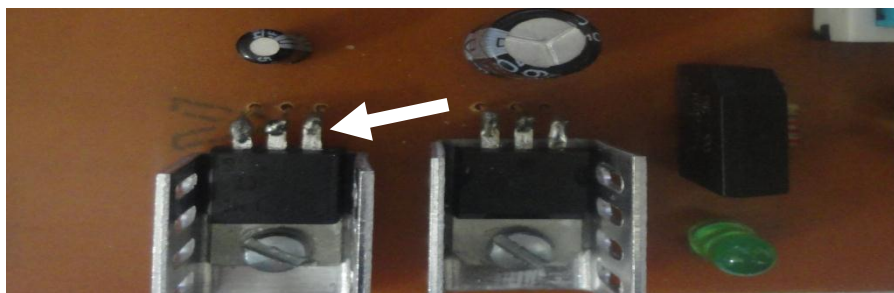
رگلاتور ولتاژ ورودی را به ۹ ولت تبدیل میکند، طوری که تغییرات ولتاژ ورودی در ولتاژ

خروجی هیچ تاثیری ندارد. برای اینکه این رگلاتور درست کار کند، ولتاژ ورودی باید بیشتر از ۱۰ ولت باشد. برای خنک کردن این رگلاتور از یک هد سینگ استفاده شده است.



شکل ۱۱-۲ رگلاتور ولتاژ

۷۸۰۵: این آی سی یک رگلاتور می باشد، و وظیفه آن در مدار تغذیه ، بسیار مهم است. این رگلاتور ولتاژ ورودی را به ۵ ولت تبدیل میکند، طوری که تغییرات ولتاژ ورودی در ولتاژ خروجی هیچ تاثیری ندارد. برای اینکه این رگلاتور درست کار کند، ولتاژ ورودی باید بیشتر از ۷ ولت باشد. برای خنک کردن این رگلاتور از یک هد سینگ استفاده شده است.

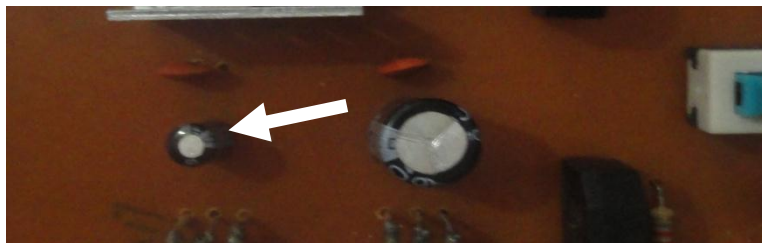


شکل ۱۲-۲ رگلاتور ولتاژ

توجه کنید که اگر رگلاتور ۷۸۰۹ در مدار نباشد در کوتاه مدت هیچ مشکلی پیش نیاید. ولی از آنجایی که تبدیل ۱۰ ولت به ۵ ولت گرمای زیادی در ۷۸۰۵ تولید میکند، این گرما باعث سوختن آن می شود.

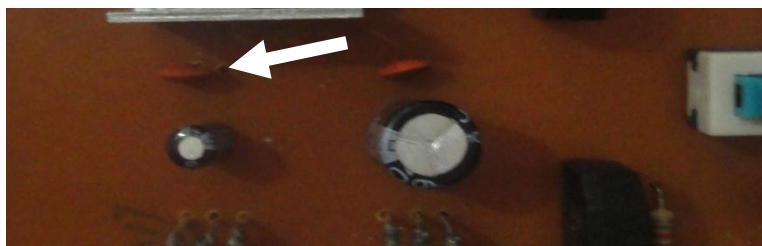
C<sup>3</sup>: خازن الکترولیت برای حذف نویز از مدار - رنج این خازن به تجربه  $1\mu\text{f}/50\text{V}$  انتخاب

شد.



شکل ۲-۱۳ حذف نویز از مدار

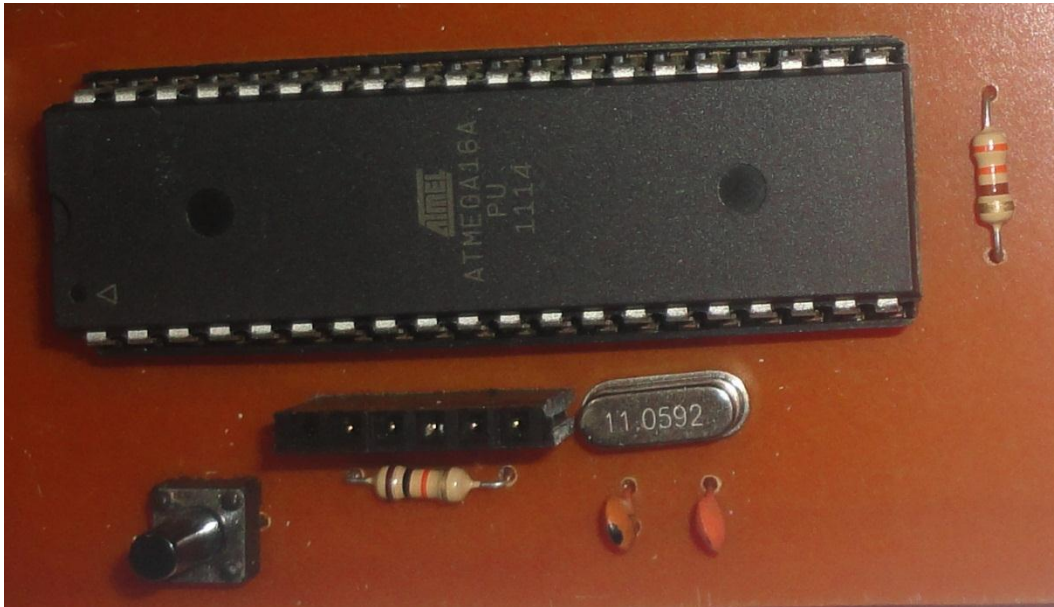
C<sup>4</sup>: خازن عدسی کوچک  $100\text{nf}$  جهت حذف نویز های ریز مدار.



شکل ۲-۱۴ حذف نویز از مدار

## فصل سوم

### کنترلر



شکل ۱-۳ کنترلر

در این فصل به توضیحات قسمت کنترلر میپردازد.

#### ۱-۳ میکروکنترلر چیست؟

۱. معرفی میکروکنترلرها:

به آی سی هایی که قابل برنامه ریزی می باشد و عملکرد آنها از قبل تعیین شده میکروکنترلر گویند میکرو کنترل ها دارای ورودی - خروجی و قدرت پردازش میباشد.

۲. بخشهای مختلف میکروکنترلر:

میکروکنترلرها از بخشهای زیر تشکیل شده اند:

Cpu واحد پردازش

Alu واحد محاسبات

I/O ورودی ها و خروجی ها

Ram حافظه اصلی میکرو

Rom حافظه ای که برنامه روی آن ذخیره می گردد

Timer برای کنترل زمان ها

و ...

۳. خانواده های میکروکنترلر

خانواده AVR - Pic - ۸۰۵۱

۴. یک میکروکنترلر چگونه برنامه ریزی میشود.

میکرو کنترلر ها دارای کامپایلرهای خاصی می باشد که با زبان های , Assembly basic

c و Basic می توان برای آنها برنامه نوشت سپس برنامه نوشته شده را توسط دستگاهی به

نام programmer که در این دستگاه ای سی قرار می گیرد و توسط یک کابل بهیگی از در

گاه های کامپیوتر وصل می شود برنامه نوشته شده روی آی سی انتقال پیدا میکند و در Rom

ذخیره می شود.

۵. با میکرو کنترلر چه کارهایی می توان انجام داد.

این آی سی ها حکم یک کامپیوتر در ابعاد کوچک و قدرت کمتر را دارند بیشتر این آی سی ها برای کنترل و تصمیم گیری استفاده می شود چون طبق الگوریتم برنامه ی آن عمل می کند این آی سی ها برای کنترل ربات ها تا استفاده در کارخانه صنعتی کار برد دارد.

۶. امکانات میکرو کنترلرها:

امکانات میکرو کنترلرها یکسان نیست و هر کدام امکانات خاصی را دارا می باشند و در قیمت های مختلف عرضه می شود.

۷. شروع کار با میکرو کنترلر:

برای شروع کار با میکرو کنترلر بهتر است که یک زبان برنامه نویسی مثل **C** یا **basic** را بیاموزید سپس یک برد **programmer** تهیه کرده و برنامه خود را روی میکرو ارسال کنید سپس مدار خود را روی برد برد بسته و نتیجه را مشاهده کنید. چنان چه در مدارهای الکترو نیکی تجربه ندارید بهتر است از برنامه های آموزش استفاده کنید.

۸. مقایسه خانواده های مختلف میکرو و کنترلرها:

خانواده ۸۰۵۱:

این خانواده از میکرو کنترلر ها جزو اولین نوع میکرو کنترلر ها می بود که رایج شده و جزو پیشکسوتان مطرح میشود . معروف ترین کامپایلر برای این نوع میکرو **keil** یا **franklin** می باشد میکرو های این خانواده به نوسان ساز نیازمند هستند و درمقابل خانواده **AVR pic** از امکانات کمتری برخوردار می باشد معروف ترین آی سی ها این خانواده **S۵۱ ۸۹** یا **C۵۱ ۸۹** می باشد.



خانواده AVR :

این خانواده از میکرو کنترلرها تمامی امکانات ۸۰۵۱ را دارا می باشد و امکاناتی چون ADC (مبدل آنالوگ به دیجیتال) - نوسان ساز داخلی و قدرت و سرعت بیشتر (EEPROM حافظه) از جمله مزایای این خانواده می باشد مهم ترین آی سی این خانواده Tiny و Mega است.

خانواده PIC :

این خانواده از نظر امکانات مانند AVR میباشد و در کل صنعتی تر است.

۹. مزایای میکرو کنترلر نسبت به مدار های منطقی :

یک میکرو کنترلر را می توان طوری برنامه ریزی کرد که کار چندین گیت منطقی را انجام دهد.

تعداد آی سی هایی که در مدار به کار میرود به حداقل میرسد.

به راحتی می توان برنامه میکرو کنترلر را تغییر داد و تا هزاران بار میتوان روی میکرو برنامه های جدید نوشت و یا پاک کرد

به راحتی میتوان از روی یک مدار منطقی کپی کرد و مشابه آن را ساخت ولی در صورتی که از میکرو کنترلر استفاده شود و برنامه میکرو را قفل کرد به هیچ عنوان نمی توان از آن کپی گرفت.

### ۳-۱-۱ آشنایی با میکروکنترلر AVR

میکروکنترلرهای AVR توسط شرکت ATMEEL طراحی و ساخته شده‌اند. اولین قطعات از میکروکنترلرهای AVR در سال ۱۹۹۳ روانه بازار شد و خیلی زود جای خود را در قلب طراحان مدارات میکروکنترلی باز کرد. نخستین قطعات که در این خانواده معرفی شدند، میکروکنترلرهای AVR در سری AT۹۰Sxxxx بودند. ولی از آنجایی که این قطعات نسبت به سایر میکروکنترلرهای AVR که بعداً در سری MEGA شدند امکانات کمتری دارند، به ندرت از آن‌ها استفاده می‌شود.

شرکت ATMEEL همزمان با ارائه میکروکنترلرهای AVR در سری MEGA اقدام به طراحی و تولید میکروکنترلرهای AVR در سری Tiny کرده است. این قطعات در بسته بندی های کوچکتری نسبت به هم نمونه های قبلی و با امکانات فوق العاده در جریان های مصرفی کم ارائه شده اند و زمینه را برای طراحی مداراتی با توان مصرفی فوقالعاده کم و کارایی بسیار بالا فراهم کرده اند.

میکروکنترلرهای AVR دارای درگاه داده ۸ بیتی و از نوع CMOS و با ساختار RISC هستند و در ساخت آنها معماری نوع Harvard بهکار برده شده است.

در این نوع معماری از باس های سه گانه مجزا ( آدرس - داده - کنترل ) برای حافظه برنامه استفاده می شود. کاربرد ساختار RISC باعث می شود که این قطعات دارای خصوصیات منحصر به فردی باشند. از آن جمله می توان به سرعت بالا ، سازگاری با کامپایلر های زبان های سطح بالا چون C و امکانات فراوان اشاره کرد.

ساختار RISC برای اولین بار در سال ۱۹۷۰ برای معماری پردازشگرها معرفی شد. پیش از این معماری CISC متداول تر بوده است. بحث در مورد این مورد از عهده ی این مقاله خارج است.

### ۳-۱-۲ آشنایی با ATmega۱۶

مشخصات این AVR از قرار زیر است.

- از معماری AVR RISC استفاده می کند.

۱. کارایی بالا و توان مصرفی کم
۲. دارای ۱۳۱ دستورالعمل با کارایی بالا که اکثرا تنها در یک کلاک سیکل اجرا می شوند.

۳. ۳۲\*۸ رجیستر کاربردی

۴. سرعتی تا ۱۶MIPS در فرکانس ۱۶ MHZ

- حافظه ، برنامه و داده غیر فرار

۱. ۱۶K بایت حافظه FLASH داخلی قابل برنامه ریزی

پایداری حافظه FLASH : قابلیت ۱۰۰۰۰ بار نوشتن و پاک کردن

(WRITH/ERASE)

۲. ۲K بایت حافظه داخلی SRAM

۳. ۱۰۲۴ بایت حافظه EEPROM داخلی قابل برنامه ریزی

پایداری حافظه EEPROM : قابلیت ۱۰۰۰۰۰ بار نوشتن و پاک کردن

۴. قفل برنامه FLASH و حفاظت داده EEPROM

• قابلیت ارتباط JTAG (IEEE Std.)

۱. برنامه ریزی برنامه FLASH ، EEPROM ، FUSE BITS ، LOCK BIT از طریق

ارتباط JTAG

• خصوصیات جانبی

۱. دو تایمر- کانتر ( TIMER / COUNTER ) ۸ بیتی با PRESCALER مجزا و

دارای مد COMPARE .

۲. یک تایمر - کانتر ۱۶ بیتی با PRESCALER مجزا و دارای مدهای COMPARE

CAPTURE ،

۳. ۴ کانال PWM

۴. ۸ کانال مبدل آنالوگ به دیجیتال ۱۰ بیتی

دارای دو کانال تفاضلی با کنترل گین ۱X ، ۱۰X ، ۲۰۰X

۵. یک مقایسه کننده آنالوگ داخلی

۶. دارای RTC ( REAL – TIMER CLOCK ) با اسیلاتور مجزا.

۷. WATCHDOG قابل برنامه ریزی با اسیلاتور داخلی

۸. ارتباط سریال SPI به صورت MASTER یا SLAVE

۹. قابلیت ارتباط با پروتکل سریال دو سیمه ( TWO – WIRE )

۱۰. USART سریال قابل برنامه ریزی

- خصوصیات ویژه میکروکنترلر

۱. POWER – ON RESET CIRCUIT

۲. BROWN – OUT DETECTION قابل برنامه ریزی

۳. دارای ۶ حالت SLEEP

ADC NOISE REDUCTION

EXTENDED STANDBY

STANDBY

POWER – SAVE

IDEL

POWER – DOWN

۴. منابع وقفه ( INTERRUPT ) داخلی و خارجی.

۵. دارای اسیلاتور RC داخلی کالیبره شده.

۶. عملکرد کاملاً ثابت

۷. توان مصرفی پایین و سرعت بالا توسط تکنولوژی CMOS

- ولتاژهای عملیاتی ( کاری )

۱. ۲.۷۷ تا ۵.۵۷ برای ( ATMEGA ۳۲L )

۲. ۴.۵۷ تا ۵.۵۷ برای ( ATMEGA ۳۲ )

- فرکانسهای کاری

۱. ۰MHz تا ۸MHz برای ( ATMEGA ۳۲L )

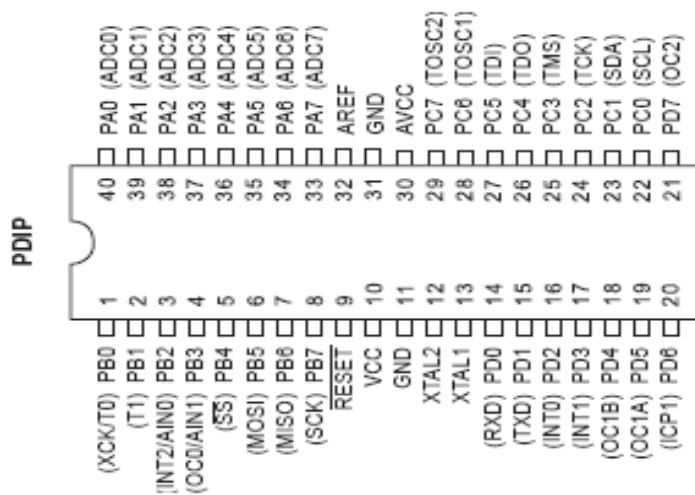
۲. ۰MHz تا ۱۶MHz برای (ATMEGA۱۶)

- خطوط I/O و انواع بسته بندی

۱. ۳۲ خط ورودی/خروجی (I/O) قابل برنامه ریزی

۲. ۴۰ پایه PDIP ، ۴۴ پایه TQFP و ۴۴ پایه MLF

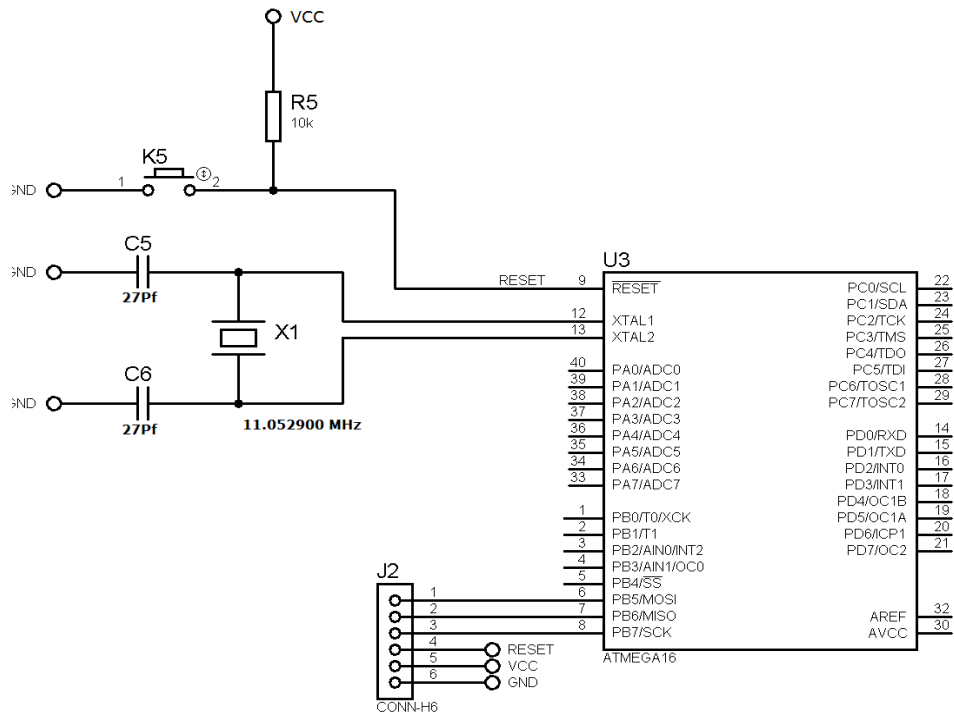
- ترکیب پایه ها



شکل ۲-۳ دیتاشیت atmega۱۶

### ۲-۳ شماتیک قسمت کنترلر

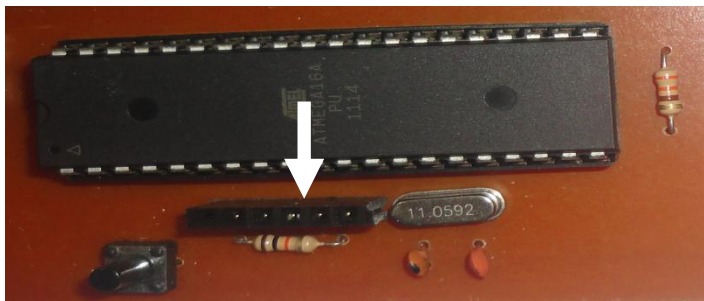
مدار این قسمت از قرار زیر است:



شکل ۳-۳ شماتیک کنترلر

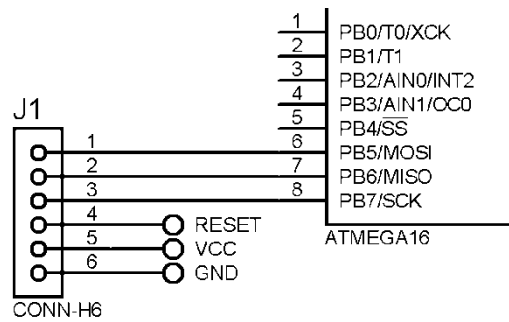
میکرو کنترلر به کار رفته در این پروژه ATmega16 از خانواده AVR است.

### ۱-۲-۳ کانکتور جهت پروگرام کردن میکرو



شکل ۳-۴ کانکتور

برای راحتی کار، یک کانکتور نیز برای SPI Programmer در نظر گرفته ایم تا هر زمان نیاز شد بتوانیم میکرو را بر روی پروژه پروگرام کنیم. و دیگر نیازی به جدا کردن میکروکنترلر از روی برد نباشد.



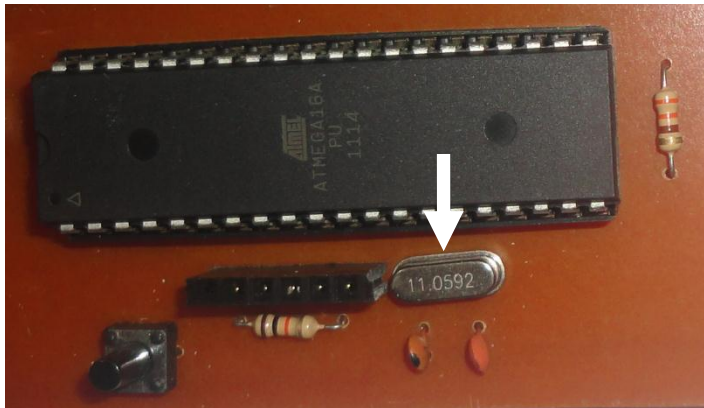
شکل ۳-۵ شماتیک کانکتور

پایه های که در پروگرامر SPI استفاده می شود از قرار زیر است:

۱. MOSI
۲. MISO
۳. SCK
۴. Reset
۵. VCC
۶. GND

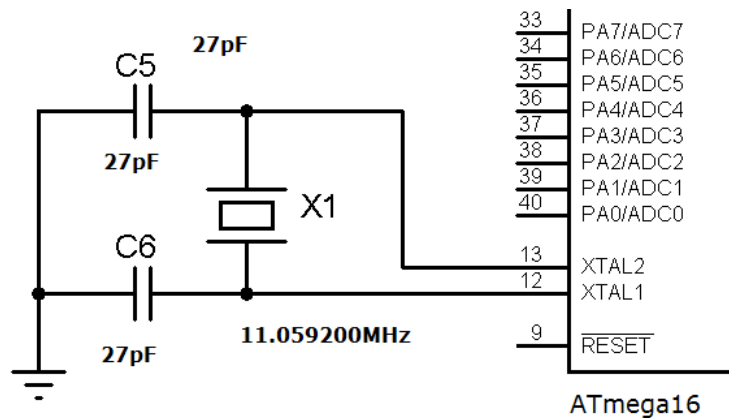


### ۲-۲-۳ کریستال خارجی



شکل ۳-۶ کریستال

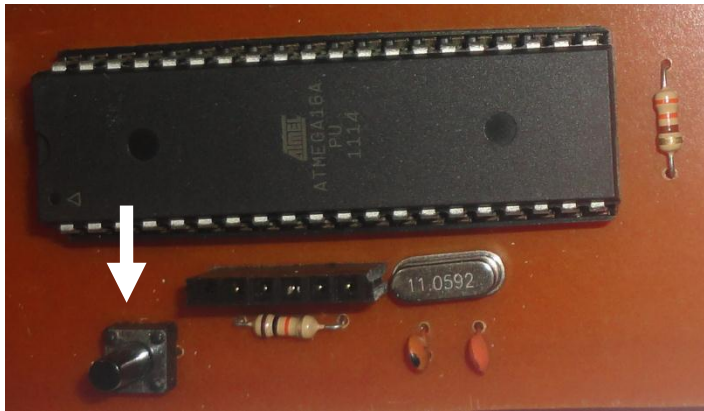
برای دقت بیشتر میکرو و کاهش نویز به جای کریستال داخلی از کریستال خارجی استفاده شده که مطابق شکل زیر به میکرو وصل شده است.



شکل ۳-۷ شماتیک کریستال

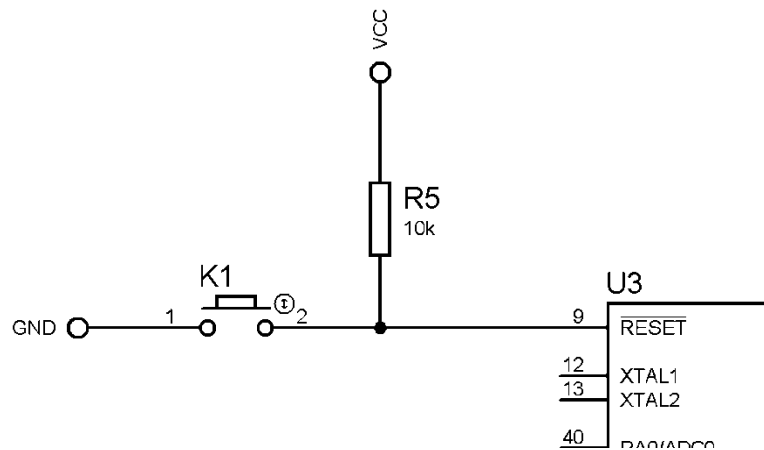
کریستال انتخاب شده برای این پروژه  $11.059200\text{MHz}$  است. و برای جلوگیری از نویز از خازن های  $27\text{pf}$  استفاده شد.

### ۳-۲-۳ کلید جهت ریست کردن میکروکنترلر



شکل ۳-۸ کلید ریست

برای اینکه بتوانیم میکروکنترلر را ریست کنیم نیاز داریم تا یک عدد میکرو سویچ را مطابق شکل زیر به میکرو متصل کنیم،



شکل ۳-۹ شماتیک کلید ریست

اگر پایه ریست رها بماند ممکن است به علت نویز محیط خود به خود ریست کند، برای جلوگیری از این امر آن را با یک مقاومت  $10K$  اهم به مثبت وصل می کنیم.

## فصل چهارم

### مانیتورینگ

قسمت مانیتورینگ پروژه شامل یک LCD کاراکتری می باشد که در شکل زیر مشخص است.



شکل ۴-۱ مانیتورینگ

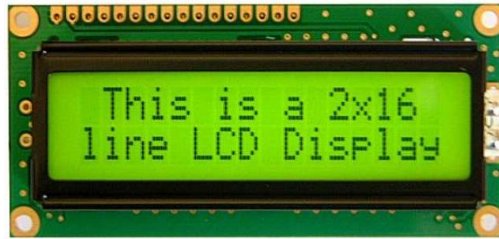
جهت مانیتورینگ مقادیر می توان از Seven-segment و یا LCD استفاده کرد.

Seven-Segment ها ارزان هستند ولی راه اندازی آنها کمی پیچیده می باشد و در ضمن قابلیت نمایش کاراکتر را ندارد. LCD ها با میکروکنترلر به راحتی پیکره بندی شده و الارغم نمایش عدد می تواند کاراکترهای انگلیسی را نیز نمایش دهد. برای مانیتورینگ پروژه از یک LCD کاراکتری ۱۶\*۲ استفاده کردیم.

#### ۴-۱ آشنایی با LCD کاراکتری

LCD ها ابزاری برای نمایش اطلاعاتی هستند که شامل حروف و اعداد و همچنین برخی کاراکترهای گرافیکی می شود. بطور معمول در تجربیات اولیه در نمایش اطلاعات دیجیتال از

نمایشگر های هفت قسمتی (seven segment) استفاده می شود که این نمایشگرها فقط ارقام (۰ تا ۹) و بعضی حروف مثل A b C را بصورت نه چندان زیبا نمایش می دهند. اما با بکار گیری LCD اطلاعات را بصورت زیبا و کاملتر می توان نمایش داد. البته استفاده از LCD برای مدارات ساده توصیه نمی شود و عموماً آنرا همراه با میکروکنترلر یا CPU ها بکار می برند. چیزی که از آن بعنوان LCD یاد می شود در واقع یک صفحه نمایشگر LCD مانند صفحه ماشین حساب است که همراه با آی سی کنترلر و مدارهای جانبی اش و عموماً با لامپ پشت صفحه در یک بسته پیش ساخته عرضه می شود. همانطور که گفته شد LCD دارای یک کنترلر است که با فرستادن اطلاعات به آن این اطلاعات را در صفحه ای که عموماً به چند سطر و ستون تقسیم شده نمایش می دهد. مثلاً برای نمایش حرف "M" کفایت کد اسکی این حرف را طبق یک پروتکل ساده به LCD ارسال کنیم. همچنین می توان دستوراتی از قبیل پاک کردن صفحه نمایش، جابجایی مکان نما، خاموش روشن کردن مکان نما و غیره را نیز به LCD ارسال کرد. LCD ها از طریق مقدار اطلاعاتی که میتوانند در صفحه نمایش بدهند انتخاب و خریداری می شوند. انواع معمول آن عبارتند از ۱۶، ۲۰، ۳۲ و ۴۰ کاراکتر در هر خط در ۱ یا ۲ یا ۴ سطر. مثلاً ۲ در ۱۶ یعنی صفحه دارای دو خط و هر خط ۱۶ کاراکتر است. همچنین LCD موردنظر میتواند همراه با لامپ پشت صفحه (Back light) یا بدون آن انتخاب شود. LCD ها کاراکترها را در ماتریس های ۵ pixel x۷ نمایش می دهند. در تصویر زیر یک نمونه ۲ در ۱۶ مشاهده می شود:



شکل ۴-۲ LCD ۱۶\*۲

تقریباً همه LCD ها دارای ۱۶ پایه هستند که ۸ خط آن مربوط به فرستادن یا خواندن داده ها یا دستورالعمل ها می باشد. پایه های دیگر خطوط کنترل و ولتاژهای تغذیه می باشند. لیست کامل خط ها بقرار زیر است:

شماره و نام خط و عملکرد

۱- VSS : زمین

۲- VCC : ولتاژ ۵ ولت برای کنترلر

۳- Vee : ولتاژ تنظیم درخشندگی (contrast)

۴- RS : انتخابگر ثبات دستور / داده

۵- RW : انتخابگر خواندن / نوشتن

۶- Enable : فعال کننده

۷-۱۴ Bus : خط گذرگاه داد یا دستور

۱۵- ولتاژ ۵ ولت برای لامپ پشت صفحه

۱۶- زمین : برای لامپ پشت صفحه

**Vee** : برای تنظیم درخشندگی کاراکترها بکار می رود که باید ولتاژی بین صفر و ۵ ولت به این پایه اعمال نمود. برای بیشترین درخشندگی این پایه را به زمین متصل کنید.

انتخابگر ثبات داده / دستور مشخص می کند که چه چیزی به LCD فرستاده می شود. اگر این خط صفر باشد کنترلر LCD بایت موجود روی خطوط ۷ تا ۱۴ را بعنوان یک دستور تلقی کرده و اگر این پایه یک باشد اطلاعات را بعنوان یک کد اسکی که باید کاراکتر معادل آنرا نمایش دهد در نظر می گیرد.

انتخابگر خواندن / نوشتن جهت اطلاعات را نشان می دهد. اگر این پایه صفر باشد اطلاعات به LCD ارسال می شود و اگر یک باشد عمل خواندن از LCD صورت می گیرد.

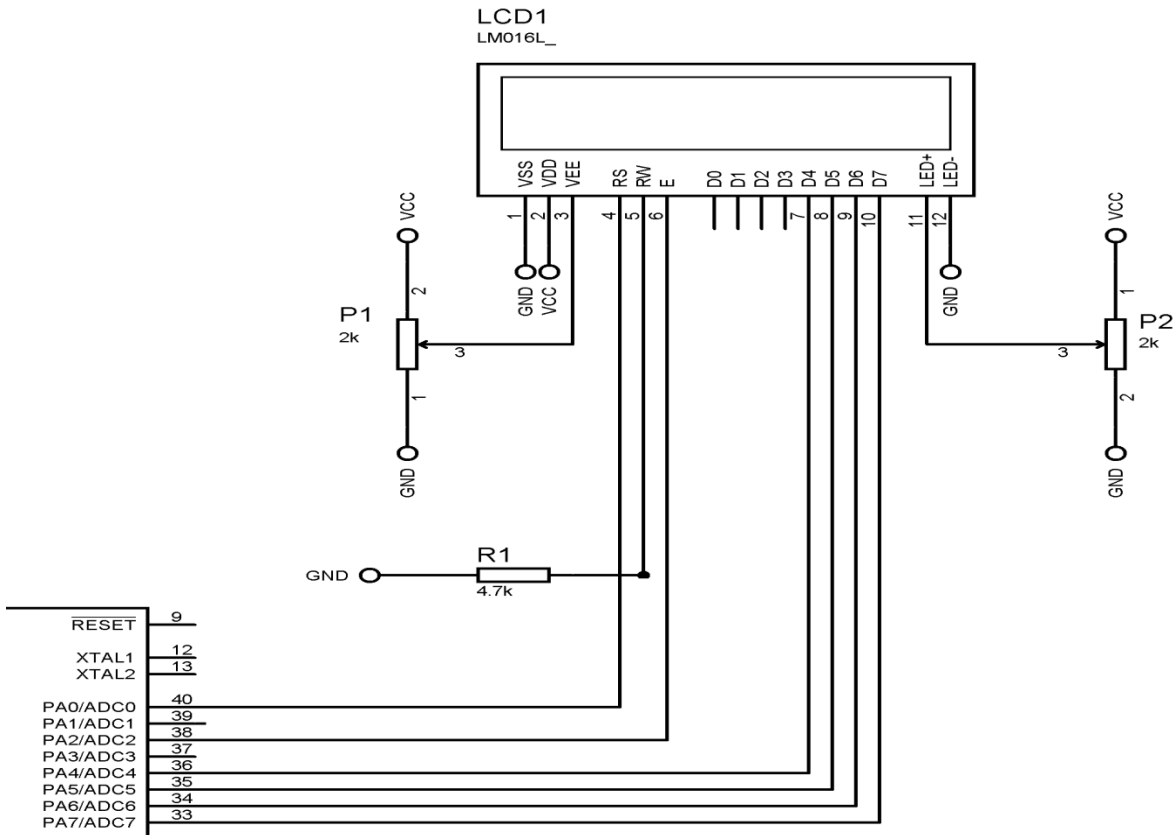
فعال کننده: برای هر دستور یا داده ای که به LCD میفرستیم یا میخوانیم از آن بخوانیم باید یک پالس پائین رونده (یعنی تغییر از سطح یک به صفر) را به این پایه اعمال کنیم تا دستور یا داده بوسیله کنترلر LCD پردازش شود. در خطوط ۷ تا ۱۴ خط ۷ کم ارزشترین بیت (LSB) و خط ۱۴ پر ارزش ترین بیت (MSB) می باشد. در صورت تمایل به روشن کردن لامپ پشت صفحه ولتاژ ۵ ولت را به پایه ۱۵ اعمال و پایه ۱۶ را به زمین متصل می کنیم. برای آزمایش می توان LCD را به پورت چاپگر متصل و اطلاعاتی را به آن ارسال نمود. در این حالت بطور معمول خطوط داده پورت به خطوط ۷ تا ۱۴ و سه خط کنترلی به پایه های ۴ تا ۶ اتصال داده می شود توجه داشته باشید که ولتاژ تغذیه و لامپ پشت صفحه LCD توسط منبع خارجی تامین می شود.

روش فرستادن یک کاراکتر:

خط خواندن نوشتن را صفر کنید تا نوشتن انتخاب شود. خط داده / دستور را یک کنید تا داده انتخاب شود. کد اسکی کاراکتر مورد نظر را روی خطوط D<sub>0</sub> تا D<sub>7</sub> قرار دهید. خط انتخاب را ابتدا یک و سپس صفر کنید. حداقل ۴۵۰ نانو ثانیه باید این خط را صفر نگه دارید تا داده پردازش شود. بعد از آن حالت خط تاثیری نخواهد داشت.

#### ۲-۴ شماتیک مدار مانیتورینگ

شماتیک این قسمت از قرار زیر است:



شکل ۳-۴ شماتیک مانیتورینگ

VSS و VDD : پایه ۱ و ۲ از LCD برای تغذیه خود LCD است.

پایه ۱۱ و ۱۲ از LCD برای روشنایی LCD می باشد. ما برای کنترل شدت نور LCD از یک پتانسیومتر ۲K اهم استفاده کردیم. این پتانسیومتر در مسیر پین ۱۱ از LCD قرار گرفته است.

پایه ۳ از LCD برای تنظیم وضوح تصویر است که برای آن از یک پتانسیومتر ۲K اهم استفاده کردیم. البته ناگفته نماند که این مقادیر در نظر گرفته شده برای پتانسیومترها تجربی می باشد. می توان از هر پتانسیومتر بالای ۲K اهم برای این منظور استفاده کرد.

پایه های RS و E و D۴ و D۵ و D۶ و D۷ را مطابق شکل بالا به میکرو وصل می کنیم .

پایه RW : این پایه بیت کنترل LCD است. برای نوشتن بر روی LCD این پایه باید ۰ شود و برای خواندن از LCD این پایه باید ۱ شود. از آنجایی که ما قصد نوشتن بر روی LCD را داریم این پایه را سخت افزاری به زمین وصل می کنیم تا ۰ شود.

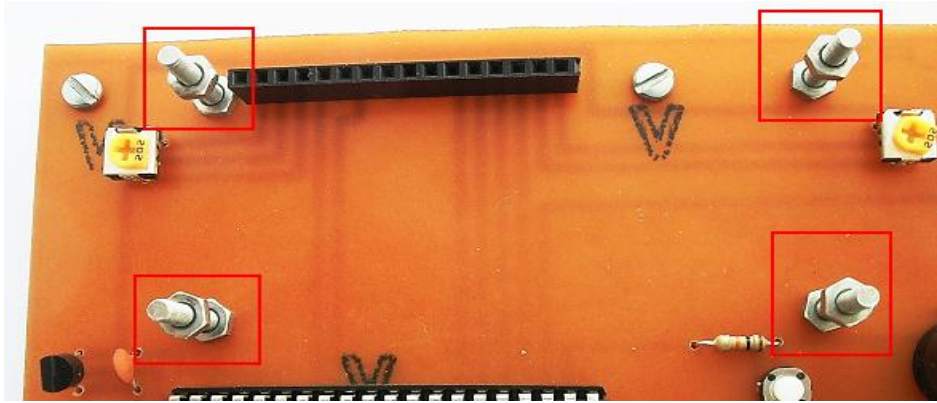
توجه داشت باشید که هیچ وقت پین های LCD را مستقیم بر روی برد لحیم نمکنیم بلکه برای این کار از پین هدر استفاده میکنیم.



شکل ۴-۵ پین هدر

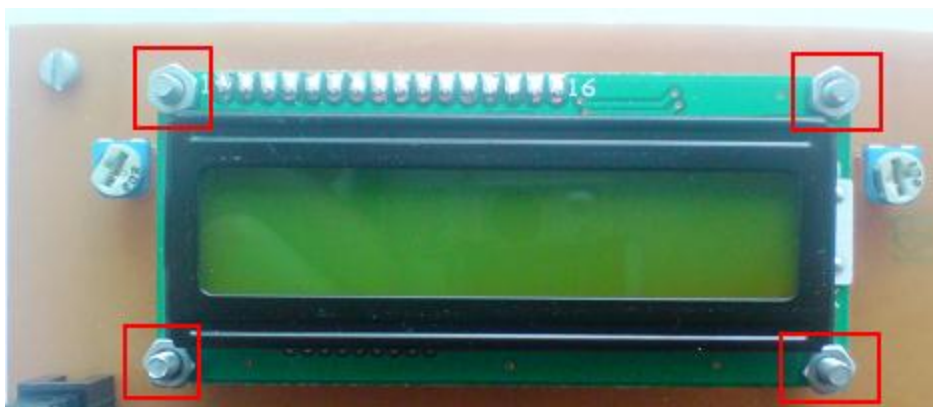


و برای محکم کردن LCD بر روی برد می توان چهار طرف LCD را با استفاده از پیچ و مهره به برد بست. این کار از جدا شدن LCD جلوگیری میکند.



شکل ۴-۶ پیچ و مهره

این کار از جدا شدن LCD جلوگیری میکند.



شکل ۴-۷ پیچ و مهره

## فصل پنجم

### درایور ( راه انداز موتور )

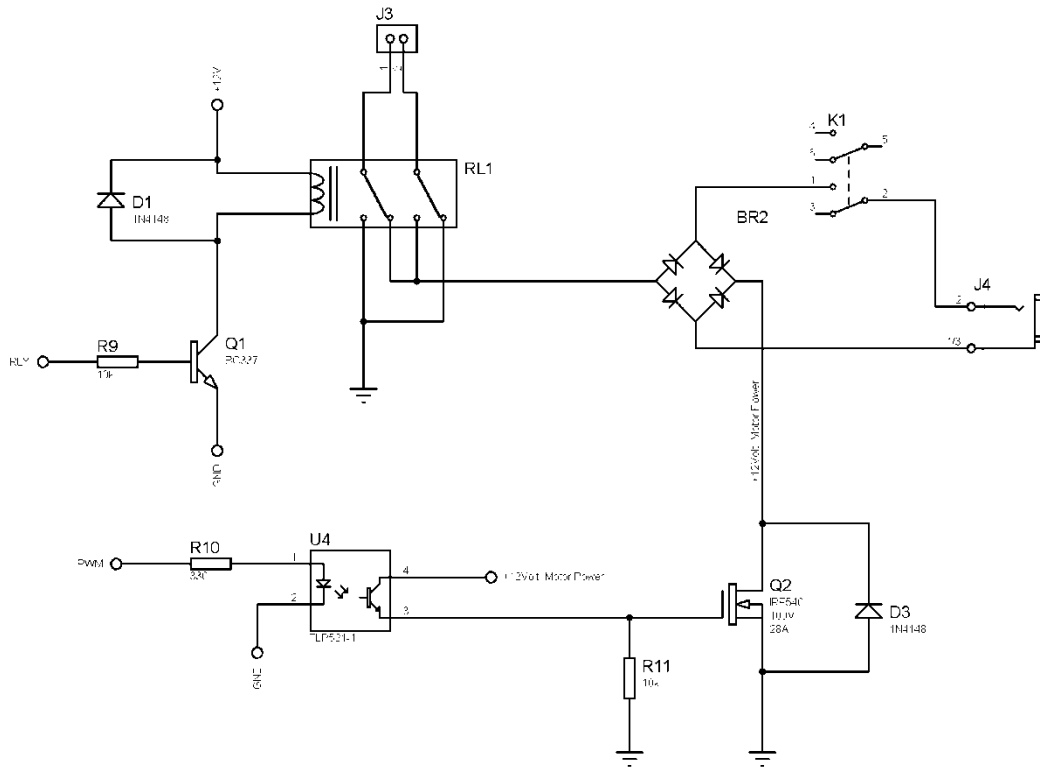


شکل ۵-۱ درایور

از آنجایی که جریان کشیده شده توسط موتور خیلی زیاد است ، میکروکنترلر مستقیماً نمی تواند برق موتور ها را تأمین کند. برای راه اندازی موتور های الکتریکی نیاز به مدار واسطی داریم تا بدون دخالت مستقیم میکروکنترلر، موتور را کنترل کنیم. فرایندهایی چون چپ گرد و راست گرد و همچنین کم و زیاد کردن سرعت موتور می باشد.

## ۱-۵ شماتیک درایور موتور

شماتیک درایور موتور از قرار زیر است:



شکل ۲-۵ شماتیک درایور موتور

J۴: جک ورودی آداپتور

آداپتوری که به این جک متصل می شود برای تغذیه موتور DC استفاده می شود.

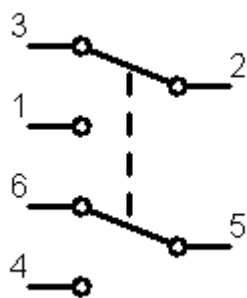
این جک در شکل زیر مشخص شده است:



شکل ۳-۵ جک آداپتور

### K1 : کلید ON/OFF کوچک

از این کلید جهت قطع و وصل کردن تغذیه موتور DC استفاده می شود. این کلید همان کلیدی است که تغذیه مدار برد الکترونیکی را قطع و وصل می کرد. این کلید دو کنتاكت است. و هم زمان برای قطع و وصل کردن دو منبع تغذیه قابل استفاده می باشد.



شکل ۴-۵ کلید ON/OFF

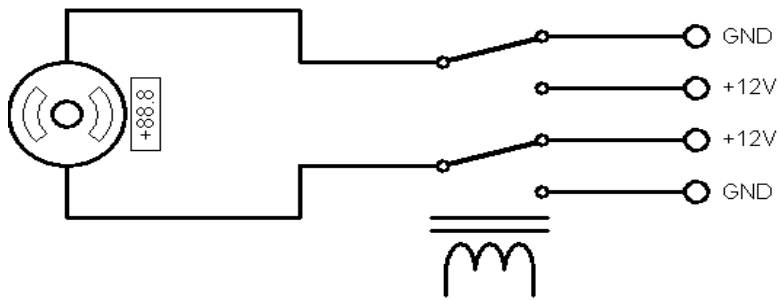
### RLY : رله دو کنتاكت ۱۲ ولت مخابراتی



شکل ۵-۵ رله دو کنتاكت

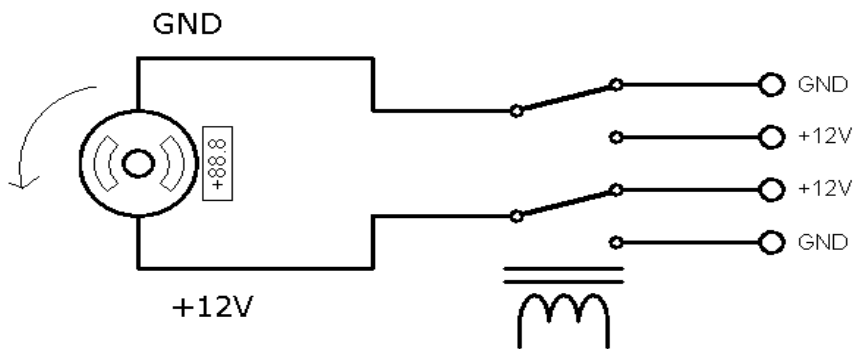
از این رله جهت چپ گرد و راست گرد موتور استفاده شده است.

نحوه قرار گرفتن این رله در مدار از قرار زیر است:



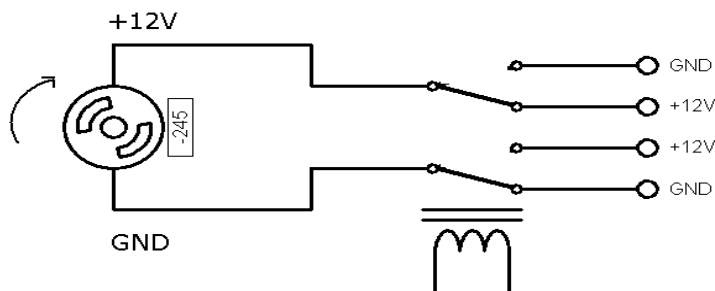
شکل ۵-۶ رله در مدار

در حالت عادی که بوبین رله وصل نیست، موتور در خلاف جهت ساعتگرد خواهد چرخید:



شکل ۵-۷ بوبین رله

با فعال شدن بوبین رله تیغه های رله تغییر وضعیت داده و باعث تغییر پلاریته دو سر موتور می شوند. و در نتیجه چرخش موتور معکوس شده و در جهت ساعت گرد خواهد چرخید.

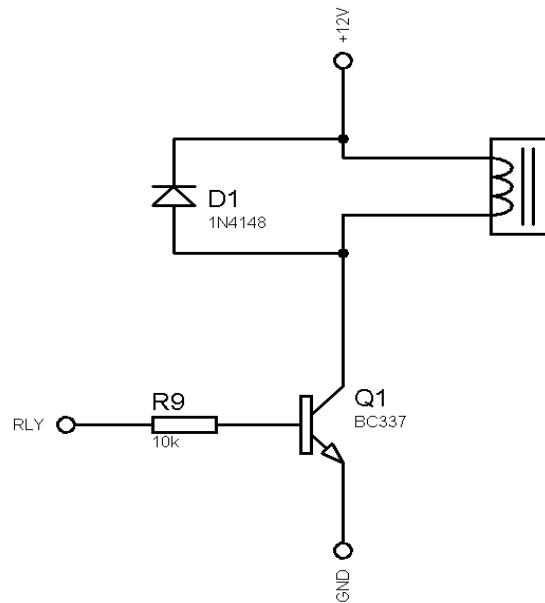


شکل ۵-۸ تغییر وضعیت رله

برای فعال کردن بوبین رله مجاز به اتصال مستقیم آن به میکروکنترلر نیستیم.

چرا که جریان استفاده شده توسط سیم پیچ های بوبین رله، بیش از تحمل پایه های میکروکنترلر است.

برای این کار از شماتیک زیر استفاده شد:



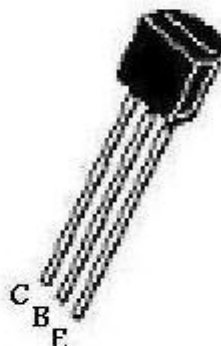
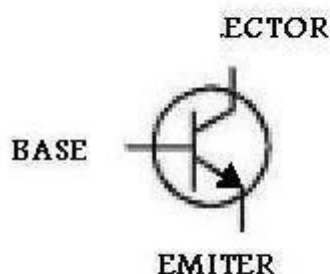
شکل ۵-۹ شماتیک فعال کردن بوبین رله

Q1: ترانزیستور NPN به شماره BC337

از آنجایی که جریان مصرفی بوبین رله، از میزان جریان دهی یک پین میکرو بیشتر است، پس باید بوبین رله را با یک ترانزیستور درایو کنیم و بیس ترانزیستور به یک پایه دلخواه از میکرو متصل شود.

مشخصات این ترانزیستور از قرار زیر است:

# BC337



## Description:

Silicon NPN Low Power High Frequency Bipolar Transistor (625mW,

NF-Tr, 50V, 0,8A, 0,625W, 100MHz

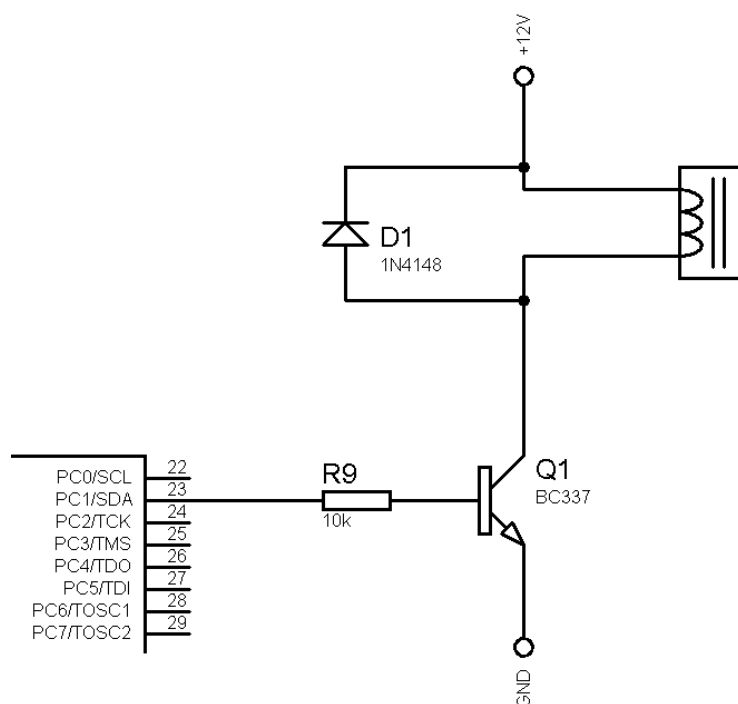
## Replacements:

BC 637, BC 639, 2SD667, ++

R9 : مقاومت ۱۰ کیلو اهم

اگر بیس ترانزیستور مستقیماً تحریک شود به علت جریان زیاد خواهد سوخت. برای تحریک کردن بیس ترانزیستور به جریانی در حدود چند میکرو آمپر نیاز است. و جریان بیشتر از این به ترانزیستور آسیب می رساند. برای محدود کردن جریان بیس ترانزیستور از مقاومت ۱۰K اهم استفاده شده است.

بیس ترانزیستور جهت کنترل چپ گرد و راست گرد موتور باید به یکی از پایه میکروکنترلر متصل شود. این پایه به دلخواه انتخاب می شود. فقط در زمان برنامه نویسی این پایه باید به عنوان خروجی تعریف شود.



شکل ۵-۱۰ فعال کردن بوبین رله

D1 : دیود شیشه ای به نام 1N4148

دیود جهت هرزگرد استفاده شده است تا جریان القایی از بوبین رله را حذف کند. اگر این دیود استفاده نشود، ترانزستور در برابر جریان القایی بوبین رله آسیب پذیر خواهد بود.



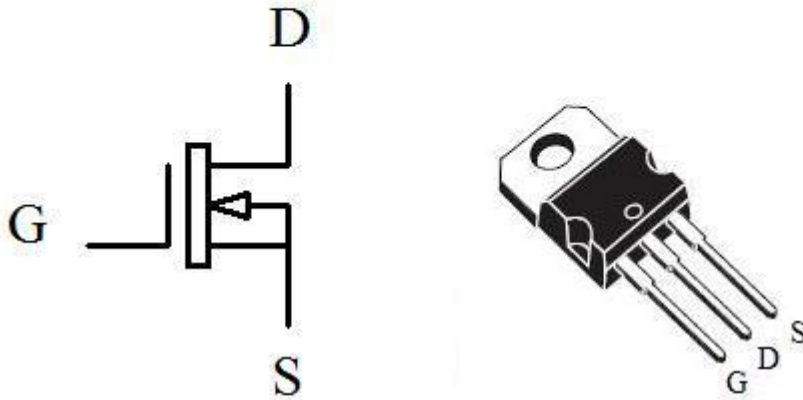
شکل ۵-۱۱ دیود 1N4148



Q۲ : ترانزیستور ماسفت

مشخصات این ترانزیستور از قرار زیر است:

IRF540

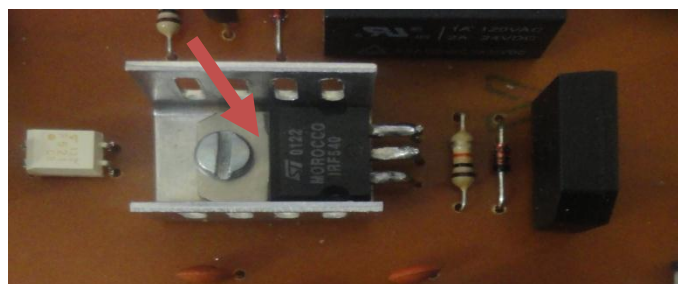


**Description:**

N-Channel MOSFET. Max. Voltage: 100V, Full Scale Current: 28A, Max. Power: 150W

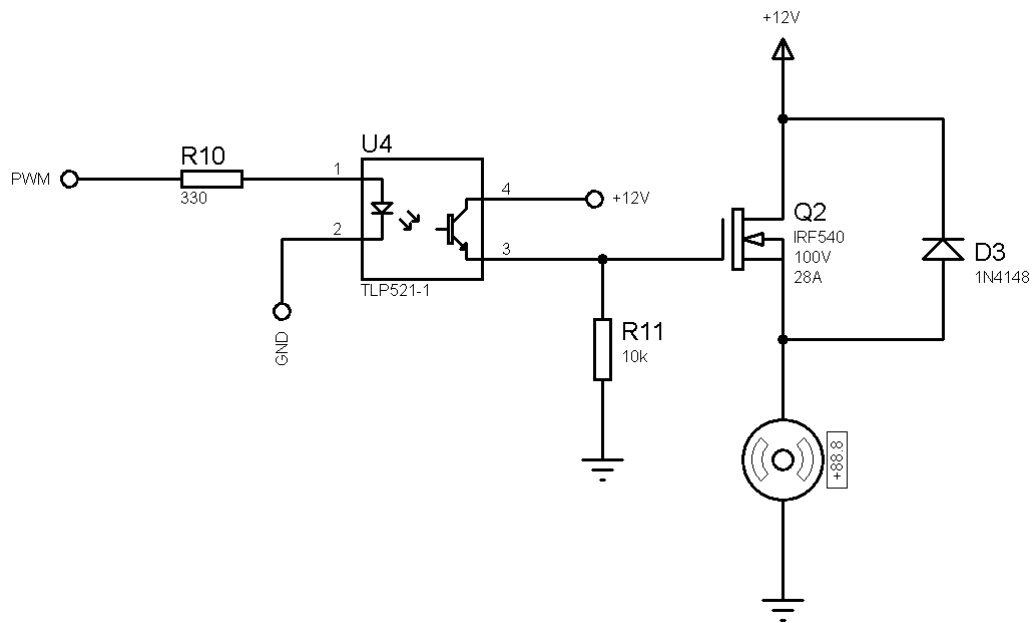
**Replacements:**

این ترانزیستور به صورت سری در مسیر تغذیه موتور DC قرار دارد تا با استفاده از سیگنال PWM که به گین آن اعمال خواهد شد، بتوانیم سرعت موتور را کنترل کنیم.



شکل ۵-۱۲ ترانزیستور ماسفت

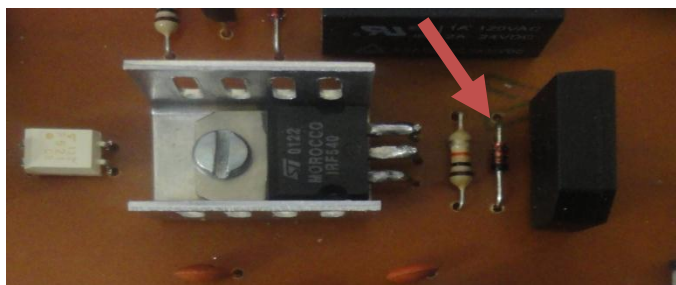
با کمی تحریف در شماتیک اصلی می توان گفت که این ترانزیستور به صورت شماتیک زیر درایو شده است:



شکل ۵-۱۳ شماتیک اتصال ترانزیستور ماسفت

در صورت تحریک گین، ترانزیستور روشن شده و موتور شروع به حرکت خواهد کرد.

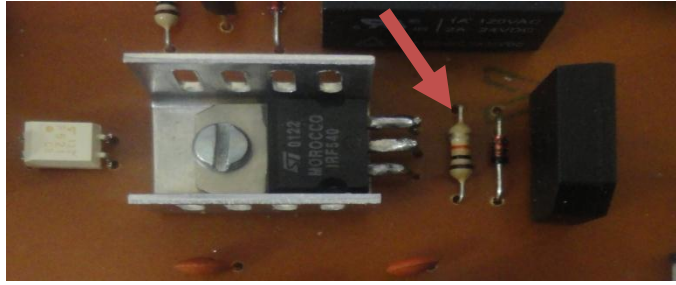
D1: دیود شیشه ای به نام ۱N۴۱۴۸



شکل ۵-۱۴ دیود ۱N۴۱۴۸

دیود جهت هرزگرد استفاده شده است تا جریان القایی از سیم پیچ های داخلی موتور DC را حذف کند. اگر این دیود استفاده نشود، ترانزیستور در برابر جریان القایی موتور آسیب پذیر خواهد بود.

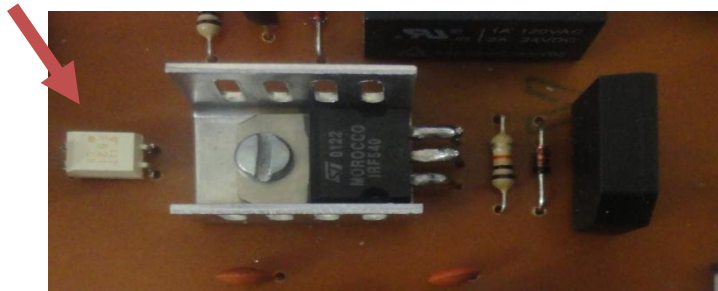
R11 : مقاومت ۱۰K اهم



شکل ۵-۱۵ مقاومت

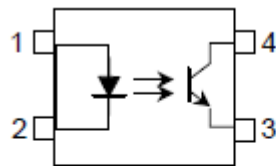
این مقاومت گین ترانزیستور را PULL DOWN کرده است. زمانی که هیچ سیگنالی به گین ترانزیستور اعمال نشود ترانزیستور را در حالت غیر فعال نگه میدارد.

U4 : اپتوکپلر نوری TLP521



شکل ۵-۱۶ اپتوکپلر نوری

شماتیک این اپتوکپلر از قرار زیر است:

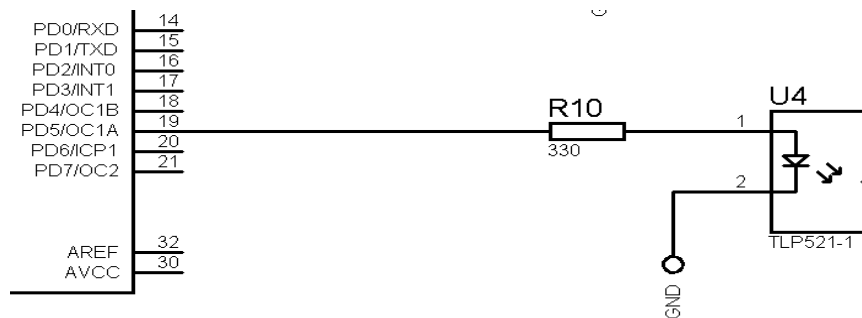


- 1 : Anode
- 2 : Cathode
- 3 : Emitter
- 4 : Collector



شکل ۵-۱۷ شماتیک اپتوکپلر

جهت ایزوله کردن قسمت درایور از قسمت کنترلر، از این اپتوکپلر استفاده شده است.



شکل ۵-۱۸ شماتیک اپتوکپلر

در داخل این تراشه یک جفت فرستنده و گیرنده نوری قرار دارد که با روشن شدن فرستنده، اطلاعات به گیرنده منتقل می شود. به دلیل اینکه هیچ ارتباط الکتریکی بین بخش ورودی با بخش خروجی ندارد، هیچ نویزی از قسمت درایور به قسمت کنترلر منتقل نمی شود. از آنجایی که قرار است با استفاده از سیگنال PWM دور موتور را کم و زیاد کنیم. بنابر این پایه ۱ از اپتوکپلر را به پایه OC1A از میکروکنترلر متصل میکنیم. این پایه در میکروکنترلر ATmega16 همان پایه ۱۹ می باشد. در زمان برنامه نویسی، جهت ایجاد سیگنال PWM از Timer ۱ میکروکنترلر استفاده خواهیم کرد. خروجی سگنال PWM در Timer ۱ همین پایه OC1A است.

R10: مقاومت ۳۳۰ اهم

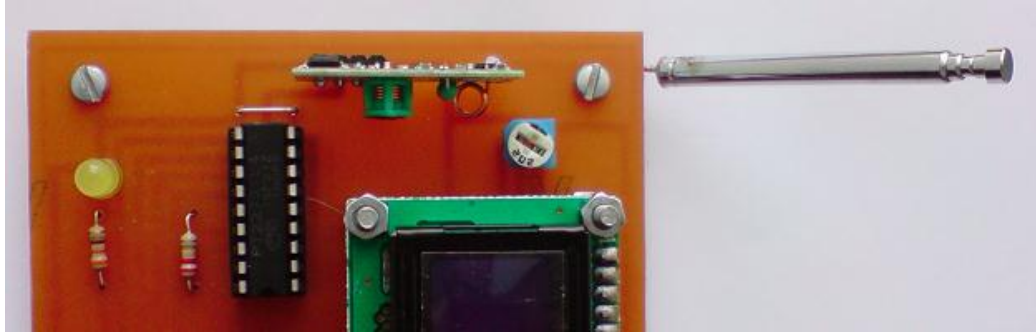


شکل ۵-۱۹ مقاومت

جهت کم کردن جریان استفاده شده است. اگر این مقاومت نباشد، احتمال آسیب دیدن اپتوکپلر وجود دارد

## فصل ششم

### گیرنده رادیویی



شکل ۱-۶ گیرنده رادیویی

#### ۱-۶ آشنایی با ماژول رادیویی ASK

ارتباط بین گیرنده و فرستنده از طریق سیگنال های رادیویی می باشد. برای این کار نیاز به فرستنده و گیرنده رادیویی داریم که بتواند اطلاعات مورد نظر ما را فرستاده و در گیرنده ، همان اطلاعات را بی کم و کاست آشکار کند. در این پروژه ما از ماژول های & TWS-315 RWS-315 استفاده کردیم.



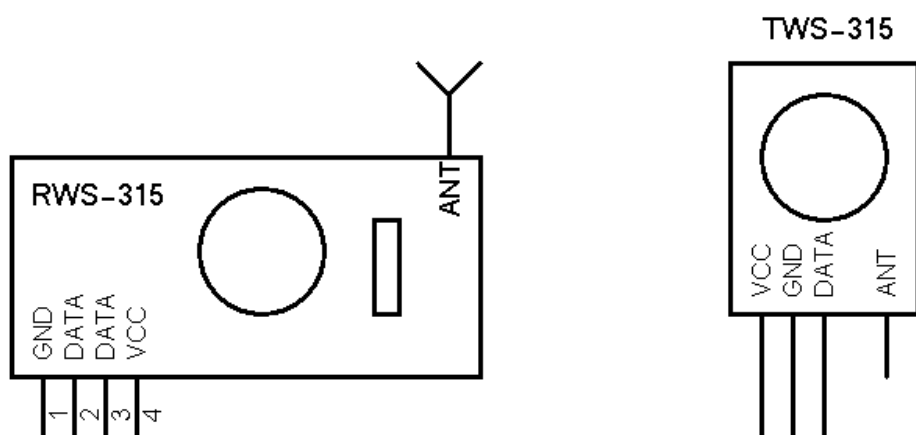
RWS-315



TWS-315

شکل ۲-۶ ماژول رادیویی ASK

ترتیب پایه های این ماژول ها از قرار زیر است.



شکل ۶-۳ دیتاشیت ماژول

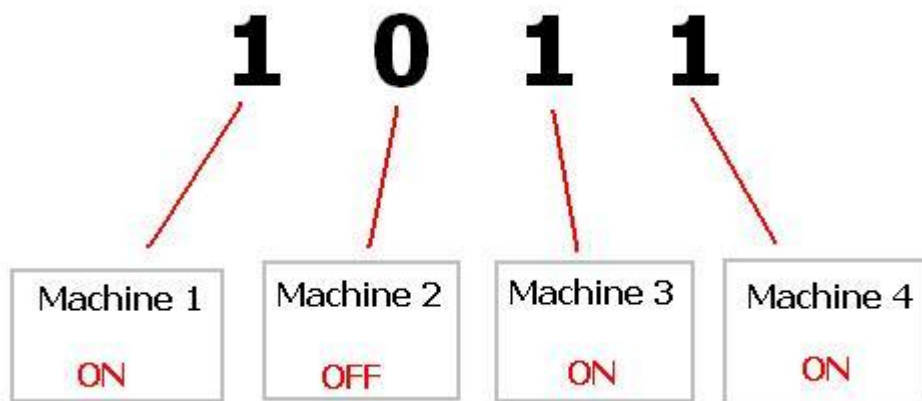
TWS-۳۱۵:

این ماژول از نوع **ASK** بوده و با فرکانس ۳۱۵ مگاهرتز کار می کند . تغذیه ماژول ۴-۱۲ ولت (حالت متوسط ۵ ولت در نظر گرفته شود) می باشد . این ماژول در فضای آزاد دارای برد ۱۰۰ متر است .

RWS-۳۱۵:

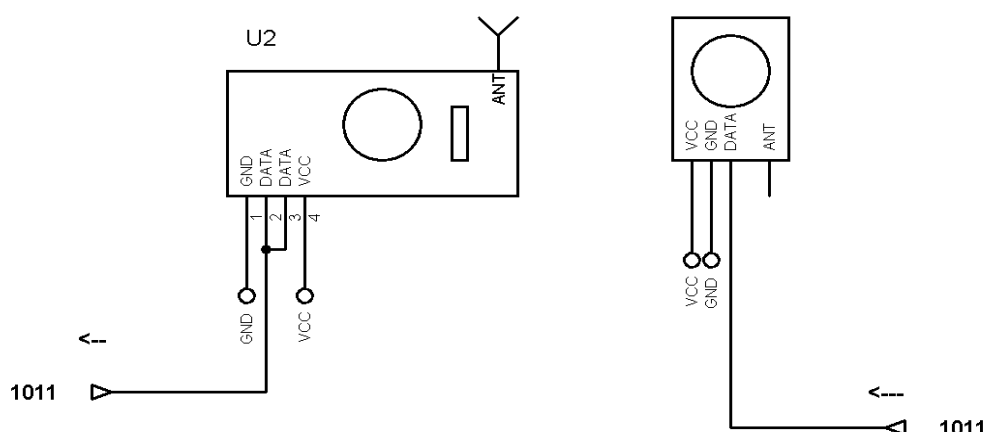
این ماژول از نوع **ASK** بوده و با فرکانس ۳۱۵ مگاهرتز کار می کند . تغذیه ماژول ۵ ولت می باشد و سرعت دریافت اطلاعات آن حدود ۶۰۰۰ بیت در ثانیه است . اطلاعات از پایه های ۲ و ۳ که متصل به یکدیگرند دریافت می شود. از این ماژول ها می توان برای فرستادن دیتای سریال استفاده کرد، به این صورت که بعد از اتصال ماژول ها به تغذیه، پایه ۲ از TWS-۳۱۵ یعنی پایه **DATA** آماده دریافت اطلاعات می باشد، برای کنترل ۴ وسیله می توان ۴ بیت را پی

در پی به پایه DATA از ماژول فرستنده فرستاد. مثلا ۱۰۱۱ هر کدام از بیت ها به ترتیب بیانگر وضعیت یکی از وسایل برقی میباشد. مانند شکل زیر:



شکل ۴-۶ بیانگر وضعیت

برای فرستاده ۰ از سطح ولتاژ زمین و برای ۱ از سطح ولتاژ مثبت استفاده می کنیم. در صورتی که تغذیه ماژول گیرنده را درست وصل کرده باشیم می توانیم همین دیتا را در گیرنده مشاهده کنیم مانند شکل زیر:



شکل ۵-۶ نمونه ارسال دیتا

فاصله این دو ماژول از یکدیگر در فضای آزاد می تواند تا ۱۰۰ متر باشد. بعد از دریافت ۴ بیت در ماژول گیرنده میتوان متناسب با هر کدام از بیت ها، دستگاه متناسب با آن را روشن یا خاموش کرد.

## ۶-۲ آشنایی با تراشه های انکدر و دیکدر

با ماژول فرستنده و گیرنده امواج رادیویی آشنا شدیم و شاهد بودیم که راه اندازی آنها چقدر ساده می باشد، ولی این روش یک عیب بزرگ دارد. چنانچه شخص دیگری ماژولی از این نوع را در اختیار داشته باشد، می تواند به اطلاعات شخصی ما دسترسی پیدا کند. برای جلوگیری از این مورد ما نیاز به کد کردن اطلاعات داریم، برای این کار روش های مختلف زیادی وجود دارد ولی ما در اینجا از IC های انکدر PT۲۲۶۲ و دیکدر PT۲۲۷۲ استفاده کردیم. این IC ها ساخت شرکت PTC می باشد. توجه داشته باشید که این IC انحصار برای یکدیگر می باشند، یعنی برای این ای سی گیرنده نمی توان از فرستنده دیگری استفاده کرد و همین طور عکس آن. این IC ها از تکنولوژی CMOS می باشد و محدوده ولتاژ کاری آن ها از ۳ تا ۱۵ ولت است. هر دو ای سی ۱۸ پایه هستند و می توانند تا حداکثر ۱۲ پایه و حداقل ۶ پایه برای آدرس داشته باشند. پایه های آدرس می توانند در سه حالت زیر قرار گیرند.

۱. زمین

۲. مثبت

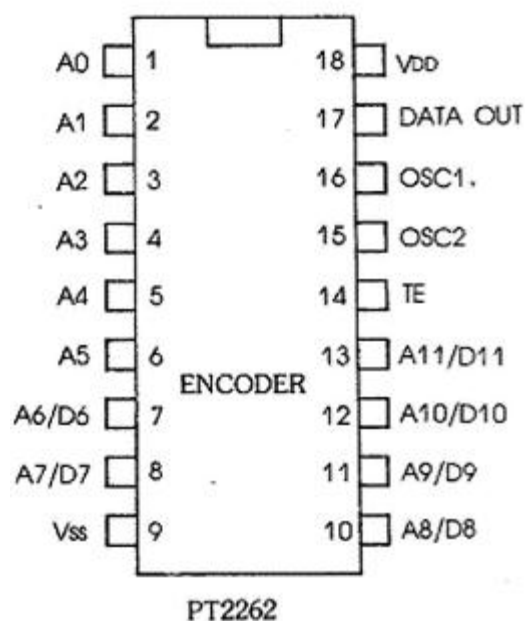
۳. رها بماند ( بدون اتصال به جایی )



برای اینکه فرستنده و گیرنده کار بکنند باید آدرس فرستنده و گیرنده یکی باشد. در این مورد جلوتر توضیحات کاملتری داده خواهد شد.

: PT2262

این IC یک انکدر می باشد پایه های این IC از قرار زیر است:



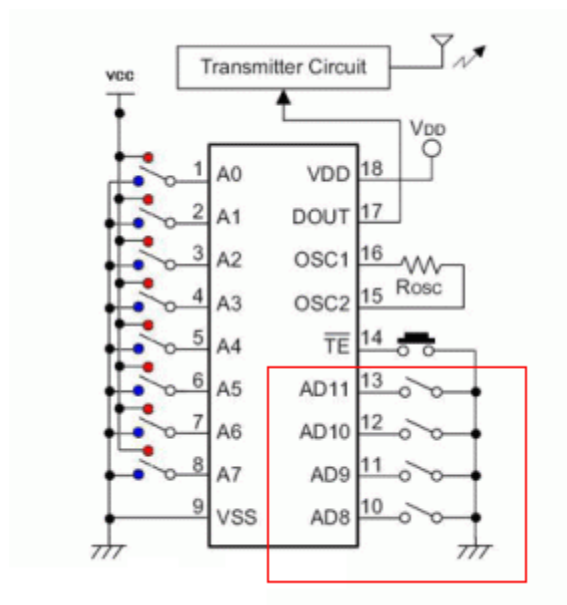
شکل ۶-۶ دیتا شیت انکدر

VDD و VSS: این دو پایه برای تغذیه IC می باشد، پایه VDD برای مثبت تغذیه و VSS برای منفی تغذیه.

A<sub>0</sub> تا A<sub>11</sub>: این ۱۲ پایه خط آدرس این آی سی می باشد، البته از A<sub>6</sub> تا A<sub>11</sub> می توان به عنوان دیتا استفاده کرد در شکل این پایه ها با نام D<sub>6</sub> تا D<sub>11</sub> مشخص شده اند. در این پروژه

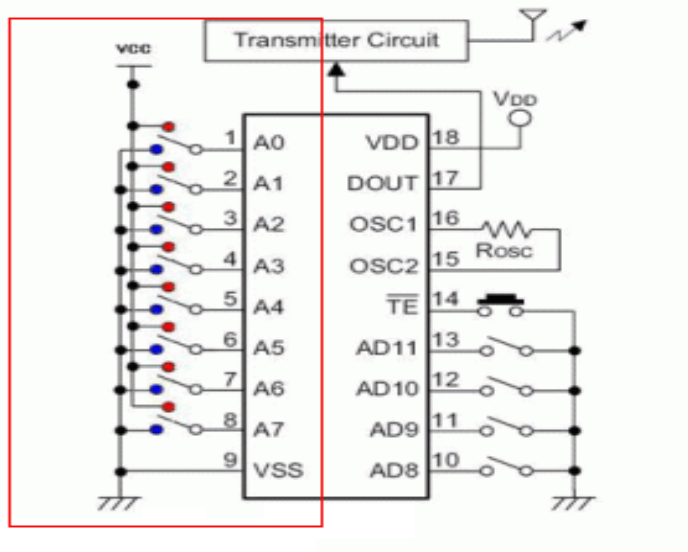
نیاز به ۴ کانال است پس از پایه های  $D_8$  ،  $D_9$  ،  $D_{10}$  ،  $D_{11}$  به عنوان خط دیتا و از پایه های  $A_0$  تا  $A_7$  به عنوان خط آدرس استفاده میکنیم .

منظور ما از خط دیتا همان ۴ بیتی است که برای کنترل دستگاه ها نیاز داریم.



شکل ۶-۷ دیتا شیت انکدر

هر کدام از پایه های  $D_8$  ،  $D_9$  ،  $D_{10}$  ،  $D_{11}$  بیانگر یک بیت کنترل است. این پایه ها می تواند در ۲ وضعیت صفر یا یک منطقی قرار داشته باشد، برای این کار ما از دیپ سویچ استفاده خواهیم کرد، که در جلوتر شرح مداری آن کامل توضیح داده خواهد شد. توجه داشته باشید که این پایه ها در هر وضعیت منطقی که باشند، همان منطق در IC دیکدر، در پایه هم نام خود، ظاهر خواهد شد.



شکل ۶-۸ خط آدرس انکودر

از خط آدرس برای کد گذاری فرستنده استفاده می شود.

منظور از کد گذاری بدین صورت می باشد که شما می توانید برای هر فرستنده و هر گیرنده یک کد خاص تعریف کنید و در صورت مطابقت کد فرستنده و گیرنده دستگاه عمل خواهد کرد این امکان جالب شما را قادر می سازد تا ۳ به توان ۸ یعنی ۶۵۶۱ فرستنده و گیرنده بسازید و اگر آن ها را در کنار یک دیگر قرار دهید فرستنده و گیرنده ها بدون هیچ گونه مشکلی کار خواهد کرد مثلا اگر فرستنده با کد ۱۰۱۱۰۰۰۰ را روشن کنید و کلید ۱ را فشار دهید فقط یک گیرنده عمل خواهد کرد که کد آن با فرستنده یکی باشد و خروجی شماره ۱ آن فعال خواهد شد.

همین امکان نیز در دزدگیر های ماشین وجود دارد بصورتی که اگر مثلا ۱۰۰ ماشین که از یک نوع دزدگیر استفاده می کنند در یک مکان باشند هیچ وقت اشتباهی درب ماشین دیگری باز نمی شود و دقیقا ماشینی که فرستنده آن فشار داده شده است عمل خواهد کرد.

خوب در مورد نحوه کد گذاری باید بگم ما در مدار فرستنده و گیرنده ۴ کاناله از یک آی سی انکودر ۴ کاناله استفاده کردیم که خروجی آن وصل مدار فرستنده می شد. در این ای سی پایه های ۱ الی ۸ پایه خط آدرس می باشد که می توانید آن را به منفی یا مثبت وصل کنید و یا آزاد بگذارید.

برای بهتر متوجه شدن مثلا اگر شما پایه های ۱ تا ۴ را به مثبت وصل کنید و پایه های ۵ و ۶ را به منفی وصل کنید و پایه ۷ و ۸ را آزاد بگذارید در حقیقت یک کد برای دستگاه خود تعریف کرده اید و در گیرنده نیز به همین صورت پایه ها را باید وصل کنید تا گیرنده شما بتواند اطلاعات ارسالی از فرستنده را دریافت کند و عمل کند البته این نکته را هم متذکر بشم که محدودیت در نحوه وصل کردن پایه وجود ندارد اصلا میتوانید همه پایه را آزاد بگذارید یا همه را به منفی وصل کنید یا همه را به مثبت در این زمینه هیچ گون محدودیتی وجود ندارد و فقط باید دقت کند که کد فرستنده و گیرنده یکی باشد تا اطلاعات شما ارسال و دریافت شود.

**DATA OUT** : این پایه خروجی انکودر است که به فرستنده رادیویی متصل می شود. در ای سی فرستنده PT۲۲۶۲ اطلاعات آدرس و دیتای آن از پایه ۱۷ که با نام **DATA OUT** مشخص شده خارج می شود.

**TE** : زمانی که در فرستنده اطلاعات ۴ بیتی در ورودی قرار گرفت برای ارسال باید یک لحظه پایه ۱۴ که با نام **TE** مشخص شده است را به منفی وصل کرد تا اطلاعات ارسال شود.

**OSC۱** و **OSC۲** : این دو پایه مربوط به اسیلاتور داخلی **IC** می باشد ، توضیحات بیشتر در قسمت مربوط به PT۲۲۷۲ داده خواهد شد.

PT2272:

ای سی PT2272 بر 4 نوع هستند.

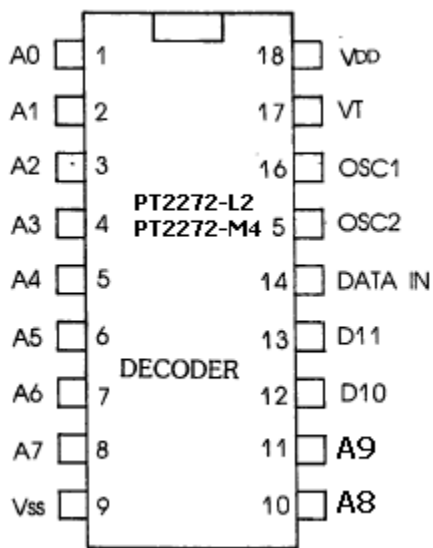
1. PT2272

2. PT2272-L2/M2

3. PT2272 L4/M4

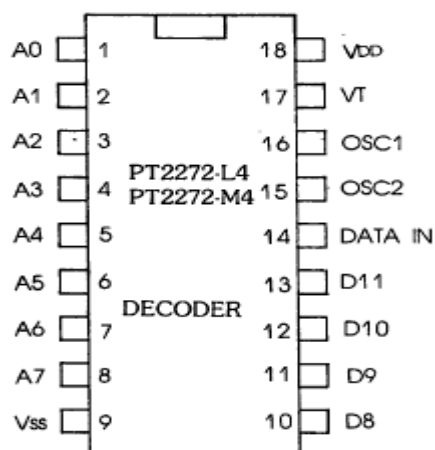
4. PT2272 L6/M6

در نوع دوم که به دو دسته PT2272-L2 و PT2272-M2 تقسیم می شوند خروجی ان ها 2 بی تی می باشد. و تفاوت ان ها در این است که در نوع M2 دیتای خروجی تا زمانی که بین فرستنده و گیرنده ارتباط برقرار است وجود دارد و در صورتی که ارتباط بین فرستنده و گیرنده قطع شود این خروجی ها نیز تغییر حالت پیدا خواهند کرد که این خاصیت جالب را میتوان در مدارات با کاربرد خاص استفاده نمود. اما در نوع L2 دیتای خروجی تا برقراری ارتباط بعدی دیتا در پایه های مربوط باقی خواهند ماند و در صورتی که در این بین ارتباطی بین فرستنده و گیرنده نباشد مشکلی پیش نخواهد آمد و خروجی ها تغییر پیدا نخواهند کرد.



شکل ۶-۹ دیتاشیت دیکودر

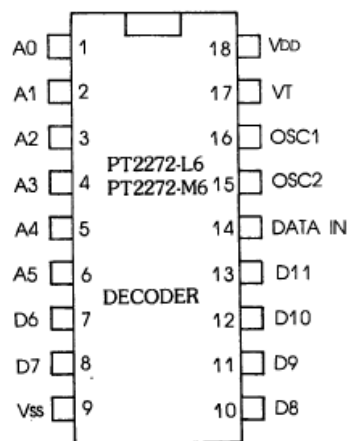
در نوع سوم که به دو دسته  $PT2272-L4$  و  $PT2272-M4$  تقسیم می شوند خروجی ان ها ۴ بیتی می باشد. و تفاوت ان ها در این است که درنوع  $M4$  دیتای خروجی تا زمانی که بین فرستنده و گیرنده ارتباط برقرار است وجود دارد. اما در نوع  $L4$  دیتای خروجی تا برقراری ارتباط بعدی دیتا در پایه های مربوط باقی خواهند ماند.



شکل ۶-۱۰ دیتاشیت دیکودر

در نوع چهارم از گیرنده که به دو دسته PT2272-L6 و PT2272-M6 تقسیم می شوند خروجی آن ها ۶ بیتی می باشد. و تفاوت آن ها در این است که در نوع M6 دیتای خروجی تا زمانی که بین فرستنده و گیرنده ارتباط برقرار است وجود دارد اما در نوع L6 دیتای خروجی تا برقراری ارتباط بعدی دیتا در پایه های مربوط باقی خواهند ماند.

لازم به ذکر است که ای سی فرستنده PT2262 می تواند اطلاعات ۴ بیتی و هم اطلاعات ۶ بیتی ارسال کند.



شکل ۶-۱۱ دیتاشیت دیکودر

این هم جدولی از انواع گیرنده ها که درباره آن توضیح دادیم:

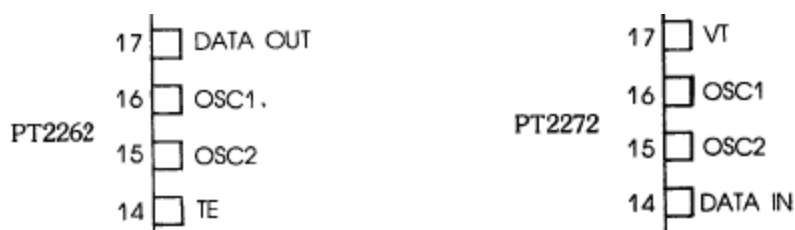
تشریح پایه های PT2272 :

دیگر نیز مانند انکدر پایه های دیتا و آدرس را دارد، باید توجه داشته باشید که پایه های آدرس دیگر باید درست مانند پایه های انکدر وصل شوند. اگر این مسئله درست رعایت نشود، شما هیچ اطلاعاتی را دریافت نخواهید کرد.

**DATA IN** : پایه ای که اطلاعات را دریافت می کند پایه ۱۴ می باشد که با نام **DATA IN** مشخص شده است.

**VT**: زمانی که اطلاعات دریافت شد پایه ۱۷ که با نام **VT** مشخص شده است ۱ منطق (**VCC**) می شود که با وصل کردن یک **LED** به این پایه می توان از دریافت اطلاعات با خبر شد.

**OSC1** و **OSC2** : ای سی های فرستنده و گیرنده دارای اسیلاتور داخلی هستند و بین دو پایه **OSC1** و **OSC2** باید طبق نمودار داده شده در دیتا شیت آن ها مقاومت مناسب قرار داد.



شکل ۶-۱۲ اسیلاتور داخلی



در اینجا ۳ نوع از این مقاومت ها آورده شده است:

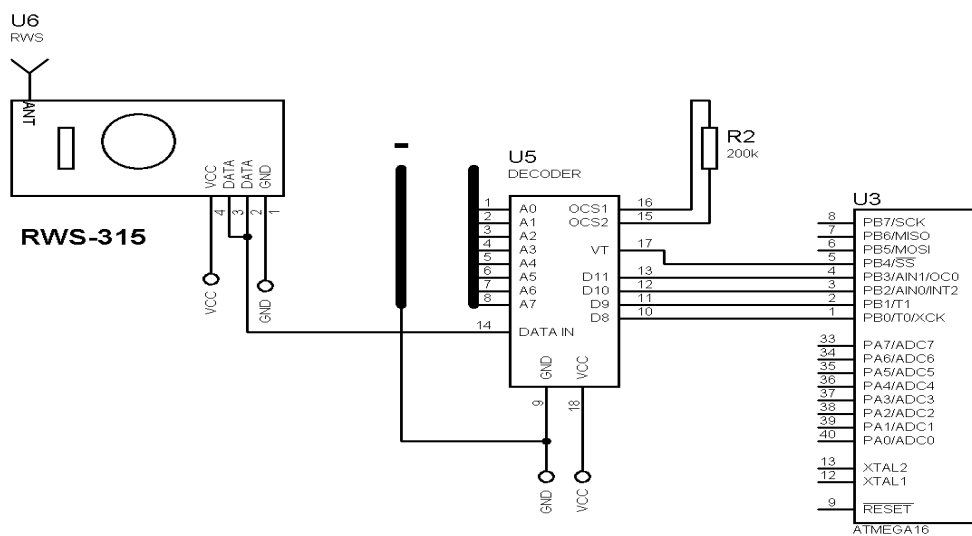
Suggested oscillator resistor values	
PT2262	PT2272
4.7 MΩ	820 KΩ*
3.3 MΩ	680 KΩ*
1.2 MΩ	200 KΩ**

شکل ۶-۱۳ مقاومت اسیلاتور

مثلا اگر مقاومت اسیلاتور فرستنده را ۴۰۷ مگا اهم قرار دادید باید مقاومت اسیلاتور گیرنده را ۸۲۰ کیلو اهم قرار بدهید.

حال شما با وصل کردن فرستنده TWS ۴۳۴ به ای سی فرستنده PT۲۲۶۲ و وصل کردن گیرنده RWS ۴۳۴ به ای سی گیرنده PT۲۲۷۲-XY می توانید اطلاعات خود را کد بندی شده تا مسافت حدود ۱۰۰ متری ارسال کنید.

### ۳-۶ شماتیک گیرنده رادیویی



شکل ۶-۱۴ شماتیک گیرنده رادیویی

U6 : ماژول گیرنده امواج رادیویی از نوع ASK

جهت دریافت امواج رادیویی از این ماژول استفاده شده است. این ماژول امواج رادیویی با فرکانس ۳۱۵MHz را دریافت کرده و پس از آشکار سازی خروجی آن از طریق پایه DATA قابل دسترسی است.

U5 : تراشه دیکودر P22272-L4

اطلاعات کد شده را از ماژول دریافت کرده و پس از دیکد کردن به صورت باینری در خروجی ظاهر میکند. خروجی ها از طریق پایه های D8, D9, D10, D11 قابل دسترسی است. این تراشه از نوع L4 می باشد. این یعنی اینکه خروجی آن چهار کاناله از نوع Lach میباشد، و اطلاعات تا زمان رسیدن اطلاعات بعدی در خروجی میمانند.

R2 : مقاومت ۲۲۰K اهم

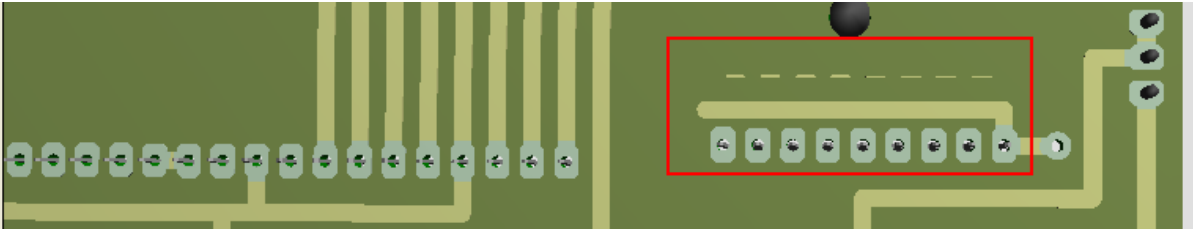
این مقاومت باید متناسب با لیست زیر انتخاب شود

PT2262	PT2272
4.7 MΩ	820 KΩ*
3.3 MΩ	680 KΩ*
1.2 MΩ	200 KΩ**

به علت موجود نداشتن مقاومت ۲۰۰K اهم از نزدیکترین رنج استفاده شد. استفاده از مقاومت ۲۲۰K اهم مشکلی ایجاد نمیکند.

توجه داشته باشید که مقاومت های استفاده شده در دیکدر PT2272 و انکدر PT2262 باید متناسب با جدول بالا باشند.

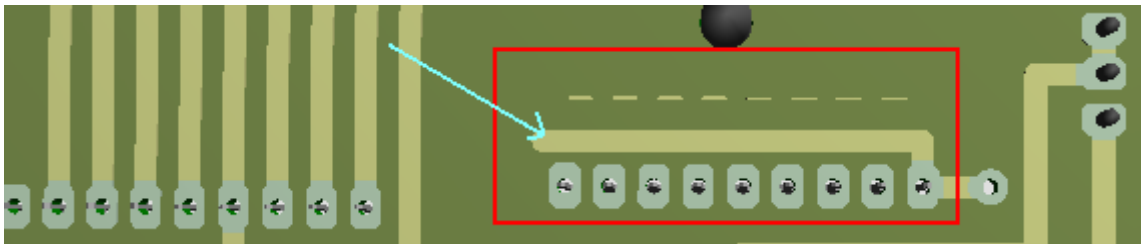
توجه: خط آدرس برد گیرنده در تصویر زیر مشخص شده است



شکل ۶-۱۵ خط آدرس برد گیرنده

از خط آدرس برای کد گذاری فرستنده استفاده می شود،

در این پروژه این خط آدرس ها به صورت پیش فرض رها شده اند، یعنی به جایی وصل نشده اند. و همین حالت در برد فرستنده نیز تنظیم شده است. در صورتی که امنیت سیگنال ها برای ما مهم باشد باید خط آدرس را تغییر دهید. برای این کار میتوانید پایه های مربوط به خط آدرس را به Trac کناری وصل گرد.



شکل ۶-۱۶ Trac

با این کار آن پایه به سطح GND متصل خواهد شد.

نحوه اتصال خط آدرس در برد گیرنده و برد فرستنده باید با هم مطابقت داشته باشند. در غیر این صورت برد گیرنده، سیگنال های برد فرستنده را نخواهد شناخت.

## فصل هفتم

### برد فرستنده

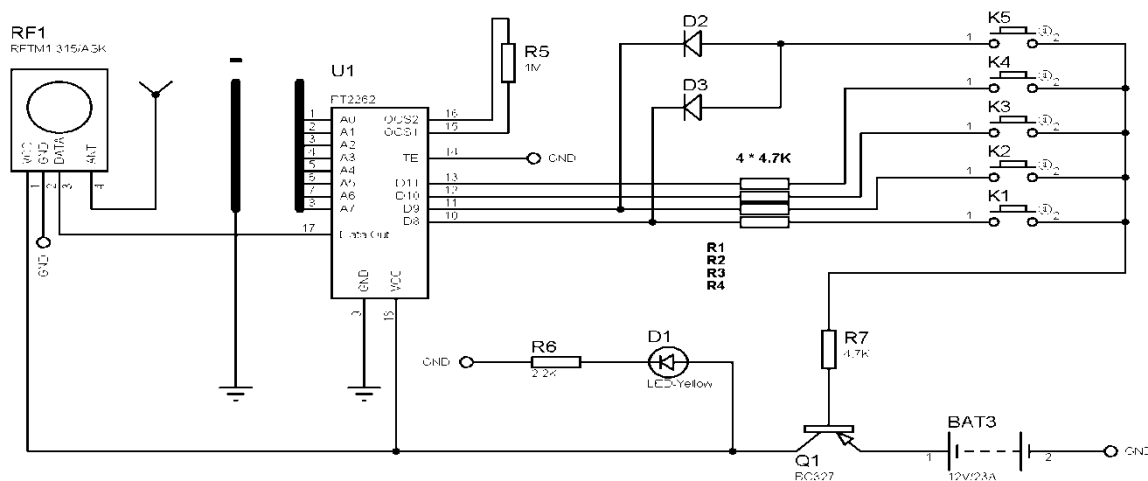


شکل ۱-۷ برد فرستنده

برد فرستنده از نظر فرکانس سیگنال ارسال شده، نوع مودلاسیون و نوع کد گذاری باید با برد گیرنده مطابقت داشته باشد.

### ۱-۷ شماتیک برد فرستنده

شماتیک برد فرستنده از قرار زیر است:



شکل ۲-۷ شماتیک برد فرستنده

BAT3 : باتری ۱۲ ولت ۲۳ آمپر

برای این پروژه از یک باتری ریموتی که اندازه کوچکی دارد استفاده شده است.



شکل ۳-۷ باتری

برای قرار دادن این باتری بر روی برد، دو سر آن به برد، لحیم شده است. این کار برای محکم کردن باتری کافیست.

Q1 : ترانزیستور PNP به شماره BC327



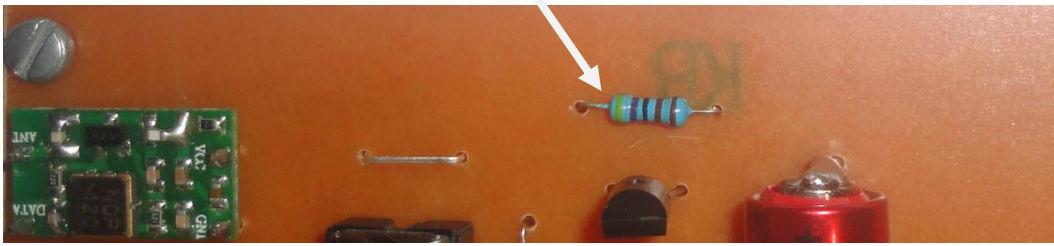
شکل ۴-۷ ترانزیستور BC327

با قرار دادن این ترانزیستور سر راه باتری، میتوان تغذیه کل مدار را قطع یا وصل کرد.

در حالت عادی که هیچ کلیدی فشرده نشود، این ترانزیستور غیر فعال بود و تغذیه مدار قطع است. فقط زمانی که یکی از کلیدها فشرده شود این ترانزیستور فعال شده و تغذیه مدار را وصل میکند. تراشه انکدر و ماژول فرستنده رادیویی تغذیه خود را از مسیر ترانزیستور میگیرند. و با فعال شدن ترانزیستور، تغذیه آنها نیز تعیین می شود.

از آنجایی که در حالت عادی تغذیه مدار قطع است، در مصرف باتری صرفه جویی شده و عمر باتری را افزایش میدهد.

RV : مقاومت ۴.۷K اهم



شکل ۵-۷ مقاومت

این مقاومت برای محدود کردن جریان بیس ترانزیستور استفاده شده است. در صورت تحریک مستقیم بیس ترانزیستور امکان آسیب دیدن ترانزیستور است.

D۱ : LED مات زرد رنگ



شکل ۶-۷ LED

این LED به کلکتور ترانزیستور متصل شده است تا در زمان فعال شدن ترانزیستور روشن شود. با روشن شدن این LED میتوانیم از صحت فشردن کلیدها مطمئن شویم.

R۶ : مقاومت ۲.۲K اهم



شکل ۷-۷ مقاومت

از این مقاومت برای محدود کردن جریان LED استفاده شده است.

توجه داشته باشید که یک LED مصرف کننده نیست. رفتار یک LED در مدار همانند دیود می باشد، برای همین هیچگاه یک LED را بدون یک مقاومت سری استفاده نشود. زیرا باعث سوختن LED خواهد شد.

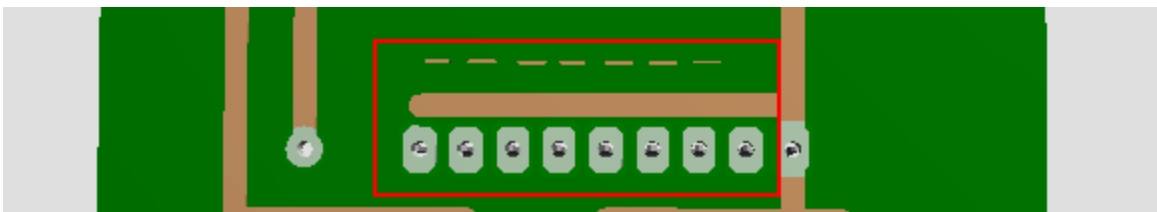
U1: انکدر PT2262



شکل ۷-۸ کد گذاری اطلاعات

از این تراشه جهت کد گذاری اطلاعات ارسالی استفاده می شود.

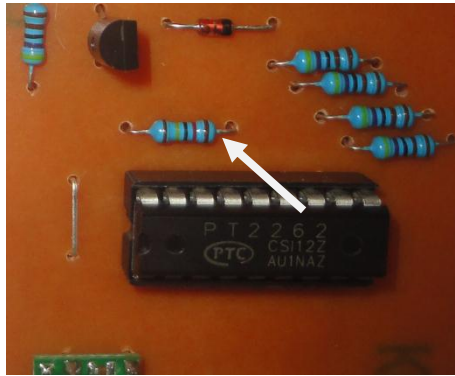
خط آدرس در شکل زیر مشخص شده است:



شکل ۷-۹ خط آدرس

حالتی که برای خط آدرس فرستنده تنظیم میشود باید مطابق با همان حالتی باشد که در برد گیرنده تنظیم شده است.

R5: مقاومت ۱M اهم



شکل ۷-۱۰ مقاومت

این مقاومت باید متناسب با جدول زیر انتخاب شود.

Suggested oscillator resistor values	
PT2262	PT2272
4.7 MΩ	820 KΩ*
3.3 MΩ	680 KΩ*
1.2 MΩ	200 KΩ**

به علت موجود نداشتن مقاومت ۱.۲M اهم از نزدیک ترین رنج ممکن استفاده شده است. استفاده از مقاومت ۱M اهم در این مورد مشکلی ایجاد نمیکند.

K1: تک سویچ فشاری

K2: تک سویچ فشاری

K3: تک سویچ فشاری

K4: تک سویچ فشاری

K5: تک سویچ فشاری

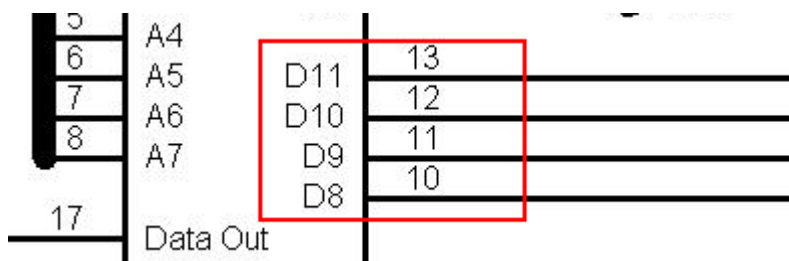
D2 و D3: دیود ۱N4۱۴۸





شکل ۷-۱۱ دیود ۱N۴۱۴۸

تراشه PT۲۲۶۲ دارای چهار پایه ورودی است که می توان برای چهار کلید مورد استفاده قرار گیرد،



شکل ۷-۱۲ پایه های اتصال دیتای ارسالی

برای استفاده از کلید پنجم، نیاز به ایجاد حالت ترکیبی می باشد. برای این کار از دو عدد دیود استفاده شده است تا در زمان فشردن کلید K۵، پایه های D۸ و D۹ را با هم تحریک کند. در صورت نیاز به کلید ها بیشتر از این روش میتوان برای ایجاد حالت جدید دیگری نیز استفاده کرد.

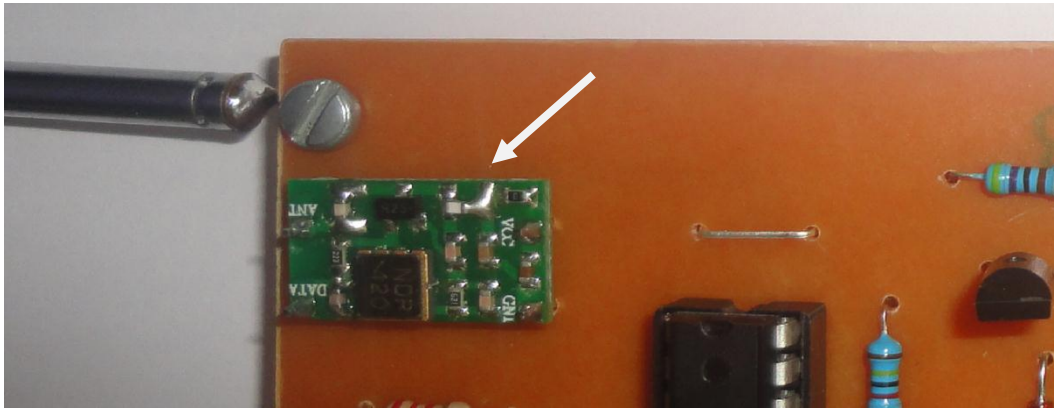
R۱ تا R۴ : مقاومت ۴.۷K اهم



شکل ۷-۱۳ مقاومت

از این مقاومت ها جهت محدود کردن جریان ورودی به پایه های ۱۰ تا ۱۳ تراشه PT۲۲۶۲ استفاده شده است.

RF۱ : ماژول فرستنده ۳۱۵MHz امواج رادیویی



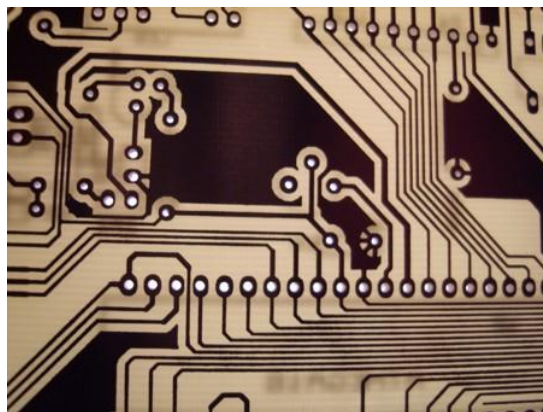
شکل ۷-۱۴ ماژول فرستنده

این ماژول جهت ارسال اطلاعات، از روش مدولاسیون ASK استفاده میکند.

برای این ماژول از آنتن ۱۲.۵CM استفاده شده است.

## فصل هشتم

### طراحی مدار PCB



شکل ۸-۱ PCB

این پروژه با مدار چاپی کار شده است. برای انجام این کار شکل PCB برد توسط نرم افزار طراحی شده و با استفاده از یکی از روش های معمول، این طرح بر روی برد مسی خام برگردانده شده و سپس برد را داخل اسید قرار میدهند.

#### ۸-۱ استاندارد طراحی مدار چاپی

تبدیل یک نقشه الکترونیکی به نقشه مدار چاپی باید طبق استانداردهای موجود و رعایت مسائلی انجام پذیرد. یک نقشه الکترونیکی میتواند در هر اندازه و شکلی کشیده شود اما مدار چاپی باید طوری تهیه شود که در آن استاندارد و اصول مدار چاپی و اندازه حقیقی المانها در نظر گرفته شود. طراحی مدار چاپی باید با رعایت فواصل پایه ها و حجم المانها متناسب با اندازه حقیقی انجام پذیرد. مثلاً اگر بخواهیم یک مقاومت ۲ وات را به طور افقی در فیبر قرار دهیم به علت اینکه فاصله بین دو پایه آن ۱۷ میلی متر است باید در طراحی مدار چاپی نیز حداقل ۱۷ میلی متر فاصله بین دو پایه در نظر بگیریم.

برای طراحی مدار چاپی از طرف کارخانه سازنده عناصر جداول و استانداردهائی در مورد پایه مقاومتها و دیودها و ترانزیستور و آی سی و غیره منتشر میگردد که طراحی مدار چاپی بر مبنای این جدولها انجام می پذیرد.

علاوه بر رعایت اندازه المانها نکات دیگری نیز باید در طراحی مدار چاپی در نظر گرفته شود. در زیر مهمترین نکاتی که باید در طراحی مدار چاپی رعایت شود آورده شده است.

- نقشه الکترونیکی بصورتی به نقشه مدار چاپی تبدیل شود که ورودیها در یک طرف و خروجیها در طرف دیگر قرار گیرند.
- عناصر حرارتی مانند مقاومتها و ترانزیستورهای پر وات کنار المانهای حساس به حرارت مانند دیودها و ترانزیستورهای کوچک قرار نگیرند.
- المانها به صورتی در مدار قرار گیرند که هنگام تعمیر به راحتی بتوان آنها را در آورد و تعویض نمود.
- در نظر گرفتن مکانهای که در نقشه الکترونیکی در نظر گرفته نمیشود مانند رادیاتورها و جای پیچ و غیره...
- در مدارهای فرکانس بالا تا جای که امکان دارد فاصله ها کوتاهتر و پایه المانها هم کوتاهتر باشد.
- پهنای خطوط ارتباط باید متناسب با جریان عبوری و مقاومت ایجاد شده باشد. ولتاژ وصل شده به مدار چاپی باید با در نظر گرفتن استانداردهای موجود انجام پذیرد. حداقل فاصله بین دو خط ارتباط متناسب با تغییر ولتاژ مدار تغییر مینماید. در صورتیکه فاصله خطوط با در نظر گرفتن ولتاژ مدار از حد مجاز کمتر شود باعث ایجاد جرقه یا ارتباط بین دو خط میگردد.

## ۲-۸ روش های فیبر مدار چاپی

برای تهیه فیبر مدار چاپی روشهای مختلفی وجود دارد، که در ادامه توضیحات با تعدادی از این روش ها آشنا می شویم.

### ۲-۸-۱ روش ماژیک

روش ماژیک یکی از روش ها ساده می باشد.

مواد لازم:

ماژیک واترپروف

اسید پاک کننده مس (کلرید آهن)

روش کار:

ابتدا یک ماژیک واترپروف (ضد آب) تهیه می کنید. توجه نمایید که حتما ضد آب و پر رنگ باشد. چون بعد از طراحی و آنهمه زحمت ممکن است داخل اسید از بین برود.

قبل از هر چیز فیبر مسی را کاملا با یک سمباده نرم (پوسته آب ۳۶۰) سمباده می زنید و اگر سمباده در دسترس نبود که اولاً تهیه نمایید اگر هم مقدور نبود آن را به هر نحوی تمیز نمایید. مطمئن شوید که هیچ لایه ای روی مس نباشد. چرب هم نباشد.

سپس با ماژیک تمام خطوط مورد نیاز را روی مس رسم نمایید. توجه نمایید که دست با مس تماس پیدا نکند چون باعث چرب شدن مس و در نتیجه داخل اسید اذیت خواهید شد. برای ایجاد اندازه های واقعی برای پایه های مثلا یک آی سی یک آی سی و یا هر قطعه دیگر که فاصله پایه های آن از همدیگر ۰.۱ اینچ هست تهیه نمایید و روی مس گذاشته و کنار آن با ماژیک بکشید.

حدالامکان از یک ماژیک نوک تیز استفاده نمایید. البته ماژیک مخصوص این کار در بازار موجود می باشد.

بعد از مطمئن شدن از مدار باید آن را داخل اسید بیاندازید.

## ۸-۲-۲ روش لتراست

روش لتراست یکی از آسان ترین روش هاست.

مواد لازم:

• لتراست

روش کار:

لتراست همان حروف برگردان است . با این تفاوت که به جای حروف خطوط و نقاط مداری روی آن کشیده شده است. ابتدا چند لتراست خطی و نقطه ای تهیه می کنید. انتخاب آن بسته به نظر خودتان و نوع مدارتان می باشد. توجه نمایید که حتما و یا لااقل تازه باشد چون برای برگرداندن روی فیبر مسی دچار مشکل خواهید شد.

قبل از هر چیز فیبر مسی را کاملا با یک سمباده نرم (پوسته آب ۳۶۰) سمباده می زنید و اگر سمباده در دسترس نبود که اولاً تهیه نمایید اگر هم مقدور نبود آن را به هر نحوی تمیز نمایید . مطمئن شوید که هیچ لایه ای روی مس نباشد. چرب هم نباشد.

سپس با لتراست خطوط مورد نیاز را روی مس رسم نمایید. برای برگرداندن خطوط و نقاط از روی لتراست به روی مس از ته خودکار و یا یک هم چین چیزی استفاده نمایید. توجه نمایید که دست با مس تماس پیدا نکند چون باعث چرب شدن مس و در نتیجه داخل اسید اذیت خواهید شد. در انتهای کار مطمئن شوید که خطوط و نقاط روی مس کاملا چسبیده باشد. اگر هم خطوطی نچسبیده بود راه حل دارد. از یک ششوار برای گرم کردن استفاده نمایید. با این کار خطوط و نقاط بعد از گرم شدن به سطح مسی خواهند چسبید.

بعد از مطمئن شدن از مدار باید آن را داخل اسید بیاندازید.

### ۸-۲-۳ روش لامینت

روش لامینت یکی از دقیق ترین روش ها می باشد.

#### مواد لازم:

- لامینت به سطح مقطع مورد نیاز (لامینت ماده ای ژلاتینی و حساس به نور است که بین دو لایه طلقی شفاف نازک قرار گرفته است)
- طلق ترانسپرننت یا فیلم و یا طرح چاپ شده مدار نگاتیو بر روی کاغذ کالک
- اسید (پرکلرودوفر)
- آب و ظرف پلاستیکی

#### روش کار:

برای شروع کار لازم است از طرح مورد نظرتان یک پرینت نگاتیو داشته باشید. پرینت نگاتیو را می توانید بر روی کاغذ کالک شفاف چاپ کنید یا به یکی از روش های عکاسی، به کمک یک فروشگاه مهرسازی از آن فیلم بگیرید. توجه نمایید که نقشه مدار چاپ شده شما حتماً باید نگاتیو باشد. (محل نوارهای مسی و خطوط خالی و بقیه جاهای مدار سیاه باشد) فیبر مسی را با سمباده نرم و یا با شستن، کاملاً تمیز و عاری از چربی نمایید. در یک جای کاملاً تاریک لامینت را از پوشش سیاه رنگ خود در آورید. توجه شود که این کار حتماً باید در یک محیط عاری از نور و تاریک انجام شود. همانطور که می بینید دو طرف لامینت یک لایه چسبیده شده که یکی از دیگری نرمتر است. آن را جدا کرده و لامینت را روی سطح مسی بچسبانید و کاملاً ماساژ دهید تا هیچ حبابی در زیر آن جمع نشود. اگر جایی هم دیدید هوا زیر لامینت قرار دارد با یک سوزن هوا را از آن خارج کنید.

فیلم و یا طلق چاپ شده از روی مدار را روی لامینت بگذارید و تنظیم نمایید. توجه داشته باشید که هنوز یک لایه به لامینت چسبیده است. یک شیشه دیگر روی طرح چاپی بگذارید تا کاملاً طرح به سطح لامینت بچسبد. تمام مراحل گفته شده باید در تاریکی انجام شود.

اکنون آنها را بدون اینکه تکان بخورند زیر نور خورشید برده و به مدت ۵ الی ۱۰ دقیقه زیر نور خورشید بگذارید. دوباره به تاریک خانه برگردید و لایه دوم را از روی لامینت جدا نمایید. این لایه از لایه قبلی ضخیم تر است.

مقدار یک قاشق غذا خوری پودر سفید رنگ ظهور لامینت را در یک لیوان آب حل نمایید. فیبر مدار را در آب بیاندازید و به آرامی تکان دهید تا طرح مدار ظاهر شود. (یعنی لامینت هایی که به آنها نور نتابیده کاملاً پاک شود) حالا فیبر را زیر آب کاملاً بشوید و با دست به آرامی روی آن بکشید تا لیزی حاصل از ماده ظهور از روی سطح مسی کاملاً پاک شود. حال پس از خشک کردن کامل فیبر، آن را از تاریک خانه خارج کنید. اکنون فیبر آماده اسیدکاری است. بعد از اتمام اسید کاری برای پاک کردن لامینت باقی مانده روی سطح مسی، با رعایت ایمنی، از محلول سود سوزآور استفاده نمایید.

#### ۸-۲-۴ روش اسپری پزتیو

مواد لازم:

اسپری پزتیو ۲۰ (positiv ۲۰)

سود سوز آور

طلق تراسپرنت یا فیلم

روش کار:

ابتدا یک اسپری پزتیو ۲۰، تازه تهیه نمایید. چرا که اگر مانده باشه در حقیقت فاسد شده است و آن حساسیت لازم در برابر نور را ندارد سود سوز آور مثل سنگ نمک خورده شده می ماند،



آن را می توانید از مغازه های فروش تجهیزات پزشکی تهیه نمایید. ابتدا باید طرح مدار را تهیه نمایید. می توانید طرح را چاپ کرده و آن را به بازار و مغازه های مختلف مه‌رسازی، چاپ سیلک و چاپخانه ها بدهید که از روی آن فیلم بگیرند. (خطوط سیاه و بقیه خالی) یا بهترین کار این است که چند طلق ترانسپرننت که قابلیت چاپ با پرینتر لیزری را دارند را تهیه نمایید. طرح خود را روی طلق با یک چاپگر دقیق لیزری چاپ نمایید. بهتر است از طرحتان دو تا پرینت بگیرید و روی هم بگذارید که دیگر نوری از جاهای چاپ شده عبور نکند. فیبر مسی را با یک سمباده نرم کاملا تمیز و عاری از هرگونه چربی نمایید. و پس از شستن و خشک کردن برد در یک مکان تاریک و بدور از گرد و غبار و وزش باد با استفاده از اسپری روی آن را رنگ پزتیو بزنید. با فاصله ۳۰ سانتی و با زاویه ۴۵ درجه و با حرکت منظم دست روی سطح مسی را بپوشانید.



شکل ۸-۲ اسپری پزتیو

لازم نیست که چند دست بزنید. یک دست هم کافیت به شرطی که مطمئن شوید همه جای آن را پوشانده است. مواظب باشید که یکنواخت باشد و جایی از جای دیگر بیشتر نباشد و یا شره نکند. فیبر را حتما افقی بگیرید که رنگ روی فیبر بماند و شره نکند. حالا نوبت به خشک کردن فیبر می باشد. می توانید آن را در جای تاریک به مدت ۱۲ تا ۲۴ ساعت به حال خود بگذارید. یا از طریق گرم کردن آن رو خشک کنید. اگه دستگاه مخصوص خشک کن فیبر رو دارید که هیچ در غیر اینصورت باید آن را داخل یک ظرف فلزی گذاشته و روی یک اجاق با حرارت خیلی ملایم بگذارید تا جایی که دمای داخل آن تقریبا ۷۰ درجه شود. بهترین

ظرف یک قابلمه بزرگ است که کف آن را چند تکه فلز بگذارید و فیبر را روی آن تا گرما مستقیماً به فیبر نخورد. بعد از ۱۰ الی ۱۵ دقیقه فیبر کاملاً خشک شده است. تمام این مراحل در محل تاریک انجام شود. حالا یک ظرف بزرگ تهیه نمایید یک لامپ گازی یا مهتابی کوچک و یا دو سه لامپ معمولی (بسته به اندازه طرحتان دارد) را داخل ظرف بگذارید. یک شیشه روی ظرف بگذارید و طرح چاپ شده یا فیلم را روی آن . سپس فیبر روی طرح گذاشته به صورتی که سطح مسی رنگ خورده رو به پایین و روی طرح چاپ شده باشد. توجه نمایید که روی فیبر را با یک وزنه و یا یک شیشه دیگر سنگین نمایید که فیبر کاملاً با فشار زیاد در مدت نور دهی به طرح بچسبد. تمام این مراحل در جای تاریک باید انجام شود. حالا لامپ و یا مهتابی را روشن نمایید. به مدت ۲۰ الی ۳۰ دقیقه به آن نور دهید این مدت زمان کاملاً بسته به شدت نور و مهمتر از آن بسته به کیفیت اسپری می باشد. اگر اسپری تازه باشد این مدت حتی می تواند کمتر باشد. چون در زمان خیلی کوتاهی در مجاورت نور عکس العمل نشان می دهد. حالا باید یک ظرف تهیه نمایید تا در آن سود را با آب حل نمایید. مقداری آب که ۱ سانت روی فیبر را بگیرد را داخل ظرف بریزید و حدود ۱۰ تا ۲۰ گرم سود را داخل آن بریزید. زیاد نریزید که کل رنگ پزتیو را خواهد خورد. و به هم بزنید تا کاملاً حل شود. دوباره در یک محل یک کم تاریک فیبر مسی را داخل این محلول سود باندازید.

خیلی آرام آرام محلول را به مدت ۳۰ تا ۶۰ ثانیه تکان دهید. می توانید به جای این کار پنبه ای را به محلول سود سوزآور آغشته کنید و مانند شکل زیر بر روی برد به آرامی بکشید. می بینید که جاهایی که لازم است خالی باشد کم در محلول حل می شود.



شکل ۸-۳ رنگ پزتیو

بعد از این مدت خیلی کوتاه سریعاً فیبر را بیرون آورده و با آب بشوید. حالا لازم است که آن را داخل محلول اسید پرکلرودوفر بیاندازید.



شکل ۸-۴ محلول اسید پرکلرودوفر

پس از اسید کاری برد مدار چاپی جهت عملیات سوراخکاری و مونتاژ آماده است .  
ما برای تهیه مدار چاپی این پروژه از روش چاپ و اتو استفاده کرده ایم که توضیحات و روش آن را در زیر شرح داده می شود.

۸-۲-۵ روش تهیه مدار چاپی با اتو

مواد لازم:

- پرینتر لیزری
- کاغذ گلاسه ی مناسب
- فیبر مسی مدار چاپی
- اتو

## روش کار:

برخلاف پریترهای جوهر افشان که در آن به جای تونر از جوهر مایع استفاده میشود در پریترهای لیزری تونر هنگام عمل چاپ از کارتریج خارج میشود و به صورت یودر بر روی صفحه ی کاغذ منتقل میشود. برای آنکه تصویر یا متنی را به وسیله ی پریترهای لیزری چاپ کنیم تونر که شامل یلاستیک است بر اثر حرارت ذوب میشود و بر روی کاغذ باقی میماند. به دلیل سخت بودن فیبر های مسی نمیتوانیم به صورت مستقیم از پریتر برای ایجاد خط ها و سایر علائم بر روی فیبر مدار چایی استفاده کنیم، بنابراین باید به صورت غیر مستقیم تونر را از پریتر بر روی فیبر مسی انتقال دهیم. یک راه ساده آن است که ابتدا بر روی کاغذی که تونر به نرمی بر روی آن مینشیند مدار را با دقت ۴۰۰ dpi پرینت بگیریم و بعد با دادن حرارت به آن به وسیله ی اتو تونر را برای بار دوم مذاب کنیم و بر روی فیبر مسی انتقال دهیم. این عمل دقیقا شبیه به انتقال تصاویر بر روی T-Shirt با اتو میباشد. نوع کاغذ استفاده شده بسیار مهم است و حتما باید از نوع گلاسه باشد تا تونر را به خود جذب نکند و با اعمال حرارت به راحتی از سطح کاغذ جدا شود و بر روی فیبر مسی منتقل شود. سطح فیبر مسی قبل از انتقال تصویر مدار باید به وسیله ی آب گرم و مایع ظرفشویی و پودر لباسشویی کاملا پاکیزه شود و در انتها آن را خشک کنید.



شکل ۸-۵ فیبر مسی

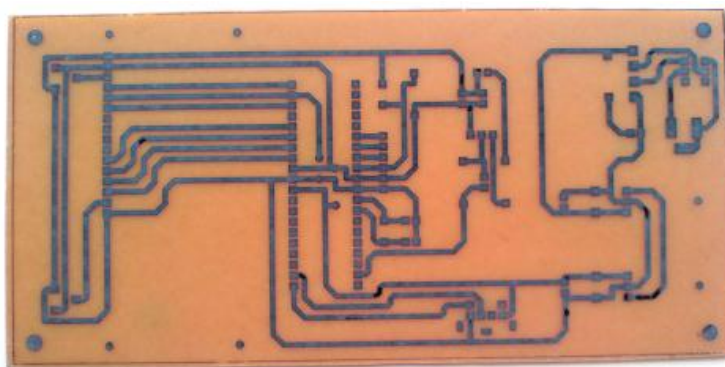
پس از تمیز کردن فیبر مسی توجه کنید که بر روی آن دست نزنید. برای تمیز کردن فیبر مسی همچنین می‌توانید از سیم ظرفشویی استفاده کنید. هنگامی که سطح فیبر کاملاً تمیز و براق شد وقت آن است که کاغذ گلاسه ای را که مدار بر روی آن با دقت ۴۰۰ dpi پرینت گرفته شده را به صورت وارون بر روی فیبر مسی قرار دهیم و ادامه ی کاغذ را در پشت فیبر مسی توسط نوار چسب ثابت کنیم تا کاغذ بر روی فیبر حرکت نکند. حال اتو را با درجه حرارت نسبتاً بالا بر روی آن تا اندازه ای میکشیم که رنگ کاغذ کمی تیره شود و کاغذ حالت چسبیده به فیبر را پیدا کند دقت کنید که اتو باید به صورت یکنواخت به تمامی نواحی گرما برساند این عمل را به صورت پیوسته انجام دهید. توجه داشته باشید که با چندین بار آزمایش و تمرین می‌توانید به بهترین زمان بندی دست پیدا کنید اما زمان مورد نیاز کمتر از ۵ دقیقه در شرایط عادی می‌باشد. فیبر مسی در این زمان دارای حرارت بسیار زیادی است پس هنگام جا به جا کردن آن مراقب باشید. حال فیبر مسی را با همان صورت برای مدتی کمتر از ۱۰ دقیقه در آب داغ قرار دهید و آن را از آب بیرون آورید و قطعه های کاغذ را از روی آن جدا کنید.



شکل ۸-۶ فیبر مدار چابی

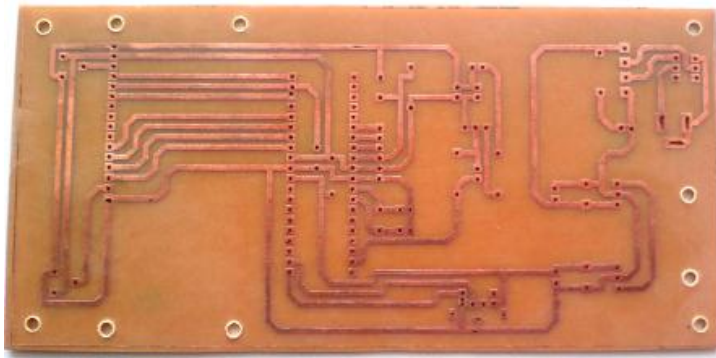
همانطور که می‌بینید مدار به صورت کاملاً دقیق بر روی فیبر مسی منتقل شده است. قدم بعدی از بین بردن قسمت های مسی اضافی از فیبر مدار چابی است.

برای از بین بردن قسمت های مسی اضافی که بدون پوشش عایق هستند از نوعی اسید به نام پرکلرودوفر که در بازار به نام اسید مدار چاپی معروف است استفاده میشود. پرکلرودوفر در بازار به صورت بلور و یا به صورت پودر یافت می شود. روش کار با این نوع اسید به این صورت است که تا اندازه ای به آب گرم از این اسید اضافه میکنیم که محلول به صورت چای پر رنگ در آید در این حالت محلول اسید آماده ی استفاده است. در هنگام کار با این نوع اسید باید توجه داشته باشید که تنها از ظروف غیر فلزی نظیر ظروف پلاستیکی و شیشه ای باید استفاده شود. همچنین توجه داشته باشید که این اسید سمی میباشد و هنگام استفاده از آن باید کاملا مراقب لباس و بدن خود باشید. توصیه میکنم عمل اسیدکاری را در فضای آزاد انجام دهید. پس از ساختن محلول اسید فیبر مدار چاپی را وارد محلول میکنیم و آهسته آن را تکان میدهیم بعد از مدتی شاهد از بین رفتن قسمت های مسی اضافی از کناره میشویم. این عمل را تا از بین رفتن تمامی قسمتهای اضافی انجام میدهیم.



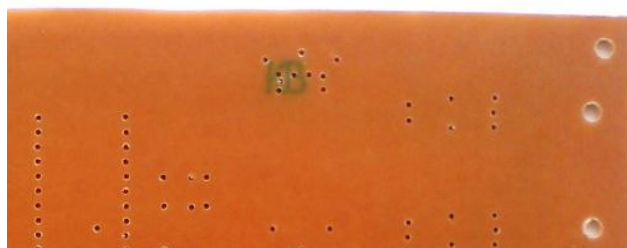
شکل ۷-۸ مدار روی فیبر

پس از از بین رفتن قسمت های اضافی فیبر مسی را از آب سرد شستشو میدهیم و به وسیله ی سیم ظرفشوئی تونرهای به جا مانده روی فیبر را از بین میبریم .



شکل ۸-۸ سوراخ کاری مدار

تنها بخش باقیمانده بخش سوراخکاری است که با دریل انجام می‌دهیم .



شکل ۹-۸ سوراخ کاری مدار

حال فیبر مدار چاپی ما آماده است و میتوانیم قطعات را بر روی آن لحیم کنیم. برای تهیه برد مدار چاپی این پروژه از روش چاپ و اتو استفاده شده است.

### ۳-۸ روش تهیه اسید مدار چاپی

طریق آماده کردن اسید مدار چاپی جهت فرایند اسید کاری برد های مدار چاپی

مواد لازم:

اسید (پرکلرودوفر)

آب

ظرف پلاستیکی

روش کار:

یک ظرف پلاستیکی تهیه نمایید . توجه نمایید که حتما پلاستیکی باشد و نه فلزی. بعد مقداری آب داخل آن بریزید به اندازه ای که حدودا ۱ سانت روی فیبر مسی را بگیرد. در یک لیتر آب مقدار ۱۰۰ تا ۲۰۰ گرم اسید بریزید و آرام ظرف را تکان دهید تا اسید کم کم حل شود چون در یک جا ایجاد گرما می کند و ممکن است که ظرف شما آب شود.

مطمئن شوید که اسید حل شده است. سپس فیبر مسی را که روی آن مدار مربوطه را طراحی کرده اید داخل آب اسید بیاندازید .حدود ۳۰ دقیقه کمتر یا بیشتر که بستگی به مقدار اسید دارد طول خواهد کشید که مسهایی که لازم نیست خورده شود.

در این مدت زمان باید ظرف را آرام آرام تکان دهید تا کل آب اسید روی سطح مسی در حرکت باشد.

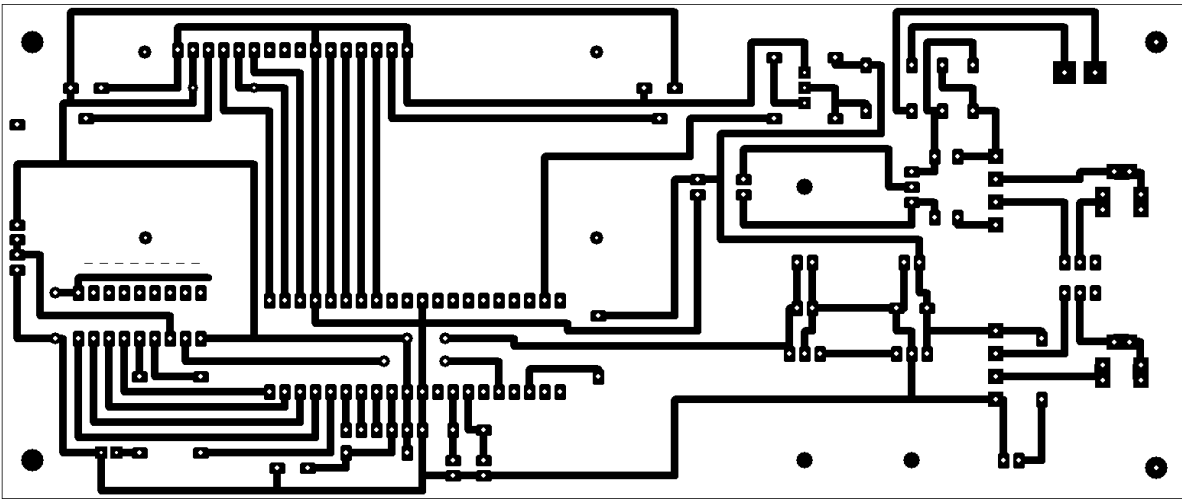
بعد از پاک شدن مس اضافی آن را از آب اسید بیرون آورده و بشوید. بیشتر از این هم داخل اسید نگذارید که قسمتهای مورد نیاز را هم خواهد خورد.

سپس روی آن را با یک سمباده نرم پاک کنید .بعد نقاطی که لازم است را با یک مته ۱ یا نیم میل سوراخکاری کنید. حالا فیبر شما آماده هر نوع بهره برداری است .



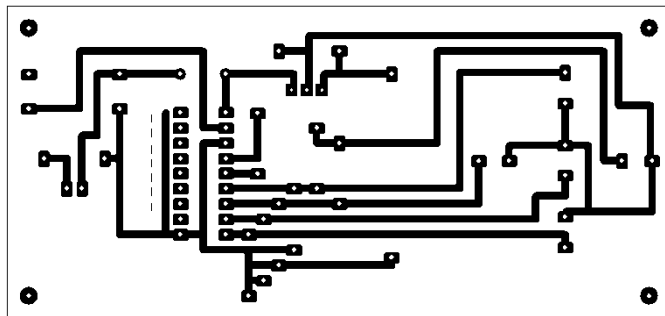
۴-۸ شکل PCB پروژه

شکل PCB برد گیرنده:



شکل ۱۰-۸ PCB برد گیرنده

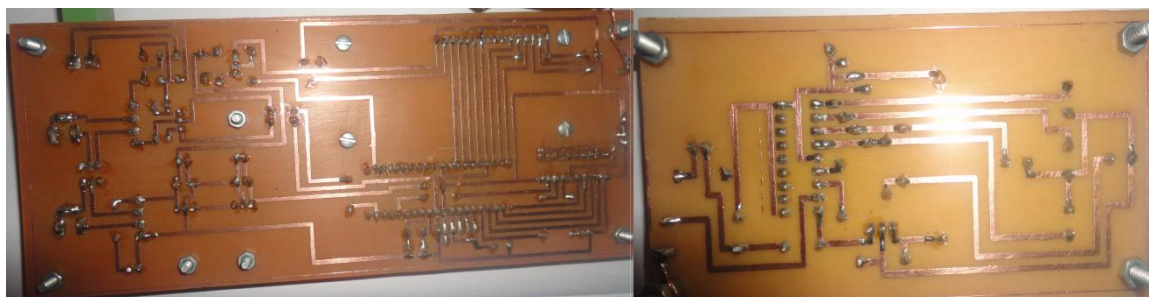
شکل PCB برد فرستنده:



شکل ۱۱-۸ PCB برد فرستنده

## ۵-۸ برد پروژه

چهار گوشه برد ها را سوراخ کردیم تا با پیچ و مهره ، برای آن پایه های کوچک درست کنیم.



شکل ۸-۱۲ برد پروژه

از این کار دو هدف داشتیم:

- جهت زیبایی پروژه
- جهت محافظت از پروژه در برابر برخورد زیر برد با سطوح فلزی. زیرا در اثر برخورد اجسام فلزی با زیر برد احتمال اتصال کوتاه و سوختن قطعات وجود دارد.

\*در این قسمت توضیحات سخت افزاری به پایان میرسد.\*

## فصل نهم

### کامپایلرهای میکروکنترلر AVR

برنامه این پروژه به زبان بیسیک و با کامپایلر Bascom AVR نوشته شده است.

#### ۹-۱ آشنایی با کامپایلرهای میکروکنترلر AVR

برای میکروکنترلر AVR کامپایلرهای متعددی با زبان برنامه نویسی C ، Basic و Pascal

وجود دارد که ما در زیر به تعدادی از آنها اشاره می کنیم:

#### Code Vision ( زبان برنامه نویسی: C )

این کامپایلر یکی از محبوب ترین کامپایلرها می باشد و تنها کامپایلر به زبان C می باشد که

مرجع فارسی دارد. از نظر بهینه بودن کد و سرعت در رده های بالا قرار دارد.

#### Win AVR ( زبان برنامه نویسی: C )

یکی از قوی ترین کامپایلرها به زبان C ، برای AVR می باشد. امروزه خیلی ها به طرف این

کامپایلر جذب شده اند و چند دلیل بر، برتری این کامپایلر نسبت به رقبای خودش وجود دارد:

۱. این کامپایلر Open source می باشد و کاملاً رایگان ارائه می شود.

۲. سرعت بالا در کامپایل.

۳. بهینه بودن کد.

۴. پروگرام کردن مستقیم IC

۵. چیزی که بیشتر از همه قابل توجه است اینه که پشتیبانی خیلی قوی شده و کتابخانه

های نوشته برای این کامپایلر بقدری زیاد است که شما رو بی نیاز به کتابخانه جدید

می کند.

IAR ( زبان برنامه نویسی: C )

این کامپایلر رقیب سر سخت Win AVR می باشد و توسط شرکت اتمل ارائه می شود و

حجمی بسیار بالا دارد و دارای لایسنس ۱ ماهه رایگان می باشد. [www.IAR.com](http://www.IAR.com)

Bascom-AVR ( زبان برنامه نویسی: Basice )

این کامپایلر یک کامپایلر خیلی قوی در زبان بیسیک می باشد و هر کسی که کمی با میکرو

آشنا باشد به سادگی می تواند با این کامپایلر کار کند. از جمله مزیت های این کامپایلر اینه که

دستورات خیلی خیلی ساده دارد و میکرو را مستقیم می تواند پروگرام کند. از جمله معایب

بزرگ این کامپایلر حجم زیاد کد تولید شده می باشد که در بکار گیری، پروژه ای حرفه ای

قابل تامل است.

Mikro AVR ( زبان برنامه نویسی: Basice )

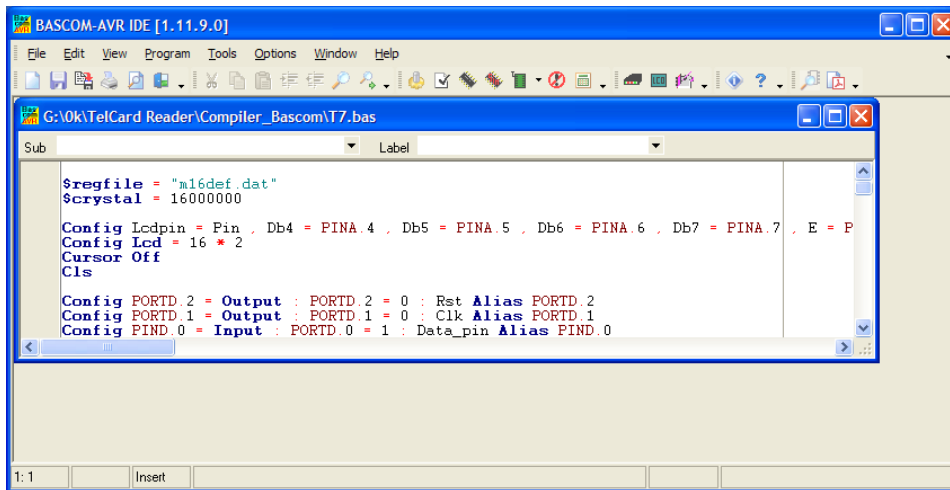
این کامپایلر هم رقیب سرسخت برای Bascom بشمار میرود و کتابخانه های آماده زیادی

دارد.

AVRco\_pascal ( زبان برنامه نویسی Pascal )

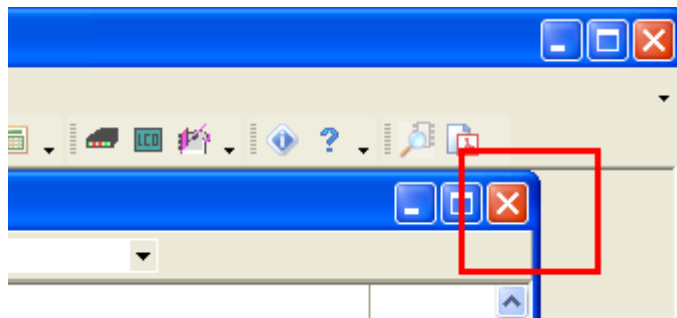
## ۲-۹ محیط کامپایلر Bascom AVR

نمای گرافیکی این کامپایلر از قرار زیر است:



شکل ۱-۹ محیط Bascom AVR

جهت ایجاد یک پروژه جدید باید برنامه قبلی را که به صورت پیش فرض در زمان اجرای نرم افزار بیسکام باز شده است را ببندید.



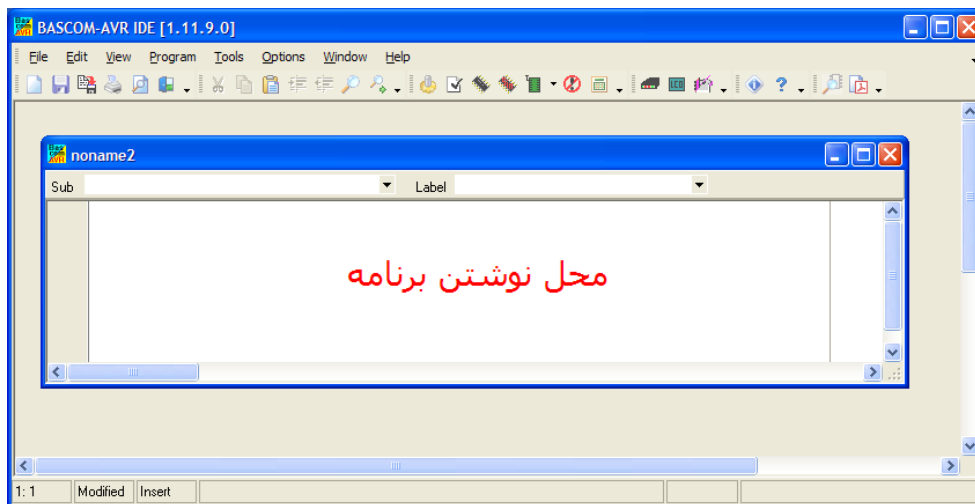
شکل ۲-۹ محیط Bascom AVR

اکنون برای ایجاد یک صفحه جدید از آیکونی که در شکل زیر مشخص شده استفاده کنید:



شکل ۹-۳ ایجاد یک صفحه جدید

پنجره جدیدی باز می شود که خالی است ، و آماده برنامه نویسی می باشد:



شکل ۹-۱ برنامه نویسی

## فصل دهم

### برنامه میکروکنترلر

برنامه این میکروکنترلر با کامپایلر بیسکام انجام شده است.

#### ۱۰-۱ برنامه میکروکنترلر

کدهای برنامه این پروژه از قرار زیر است:

```
$regfile = "m16def.dat"  
$crystal = ۱۱۰۵۹۲۰۰  
  
Config Lcd = ۱۶ * ۲  
Config Lcdpin = Pin , Db۴ = Porta.۴ , Db۵ =  
Porta.۵ , _  
Db۶ = Porta.۶ , Db۷ = Porta.۷ , E = Porta.۲ ,  
Rs = Porta.۰  
Cursor Off  
Cls  
  
Config Timer\ = Pwm , Pwm = ۸ , Compare A Pwm =  
Clear Down , Prescale = ۶۴  
Pwm\A = ۰  
  
Config Portc.\ = Output : Portc.\ = ۰ : Relay  
Alias Portc.\  
  
Config Portb.۰ = Input : Portb.۰ = ۱
```

```

Config Portb.1 = Input : Portb.1 = 1
Config Portb.2 = Input : Portb.2 = 1
Config Portb.3 = Input : Portb.3 = 1
Config Portb.4 = Input : Portb.4 = 1
Config Portb.5 = Output : Portb.5 = 0 : Led
Alias Portb.5

```

```

Dim Pwm_motor As Byte
Dim Pwm_motor_eeprom As Eram Byte

```

```

Gosub Load_of_eeprom
Gosub Start_sub

```

```

Do

```

```

    Led = Pinb.4

```

```

    If Pinb.4 = 1 Then

```

```

        If Pinb.0 = 1 And Pinb.1 = 0 And Pinb.2 =

```

```

0 And Pinb.3 = 0 Then Gosub Up_motor

```

```

        If Pinb.0 = 0 And Pinb.1 = 1 And Pinb.2 =

```

```

0 And Pinb.3 = 0 Then Gosub Right_motor

```

```

        If Pinb.0 = 0 And Pinb.1 = 0 And Pinb.2 =

```

```

1 And Pinb.3 = 0 Then Gosub Stop_motor

```

```

        If Pinb.0 = 0 And Pinb.1 = 0 And Pinb.2 =

```

```

0 And Pinb.3 = 1 Then Gosub Left_motor

```

```

        If Pinb.0 = 1 And Pinb.1 = 1 And Pinb.2 =

```

```

0 And Pinb.3 = 0 Then Gosub Down_motor

```

```

        Gosub Save_to_eeprom

```

```

    End If

```

```

Loop

```

```

End

```

```

'////////////////////////////////////

```

```

End Program

```



```

'*****
Load_of_eeprom:
    Pwm_motor = Pwm_motor_eeprom
Return

'*****
Save_to_eeprom:
    Pwm_motor_eeprom = Pwm_motor
Return

'*****
Start_sub:
    Locate \ , \ : Lcd "Stop Motor"
    Locate ʹ , \ : Lcd "PWM= " ; Pwm_motor
Return

'*****
Up_motor:
    Pwm_motor = Pwm_motor + ʘ
    Pwm\a = Pwm_motor
    Locate ʹ , \ : Lcd "PWM= " ; Pwm_motor ; "
"
Return

'*****
Down_motor:
    Pwm_motor = Pwm_motor - ʘ
    Pwm\a = Pwm_motor
    Locate ʹ , \ : Lcd "PWM= " ; Pwm_motor
Return

'*****
Stop_motor:
    Pwm\a = ʘ
    Locate \ , \ : Lcd "Stop Motor"
Return

```

```

'*****
Left_motor:
    Set Relay
    Pwm\A = Pwm_motor

    Locate ۱ , ۱ : Lcd "Rotate Left"
    Locate ۲ , ۱ : Lcd "PWM= " ; Pwm_motor

Return

'*****
Right_motor:
    Reset Relay
    Pwm\A = Pwm_motor

    Locate ۱ , ۱ : Lcd "Rotate Right"
    Locate ۲ , ۱ : Lcd "PWM= " ; Pwm_motor

Return

```

### ۱-۱-۱۰ توضیحات برنامه میکروکنترلر

```
$regfile = "m16def.dat"
```

پیکره بندی میکروکنترلر Atmega۱۶

```
$crystal = ۱۱۰۵۹۲۰۰
```

معرفی فرکانس کاری میکروکنترلر

```
Config Lcd = ۱۶ * ۲
```

تعیین نوع LCD استفاده شده. LCD استفاده شده در این پروژه دارای ۱۶ ستون و ۲ سطر می باشد.

```

Config Lcdpin = Pin , Db۴ = Porta.۴ , Db۵ =
Porta.۵ , _
Db۶ = Porta.۶ , Db۷ = Porta.۷ , E = Porta.۲ , Rs = Porta.۰

```

تعریف پایه LCD که به میکروکنترلر متصل شده اند.

Cursor Off

تعیین حالت مکا نمای LCD . با این دستور مکان نما چشمک زن خواهد بود.

Cls

پاک کردن کل صفحه نمایش LCD

Config Timer<sub>1</sub> = Pwm , Pwm = ۸ , Compare A Pwm = Clear Down ,

Prescale = ۶۴

پیکره بندی TIMER<sub>1</sub> به عنوان PWM

Pwm<sub>1a</sub> = ۰

مقدار ۰ برای PWM باعث می شود که خروجی تولید پالس در سطح منطقی ۰ قرار بگیرد. خروجی پالس PWM از طریق پایه OC<sub>1A</sub> قابل دسترسی است. این پایه در میکروکنترلر Atmega<sub>16</sub> پایه شماره ۱۹ میباشد.

Config Portc.<sub>1</sub> = Output : Portc.<sub>1</sub> = ۰ : Relay Alias Portc.<sub>1</sub>

PORTC.<sub>1</sub> به عنوان خروجی پیکره بندی شده.

PORTC.<sub>1</sub> را صفر میکنیم

با دستور Alias به PORTC.<sub>1</sub> برجسبی به نام Relay می زنیم. هر کجا نیاز به

PORTC.<sub>1</sub> داشتیم می توانیم از Relay استفاده کنیم.

Config Portb.<sub>۰</sub> = Input : Portb.<sub>۰</sub> = ۱

PORTB.<sub>۰</sub> به عنوان ورودی پیکره بندی شده.

PORTB.<sub>۰</sub> را با مقدار ۱ به سطح ولتاژ VCC می رسانیم. با این کار در داخل میکروکنترلر

به صورت سخت افزاری یک عدد مقاومت  $10K$  اهم که یک سر آن به  $VCC$  متصل است به این پایه متصل می شود. به اصلاح میگوییم  $PULL\ UP$  شده است.

**Config Portb.۱ = Input : Portb.۱ = ۱**

۱.  $PORTB$  به عنوان ورودی پیکره بندی شده.

۱.  $PORTB$  را با مقدار ۱ به سطح ولتاژ  $VCC$  می رسانیم. با این کار در داخل میکروکنترلر به صورت سخت افزاری یک عدد مقاومت  $10K$  اهم که یک سر آن به  $VCC$  متصل است به این پایه متصل می شود. به اصلاح میگوییم  $PULL\ UP$  شده است.

**Config Portb.۲ = Input : Portb.۲ = ۱**

۲.  $PORTB$  به عنوان ورودی پیکره بندی شده.

۲.  $PORTB$  را با مقدار ۱ به سطح ولتاژ  $VCC$  می رسانیم. با این کار در داخل میکروکنترلر به صورت سخت افزاری یک عدد مقاومت  $10K$  اهم که یک سر آن به  $VCC$  متصل است به این پایه متصل می شود. به اصلاح میگوییم  $PULL\ UP$  شده است.

**Config Portb.۳ = Input : Portb.۳ = ۱**

۳.  $PORTB$  به عنوان ورودی پیکره بندی شده.

۳.  $PORTB$  را با مقدار ۱ به سطح ولتاژ  $VCC$  می رسانیم. با این کار در داخل میکروکنترلر به صورت سخت افزاری یک عدد مقاومت  $10K$  اهم که یک سر آن به  $VCC$  متصل است به این پایه متصل می شود. به اصلاح میگوییم  $PULL\ UP$  شده است.

## Config Portb.۴ = Input : Portb.۴ = ۱

۴. PORTB به عنوان ورودی پیکره بندی شده.

۴. PORTB را با مقدار ۱ به سطح ولتاژ VCC می‌رسانیم. با این کار در داخل میکروکنترلر به صورت سخت افزاری یک عدد مقاومت ۱۰K اهم که یک سر آن به VCC متصل است به این پایه متصل می‌شود. به اصلاح می‌گوییم PULL UP شده است.

## Config Portb.۵ = Output : Portb.۵ = ۰ : Led Alias Portb.۵

۵. PORTB به عنوان خروجی پیکره بندی شده.

۵. PORTB را صفر می‌کنیم

با دستور Alias به PINC.۱ برچسبی به نام LED می‌زنیم. هر کجا نیاز به ۱. PORTC داشتیم می‌توانیم از LED استفاده کنیم.

## Dim Pwm\_motor As Byte

تعریف متغیر مورد استفاده در برنامه

مقدار PWM که تنظیم میشود در این متغیر ذخیره میشود.

این متغیر از جنس حافظه SRAM است. سرعت دسترسی به این نوع حافظه ها بالاست ولی اطلاعات ذخیره شده در آن با قطع تغذیه پاک میشود.

## Dim Pwm\_motor\_eeprom As Eram Byte

تعریف متغیر مورد استفاده در برنامه

این متغیر از جنس حافظه EEPROM است. سرعت دسترسی به این نوع حافظه ها کند است، ولی اطلاعات ذخیره شده در آن با قطع تغذیه پاک نمیشود.

قبل از توضیح ادامه برنامه یاد آوری میکنیم که برنامه این پروژه با ساب روتین نوشته شده است.

#### سابروتین چیست:

در گذشته جهت جابجایی در قسمت های مختلف برنامه از دستورات goto و jump استفاده می کردند. این کار باعث پیچیده شدن برنامه و ناخوانایی آن می شد. برای رفع این مشکل ساختار زیربرنامه و تابع ایجاد شد. با این روش یک برنامه تشکیل شده از یک Main که هسته اصلی برنامه است ، و تعداد زیادی از زیرتابع هر زیرتابع فقط یک کار انجام میدهد و بعد از اتمام کار ، برنامه به همان جایی که فراخوانی شده بود بر میگردد. استفاده از زیرتابع ها در برنامه نویسی را برنامه نویسی سابروتین می گویند.

از یک زیر تابع می توان برای برنامه های دیگر نیز استفاده کرد. خوانایی برنامه افزایش می یابد. توجه داشته باشید که بعد از استفاده از زیرتابع ، در آخر باید از دستور Return استفاده کنید تا برنامه به جای اول خود برگردد. در این برنامه از زیرتابع های زیر استفاده شده است:

- Load\_of\_eeprom
- Save\_to\_eeprom
- Start\_sub
- Up\_motor
- Down\_motor

- Stop\_motor
- Left\_motor
- Right\_motor

وظایف این زیر تابع ها از قرار زیر است:

- Load\_of\_eeprom

اطلاعات ذخیره شده در حافظه EEPROM را خوانده و در متغیر SRAM قرار میدهد.

از آنجایی که سرعت دسترسی با حافظه های EEPROM خیلی کند است ، CPU نیاز به یک

منبع ذخیره با سرعت بالاست. برای همین نیاز داریم تا اطلاعات خوانده شده از EEPROM

را در SRAM ذخیره کنیم.

- Save\_to\_eeprom

اطلاعات حافظه SRAM را در حافظه های دائمی EEPROM ذخیره میکند.

- Start\_sub

نمایش Menu شروع بر روی LCD

- Up\_motor

افزایش مقدار PWM جهت افزایش سرعت گردش موتور

- Down\_motor

کاهش مقدار PWM جهت کاهش سرعت گردش موتور

- Stop\_motor

با قرار دادن مقدار ۰ در PWM موتور را از حرکت نگه می دارد.

- Left\_motor

با تغییر وضعیت رله باعث تغییر در جهت گردش موتور می شود.

- Right\_motor

با تغییر وضعیت رله باعث تغییر در جهت گردش موتور می شود.

Gosub Load\_of\_eeprom

Gosub Start\_sub

Do

Led = Pinb.۴

If Pinb.۴ = ۱ Then

If Pinb.۰ = ۱ And Pinb.۱ = ۰ And Pinb.۲ =

۰ And Pinb.۳ = ۰ Then Gosub Up\_motor

If Pinb.۰ = ۰ And Pinb.۱ = ۱ And Pinb.۲ =

۰ And Pinb.۳ = ۰ Then Gosub Right\_motor

If Pinb.۰ = ۰ And Pinb.۱ = ۰ And Pinb.۲ =

۱ And Pinb.۳ = ۰ Then Gosub Stop\_motor

If Pinb.۰ = ۰ And Pinb.۱ = ۰ And Pinb.۲ =

۰ And Pinb.۳ = ۱ Then Gosub Left\_motor

If Pinb.۰ = ۱ And Pinb.۱ = ۱ And Pinb.۲ =

۰ And Pinb.۳ = ۰ Then Gosub Down\_motor

Gosub Save\_to\_eeprom

End If

Loop



```

End
'////////////////////////////////////
End Program

'*****

Load_of_eeprom:
    Pwm_motor = Pwm_motor_eeprom
Return

'*****

Save_to_eeprom:
    Pwm_motor_eeprom = Pwm_motor
Return

'*****

Start_sub:
    Locate 1 , 1 : Lcd "Stop Motor"

    Locate 2 , 1 : Lcd "PWM= " ; Pwm_motor

Return

'*****

Up_motor:
    Pwm_motor = Pwm_motor + 5

    Pwm\|a = Pwm_motor

    Locate 2 , 1 : Lcd "PWM= " ; Pwm_motor ; "

```

```

"
Return

'*****
Down_motor:
    Pwm_motor = Pwm_motor - 5

    Pwm\A = Pwm_motor

    Locate 2 , 1 : Lcd "PWM= " ; Pwm_motor

Return

'*****
Stop_motor:
    Pwm\A = 0

    Locate 1 , 1 : Lcd "Stop Motor"

Return

'*****
Left_motor:
    Set Relay

    Pwm\A = Pwm_motor

    Locate 1 , 1 : Lcd "Rotate Left"

    Locate 2 , 1 : Lcd "PWM= " ; Pwm_motor

Return

```

```
*****
```

```
Right_motor:
```

```
Reset Relay
```

```
Pwm\A = Pwm_motor
```

```
Locate \ , \ : Lcd "Rotate Right"
```

```
Locate 2 , \ : Lcd "PWM= " ; Pwm_motor
```

```
Return
```

## منابع:

[۱] کاهه، علی، کتاب میکروکنترلرهای AVR

[۲] [www.avr-electronic.ir](http://www.avr-electronic.ir)

[۳] [www.picpars.com](http://www.picpars.com)

[۴] [www.techno-electro.com](http://www.techno-electro.com)