

بنام خدا

## پروژه تشخیص رنگ سنگو

پروژه تشخیص رنگ سنگو



### مقدمه ای بر نور و رنگ

«رنگ» یکی از نعمت های خداوند است که به انسان ها کمک می کند که توسط آن بتوانند بینایی بهتری روی محیط پیرامون خود داشته باشند. در علم الکترونیک، تشخیص رنگ توسط سنسور های مخصوصی که برای این منظور تولید می شوند انجام می شود. تشخیص رنگ بصورت الکترونیکی کاربرد بسیار زیادی در صنایع مختلف بخصوص صنایع بسته بندی و چاپ دارد.

نوری که چشم انسان آنرا حس می کند ناحیه ای باریکی از انرژی الکترومغناطیسی را تشکیل می دهد. انرژی الکترومغناطیسی در شکلی از امواج منتقل می شود که بوسیله دامنه و فرکانس توصیف می گردد. معمولا نور را بوسیله طول موج در واحد نانومتر بیان می کنند. وقتی که از نور می گوئیم منظورمان همان تابشی است که با چشم می بینیم، پس تابش فرابنفش و فرورسرخ را نباید به عنوان نور دانست زیرا با چشم قابل رؤیت نیستند. بازه طول موجهایی که با چشم قابل درک است از حدود ۴۰۰ نانومتر تا ۷۰۰ نانومتر است. با تاباندن نور سفید به یک منشور می توانیم طول موجهای قابل رویت را داشته باشیم. فرایند دیدن زمانی رخ می دهد که نور از اشیاء منعکس و به چشم ما برسد.

### تشخیص رنگها با ابزار الکترونیکی

از زمان آشکارسازی طیف رنگها توسط گاليله به وسیله منشور و بعد از آن اختراع کامپیوتر، استفاده رنگها گسترش بسیاری پیدا کرد. بنابراین وسائل ساخته شدند که بتوانند رنگها را تشخیص دهند. اساس کار این المانها بر اساس بازتابش نور از سطح مواد می باشد. می دانیم که اجسام یک سری از طول موجهای نور را دریافت و یکسری دیگر را بازتابش می کند. طول موج بازتابش شده که به چشم ما می رسد رنگ آن ماده می باشد. آشکار ساز رنگها که به سنسور رنگ معروفند برای آشکار سازی رنگ از این خاصیت استفاده می کنند.

سنسور رنگ به این صورت عمل می کند که امواج ساطع شده از مواد را دریافت و آشکار می کند آن را به یکی از راههای جریان، ولتاژ و یا فرکانس اعلام می کند.

سنسورهای متفاوتی تا به حال ساخته شده است که بر حسب نیاز کاری مورد استفاده قرار می گیرند. سنسورهایی وجود دارد که یکی از سه رنگ اصلی قرمز، سبز و آبی را تشخیص می دهد و با استفاده از این سنسورها در یک مدار می توان مداری را طراحی کرد که تعداد زیادی از رنگها را تشخیص دهد.

البته سنسورهایی طراحی شده است که می تواند تعداد زیادی از رنگها را تشخیص دهد که به سنسورهای RGB معروفند. دو نوع از این سنسورها که در بازار موجود می باشند PD153 و TCS230 می باشد که خصوصیات هر یک را بیان می کنیم.

## سنسور رنگ PD153

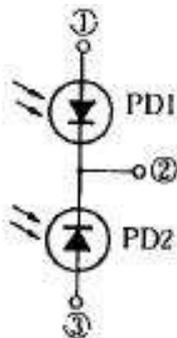
در دیتاشیت مشخصات این سنسور به صورت زیر می باشد:

- ۱- در طول موجهای زیاد دارای حساسیت زیاد می باشد.
- ۲- این سنسور تا ۱۸ رنگ را می تواند تشخیص دهد.
- ۳- خروجی این سنسور مطابق با ورودی تغییر می کند.
- ۴- این سنسور دارای اطمینان برای کارکرد صحیح برای اهداف مخصوص می باشد.
- ۵- این سنسور توانایی اندازه گیری نور آبی در نزدیکی مادون قرمز را دارا می باشد. (۱۰۰۰ نانومتر طول موج)

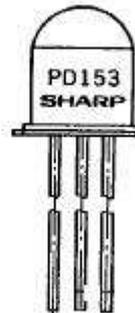
## مشخصات ظاهری سنسور رنگ PD153

این سنسور دارای سه پایه می باشد که در شکل ۱ نشان داده شده است و مدار داخلی آن را به صورت شکل ۲

مدل می کند.



شکل ۲



شکل ۱

مدار داخلی این سنسور را به صورت دو دیود نوری که کاتود آنها به هم اتصال دارد مدل می کنند که پایه ها را آنود یک، آنود دو و کاتود می نامند.

## کاربردهای این سنسور

- ۱- خواندن رنگهای رایج بر روی پولهایی که مبادله می شوند.
- ۲- بازخوانی دمای رنگ و طول موج آن.
- ۳- این سنسور می تواند رنگ اشیاء را تشخیص دهد.
- ۴- این سنسور توانایی تشخیص رنگ شیء از طریق بازتابش آینه را نیز دارد.

خروجی سنسور سیگنال آنالوگ می باشد که باید تقویت شود و سپس برای تجزیه و تحلیل به مدار بعدی (مثلاً یک میکرو) اعمال شود.

## سنسور تشخیص رنگ TCS230

تعدادی از توانایی این سنسور به صورت زیر می باشد:

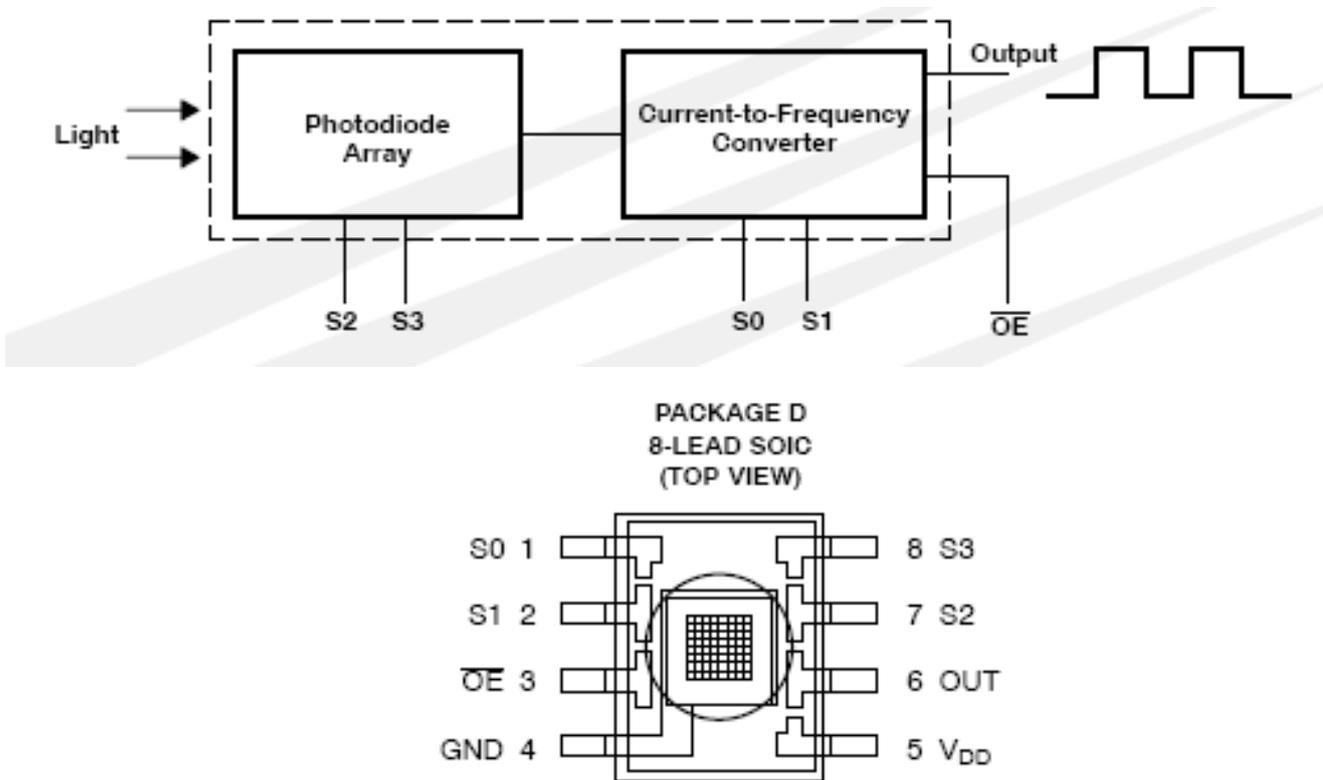
- ۱- توانایی تبدیل شدت نور به فرکانس
- ۲- توانایی پردازش رنگ و تطابق کامل آن با فرکانس خروجی
- ۳- حفاظت شده در مقابل نوسانات برق و ایجاد مشکل
- ۴- توانایی تحمل ضریب حرارت تا 200 ppm

### توضیحات

TCS230 رنگ نور را پردازش می کند و آنرا به فرکانس تبدیل می کند. خروجی این سنسور یک موج مربعی با دیوتی سایکل ۵۰٪ است و مقدار فرکانس خروجی مطابق با شدت نور است. فرکانس خروجی را می توان بوسیله دو پایه ورودی در سه حالت از قبل تنظیم کرد. خروجی این سنسور را می توان مستقیماً به یک میکرو کنترلر و یا یک مدار دیجیتال وصل کرد. ۱۶ فتودیود فیلترهای آبی، ۱۶ فتودیود فیلترهای سبز، ۱۶ فتودیود فیلترهای قرمز و ۱۶ فتودیود بدون فیلتر می باشد. این ۴ نوع فتودیود به هم پیوسته هستند و غیر یکنواختی تابش های واقع شده را به حداقل می رسانند. همه دیودهای با رنگهای یکسان به طور موازی به هم متصل هستند و نوع فتودیودها توسط کاربر، توسط پین های ورودی تعیین می شود.

## مشخصات بلوک دیاگرام

این سنسور به صورت نصب سطحی می باشد که دارای ۸ پایه است و شکل پایه های آن به صورت زیر می باشد.



پایه های سنسور طبق جدول زیر می باشد:

نام	شماره پایه	I/O	توضیحات
زمین	۴		این پایه به زمین منبع وصل می شود
OE	۳	I	فعال کننده فرکانس خروجی
Out	۶	O	فرکانس خروجی
S0,S1	۱/۲	I	تعیین مقیاس فرکانس خروجی
S2,S3	۷/۸	I	انتخاب نوع فتودیود در ورودی
VDD	۵		تغذیه

با تغییر **S0** و **S1** می توان محدوده فرکانس خروجی را تغییر داد.

با تغییر **S2** و **S3** می توان نوع فتودیودها را انتخاب کرد.

S <sub>2</sub>	S <sub>3</sub>	نوع فتودیودها
L	L	قرمز
L	H	آبی
H	L	بدون فیلتر
H	H	سبز

محدوده فرکانسی در هر یک از حالت های **S2** و **S3** به صورت زیر می باشد:

واحد	محدوده فرکانسی	شرایط پایه ها	پارامترها
KHz	۵۰۰-۶۰۰	<b>S<sub>0</sub>=H , S<sub>1</sub>=H</b>	مقیاس کامل فرکانس
KHz	۱۰۰-۱۲۰	<b>S<sub>0</sub>=H , S<sub>1</sub>=L</b>	
KHz	۱۰-۱۲	<b>S<sub>0</sub>=L , S<sub>1</sub>=H</b>	

## اطلاعات برای استفاده از سنسور

۱- خط منبع تغذیه:

خط منبع تغذیه به وسیله یک خازن ۰/۱ میکرو یا ۰/۰۱ میکرو به **VDD** وصل می شود. بین پایه های **OE** و زمین یک مقاومت کم اهم برای بهبود و کاهش نویز قرار می گیرد.

۲- وجه مشترک خروجی:

در فاصله های کم بین خروجی سنسور و طبقه بعد می توان خروجی را مستقیماً به طبقه بعد وصل کرد. در فاصله های بیش از ۱۲ اینچ استفاده از یک بافر یا راه انداز توصیه می شود. وقتی پایه **OE** برابر با یک است امپدانس خروجی بالا می رود و می توان آنرا به چندین میکرو اتصال داد (یا طبقات مختلف دیگر).

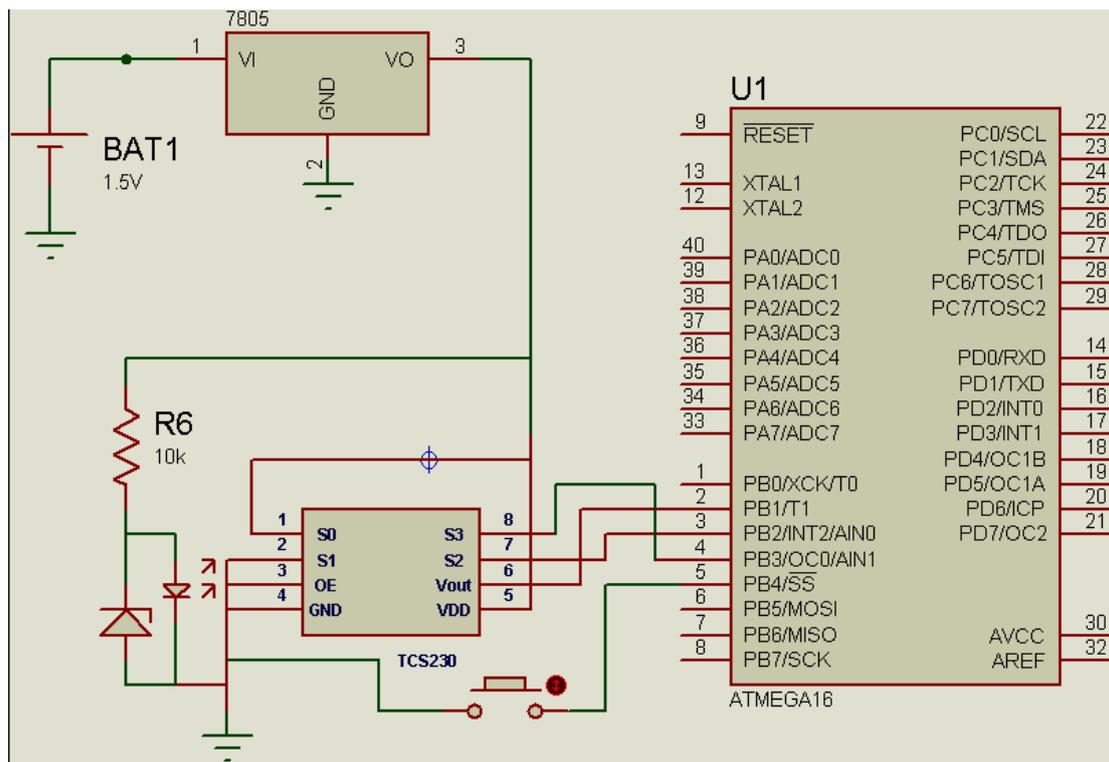
اگر S0 و S1 در وضعیت L/L باشند سنسور غیرفعال می شود و خروجی در وضعیت صفر می باشد و سنسور در وضعیتهای مختلف S2 و S3 نمی تواند فعال شود.

### ۳- انتخاب نوع فتودیود:

نوع فتودیودها برای استفاده توسط دو ورودی دیجیتال S2 و S3 کنترل می شود.

## ارتباط بین میکروکنترلر و سنسور

در این وسیله پایه های S2 و S3 و Vout سنسور به میکرو متصل می شود با صفر و یک کردن پایه های S2 و S3 در سه حالت می توان مقدار فرکانس سه رنگ اصلی قرمز ، سبز و آبی هر رنگی را در خروجی سنسور که پایه V-out می باشد تولید کرد. پایه های S2 و S3 به میکرو متصل می شود و عمل صفر و یک کردن توسط میکرو انجام می شود. فرکانسی که در پایه V-out تولید می شود نیز به میکرو اعمال می شود تا پردازش های لازم روی آن صورت گیرد.



طریقه ارتباط سنسور با میکرو

## تراشهای ضبط صدا و وسائل پخش با مدت زمان ۶۰ ثانیه (APR9600)

خصوصیات:

۱- تراشه با کیفیت ضبط و پخش عالی می باشد.

بدون نیاز به IC های خارجی

حداقل اجزای خارجی

۲- دارای تکنولوژی حافظه فلش می باشد که می تواند اطلاعات را حتی در قطع برق حفظ کند.

بدون نیاز به IC های پشتیبان

۳- دارای پایه های قابل تنظیم برای پیغام قابل انتخاب توسط کاربر

دسترسی تصادفی به چندین مدت زمان ثابت

دسترسی دائم از چندین زمان غیر دائم پیغام ها

۴- کاربرپسند، استفاده و عملکرد آسان

نیاز به برنامه ریزی و پیشرفت سیستم ندارد

ثبت سطح فعالیت و لبه فعالیت پین های پخش

۵- مصرف پایین

جریان در حال کار ۲۵ میلی آمپر می باشد

جریان حالت Standby یک میکروآمپر می باشد

خاموشی اتوماتیک

۶- دارای تراشه ای برای فعال کردن پایه برای توسعه پیام های ساده می باشد

توضیحات تکمیلی

**APR9600** تراشه یکپارچه برای ضبط می باشد و می تواند اطلاعات را ذخیره و حتی در زمان قطع

برق آنرا حفظ کند و توانایی ضبط صدا تا ۶۰ ثانیه را دارا می باشد. این تراشه هر دو حالت دسترسی تصادفی و

ترتیبی را از چندین پیغام دارا می باشد. آزمایش آن توسط کاربر آسان می باشد و این امر به طراحان اجازه می

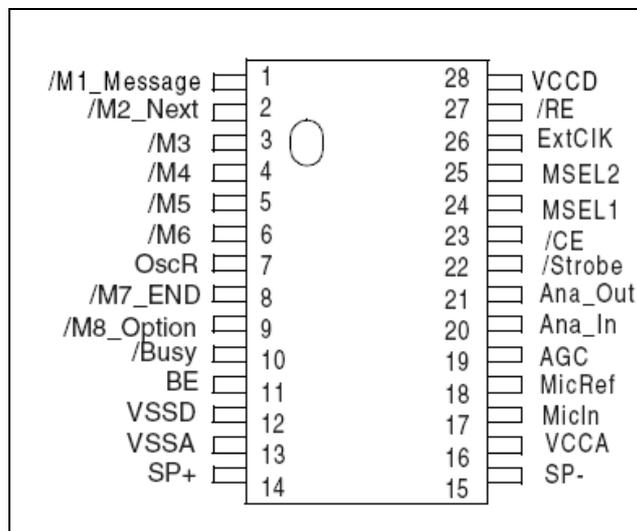
دهد طرحشان را برای کیفیتی عالی و ذخیره زمان بر حسب نیاز بهبود دهند. این تراشه دارای تقویت کننده

یکپارچه خروجی، تقویت کننده میکروفون و طراحی سیستم MPLife مدارهای کنترل گین اتوماتیک می باشد.

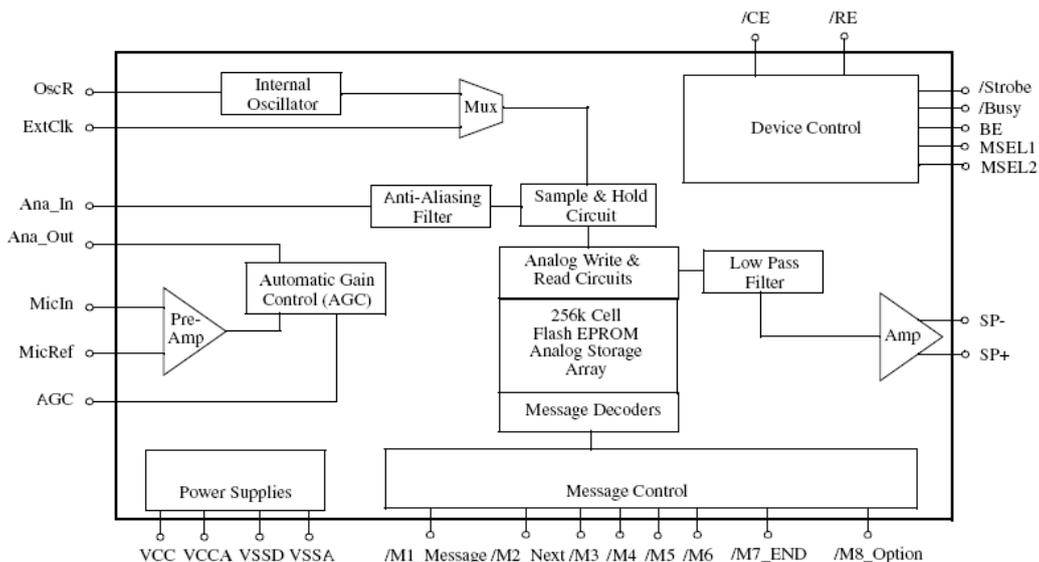
این تراشه برای استفاده در ضبط کننده های قابل حمل، اسباب بازی ها، کاربردهای صنعتی و مصارف دیگر مناسب می باشد.

این تراشه دارای تکنولوژی می باشد که در آن با داشتن حافظه غیرفعال و سطح مصرفی اختصاصی انرژی که در آن اطلاعات در چندین سطح حافظه پس از پردازش ذخیره می شوند که در حافظه پیل می تواند تا ۲۵۶ ولت را ذخیره کند. این تکنولوژی تراشه را قادر می سازد که سیگنال ذخیره شده را به شکل طبیعی بازسازی کند و نیاز به رمزگذاری و تراکم اطلاعات که اغلب باعث اوجاج می شود را برطرف کند.

شکل آی سی و پایه های آن به صورت زیر می باشد.



بلوک دیاگرام مدار داخلی IC به صورت زیر می باشد:



## مدیریت پیغام ها

عملیات ضبط و پخش به وسیله مدارهای تراشه کنترل می شود. چندین ساختار سودمند، وابسته به عملکرد مطلوب تراشه وجود دارد این ساختار پیغام ها شیوه مدیریت پیام های طولانی و شمارش قسمتهای خارجی را تعیین می کنند. سپس طراح قبل از طرح باید نوع ساختار را طرح کند. نوع ساختار تاثیر روی کیفیت صدا ندارد. برای اطلاعات در مورد کیفیت صدا به میزان نمونه گیری قسمت کیفیت صدا می تواند به اطلاعات موثر سازنده رجوع کنید.

این وسیله با سه بین حالتهای مدیریت پیغام را پشتیبانی می کند که عبارتند از: پایه های MSEL1 ، MSEL2 و M8 که این پینها را در شکل بالا نشان داده شده است.

ساختار	MSEL1	MSEL2	M8- Option
دسترسی تصادفی دو مدت ثابت پیغام ها	0	0	وصل این پین به VCC با یک مقاومت 100k
دسترسی تصادفی چهار مدت ثابت پیغام ها	1	0	وصل این پین به VCC با یک مقاومت 100k
دسترسی تصادفی هشت مدت ثابت پیغام ها	1	1	با پین ورودی راه اندازی می شود
روش ضبط و عمل کردن زمان	0	0	0
روش ضبط، عملکرد اتوماتیک پخش صدا	0	0	1

این IC را در سه ساختار می توان استفاده کرد که عبارتند از:

۱- ساختار تنظیم معمولی

۲- ساختار برگرداندن اتوماتیک

۳- ساختار دسترسی تصادفی

که برحسب نیاز طراحی می توان هر یک از این ساختار را استفاده کرد.

## ساختار تنظیم معمولی:

در این ساختار می توان با تغییر پایه های ۱ و ۲ پیامها را پشت سر هم ضبط یا پخش کرد

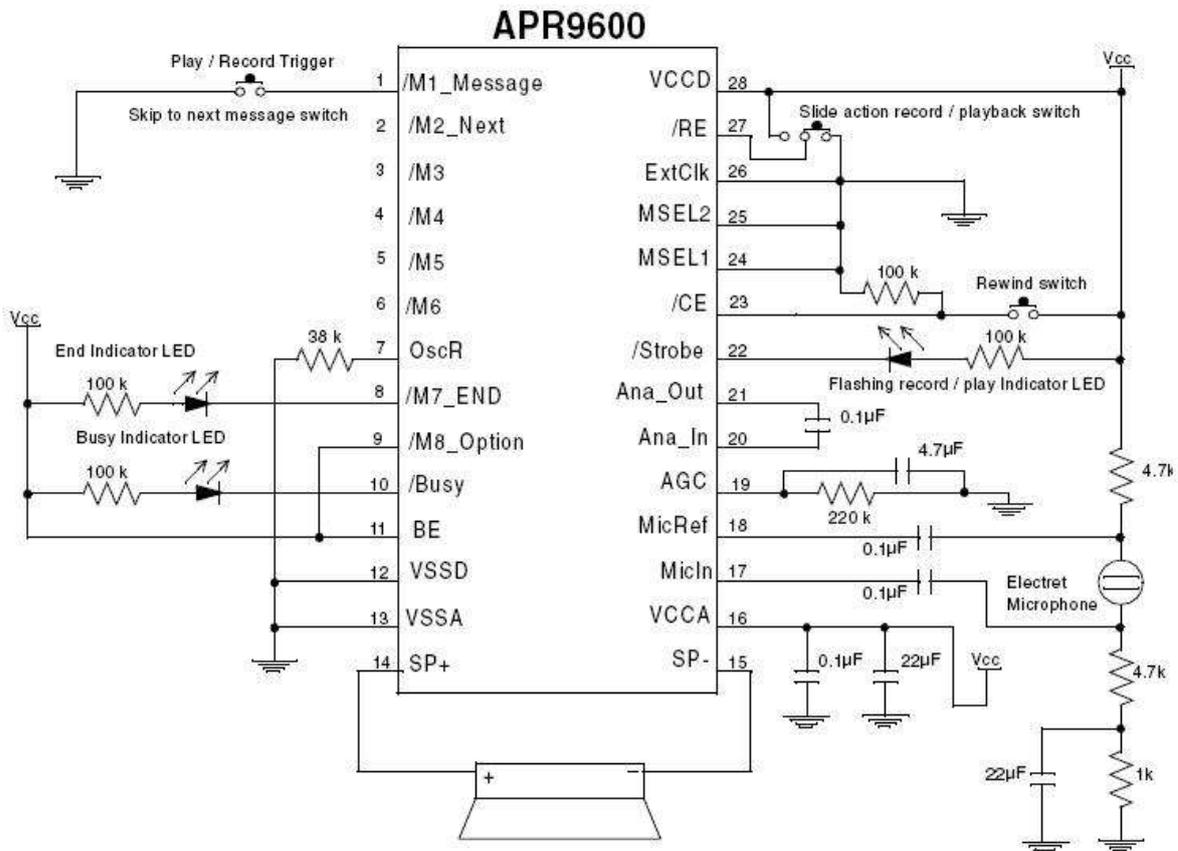
### ضبط پیام:

برای ضبط کردن پیام پایه ۲۷ را به زمین وصل می کنیم و سپس پایه ۱ را ۰ می کنیم و IC شروع به ضبط صدا می کند و با دادن پالس های منفی به پایه ۲ پیام های متعددی ضبط می شود.

### پخش پیام:

برای پخش کردن پیام پایه ۲۷ را به خط تغذیه وصل می کنیم و سپس پایه ۱ را ۰ می کنیم و IC شروع به پخش صدا می کند و با دادن پالس منفی به پایه ۲ می توان پیام های ضبط شده را به ترتیب پخش کرد.

مدار ساختار تنظیم معمولی به صورت زیر می باشد:



## ساختار برگردان اتوماتیک:

در این ساختار می توان یک پیام را در IC ضبط یا پخش کرد.

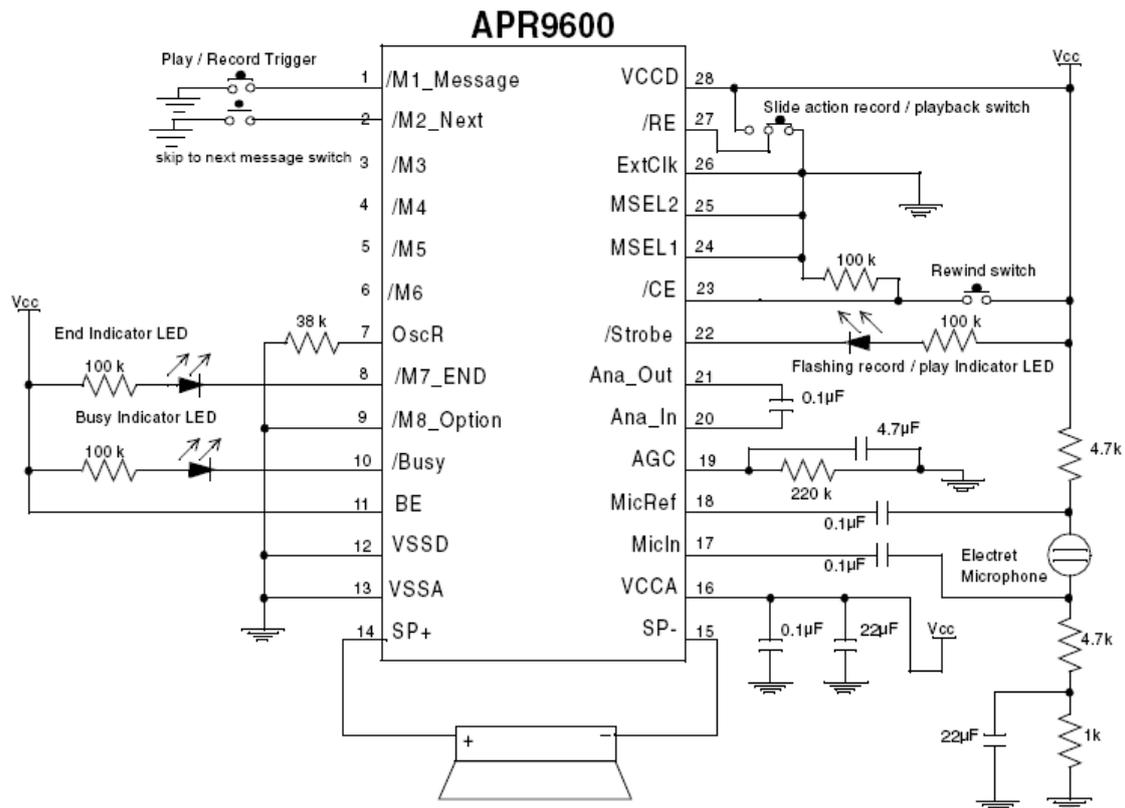
ضبط پیام:

برای ضبط پیام در این ساختار پایه ۲۷ را به زمین و با دادن یک پالس با لبه پایین رونده به پایه ۱ می توان در IC ضبط کرد.

پخش پیام:

برای پخش پیام پایه ۲۷ را به خط تغذیه وصل می کنیم و با اعمال یک پالس پایین رونده به پایه ۱ می توان پیام ضبط شده را پخش کرد.

مدار ساختار برگرداندن اتوماتیک به صورت زیر می باشد:



## ساختار دسترسی تصادفی:

در این ساختار با توجه به تغییر بین های ۹، ۲۴ و ۲۵ که در جدول X مشخص شده است ۲ یا ۴ یا ۸ پیام را ضبط کرد.

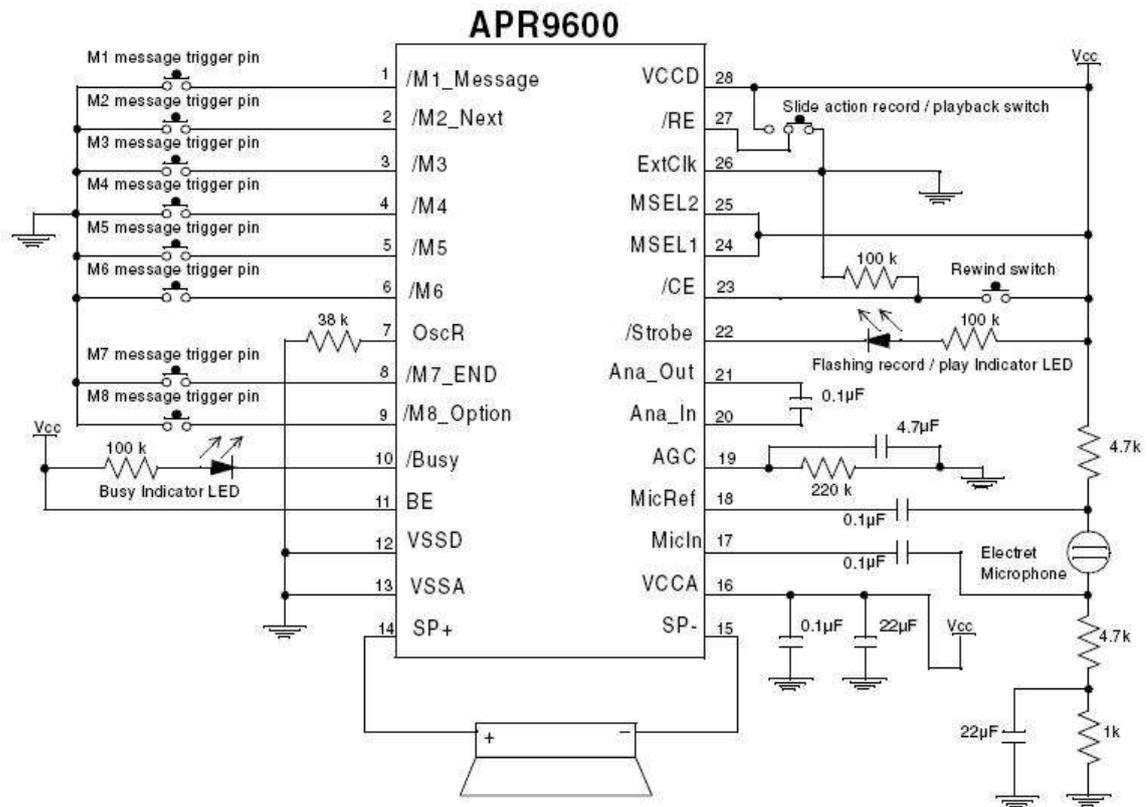
### ضبط پیام:

در این ساختار روی پایه های ۱ تا ۸ می توان صدا ضبط کرد. به این صورت که پایه ۲۷ را به زمین وصل می کنیم و با دادن یک پالس سطح پایین به هر کدام از پایه ها در آن پایه می توان پیام را ضبط کرد.

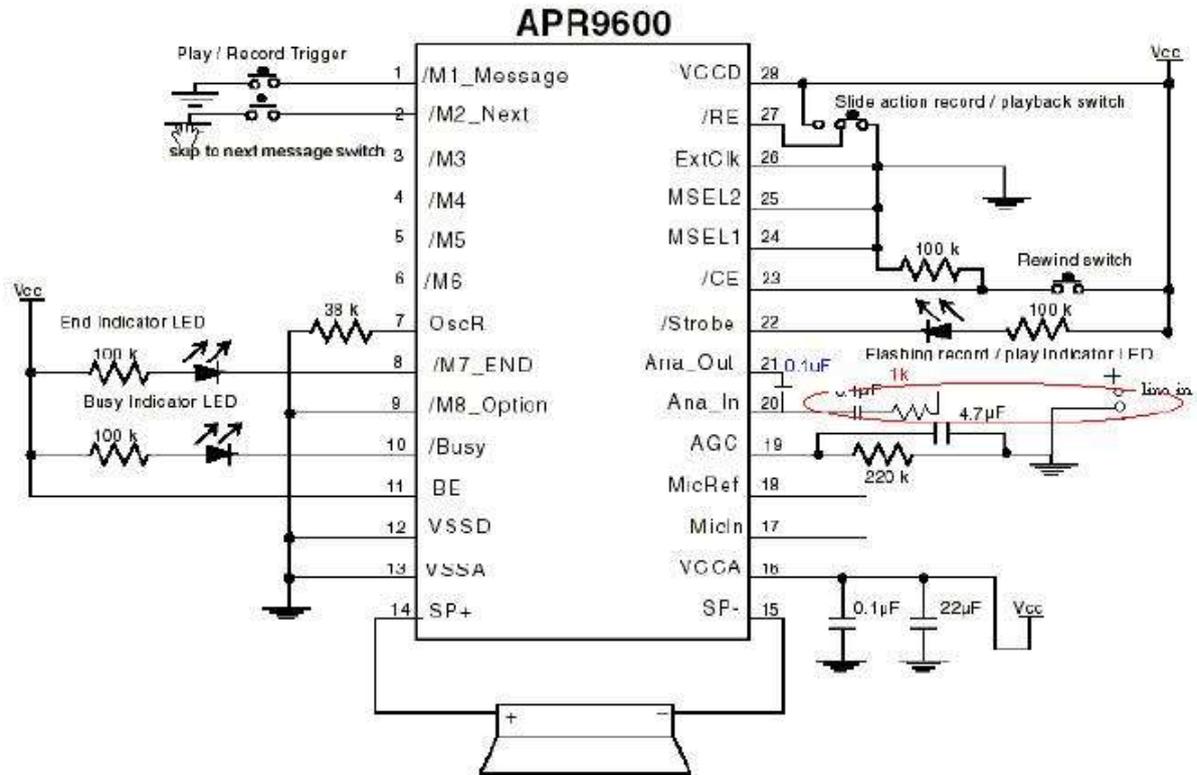
### پخش پیام:

برای پخش پیام پایه ۲۷ را به خط تغذیه وصل می کنیم و با دادن پالس منفی به هر کدام از پایه ها پیام مربوط به آن پایه پخش می شود.

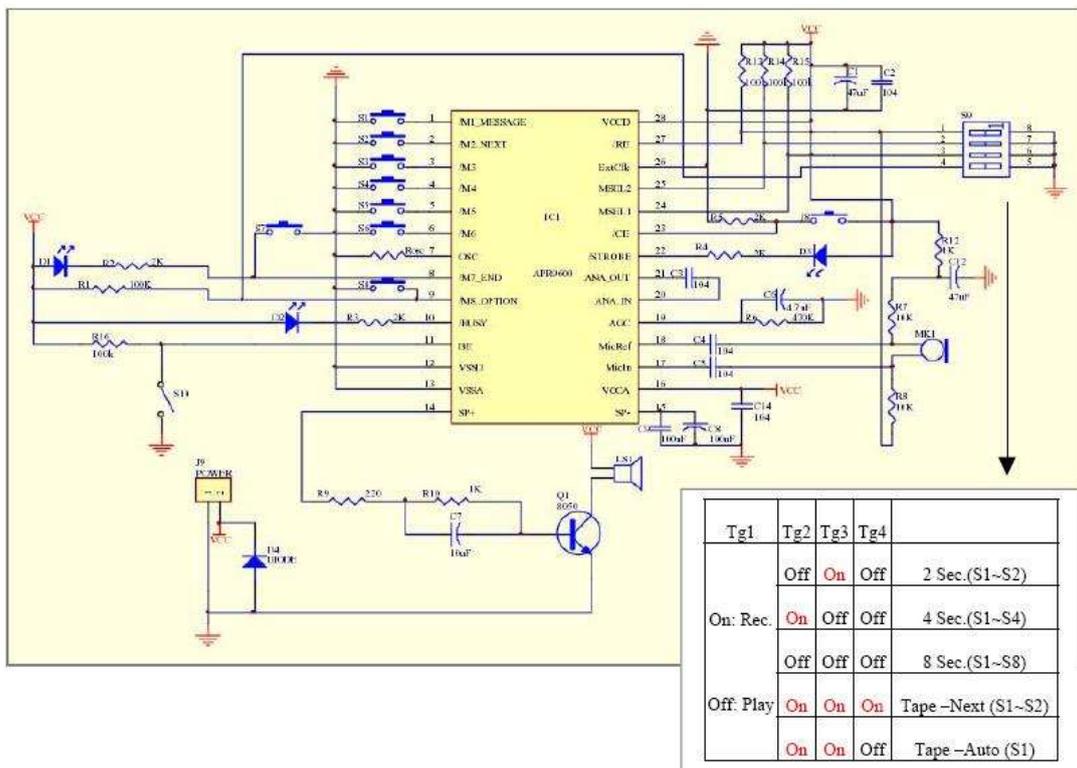
مدار ساختار دسترسی تصادفی به صورت زیر می باشد:



اگر بخواهیم صدایی از وسیله ای ضبط شود (مثلاً از یک ضبط صوت) از مدار زیر استفاده می کنیم.



اگر بخواهیم از تمام ساختارهای این IC استفاده کنیم می توانیم از مدار زیر استفاده کنیم:



## عملکرد پایه های آی سی ضبط و پخش صوت :

نام پایه	شماره پایه	کارکرد در ساختار دسترسی تصادفی	عملکرد در روش ضبط کردن	
			تنظیمات برگردان اتوماتیک	تنظیمات معمولی
<b>/M1- Message</b>	۱	پیام ۱: این پین برای ضبط و پخش پیام ۱ استفاده می شود	پیام: با اعمال یک لبه پایین رونده روی این پایه می توان پیام بعدی را ضبط یا پخش کرد	با اعمال یک لبه پایین رونده روی این پایه می توان پیام بعدی را ضبط یا پخش کرد
<b>/M2-Next</b>	۲	پیام ۲: این پین برای ضبط و پخش پیام ۲ استفاده می شود	در این ساختار این پایه باید جدا باشد	پیام بعدی: این پین با سطح پایین فعال می شود و با فعال کردن این پین به پیام بعدی چه در حال ضبط و چه در حال پخش می رود
<b>/M3</b>	۳	پیام ۳: این پین برای ضبط و پخش پیام ۳ استفاده می شود	در این ساختار این پایه باید جدا باشد	در این ساختار این پایه باید جدا باشد
<b>/M4</b>	۴	پیام ۴: این پین برای ضبط و پخش پیام ۴ استفاده می شود	در این ساختار این پایه باید جدا باشد	در این ساختار این پایه باید جدا باشد
<b>/M5</b>	۵	پیام ۵: این پین برای ضبط و پخش پیام ۵ استفاده می شود	در این ساختار این پایه باید جدا باشد	در این ساختار این پایه باید جدا باشد

/M6	۶	پیام ۶: این پین برای ضبط و پخش پیام ۶ استفاده می شود	در این ساختار این پایه باید جدا باشد	در این ساختار این پایه باید جدا باشد
OSCR	۷	این پایه، پایه نوسان ساز می باشد که توسط مقاومت به زمین وصل می شود (مراجعه کنید به جدول X)	مانند ساختار ۱ می باشد	مانند ساختار ۱ می باشد
/M7-End	۸	پیام ۷: این پین برای ضبط و پخش پیام ۷ استفاده می شود	با اعمال یک سطح پایین به این پایه تمام پیام های ضبط شده پخش می شود	با اعمال یک سطح پایین به این پایه تمام پیام های ضبط شده پخش می شود
/M8-Option	۹	پیام ۸: این پین برای ضبط و پخش پیام ۸ استفاده می شود	تنظیمات: این پین در ترکیب MSEL1 و MSEL2 عملیات ضبط و پخش را انجام می دهد. (مراجعه کنید به جدول ۱ برای رمزگشایی اطلاعات)	MSEL2 و MSEL1 ساختار عملیات ضبط و پخش را تعیین می کند (پیشنهاد جدول ۱ برای رمزگشایی اطلاعات)
/Busy	۱۰	این پایه فعالیت تراشه را نشان می دهد	مانند ساختار ۱ می باشد	مانند ساختار ۱ می باشد
BE	۱۱	اگر این پایه به روی Vcc وصل شود صدای Beep فعال می شود و اگر به زمین وصل شود صدای Beep غیر فعال می شود	مانند ساختار ۱ می باشد	مانند ساختار ۱ می باشد
VssD	۱۲	اتصال زمین دیجیتال می باشد و به زمین مدار وصل می شود	مانند ساختار ۱ می باشد	مانند ساختار ۱ می باشد
VssA	۱۳	اتصال زمین آنالوگ می باشد و به	مانند ساختار ۱ می باشد	مانند ساختار ۱ می باشد

		زمین مدار وصل می شود		باشد
<b>SP+</b>	۱۴	خروجی مثبت بلندگو می باشد این پایه به ترمینال مثبت بلندگو وصل می شود توان خروجی این پایه یک وات و مقاومت آن ۱۶ اهم می باشد و بلندگو نباید کمتر از ۸ اهم باشد زیرا ممکن است به قطعه آسیب برساند	مانند ساختار ۱ می باشد	مانند ساختار ۱ می باشد
<b>SP-</b>	۱۵	این پایه به ترمینال منفی بلندگو وصل می شود	مانند ساختار ۱ می باشد	مانند ساختار ۱ می باشد
<b>VccA</b>	۱۶	منبع تغذیه مثبت آنالوگ: این پایه برای تغذیه مدارهای آنالوگ می باشد که روی تراشه قرار دارد و باید به خط مثبت تغذیه وصل شود	مانند ساختار ۱ می باشد	مانند ساختار ۱ می باشد
<b>MicIn</b>	۱۷	این پایه به میکروفون ورودی وصل می شود (مانند شکل در طرح اصلی)	مانند ساختار ۱ می باشد	مانند ساختار ۱ می باشد
<b>MicRef</b>	۱۸	این پایه به ترمینال منفی میکروفون وصل می شود	مانند ساختار ۱ می باشد	مانند ساختار ۱ می باشد
<b>AGC</b>	۱۹	کنترل اتوماتیک بهره زمان بندی شده: این پایه به یک شبکه RC متصل می شود و این شبکه RC زمانبندی AGC را مشخص می کند این زمان بندی قبل از شروع کار AGC تاخیر ایجاد می کند. مقدار RC در طرح اصلی نشان داده شده	مانند ساختار ۱ می باشد	مانند ساختار ۱ می باشد

		است و کارکرد این IC را برای کاربردهای صدا بهینه می کند.		
Ana-In	۲۰	ورودی آنالوگ: این پایه به وسیله ۰/۱ میکرو به پایه Ana-Out وصل می شود	مانند ساختار ۱ می باشد	مانند ساختار ۱ می باشد
Ana-Out	۲۱	خروجی آنالوگ: این پایه به وسیله یک خازن ۰/۱ میکرو به پایه Ana-In وصل می شود	مانند ساختار ۱ می باشد	مانند ساختار ۱ می باشد
/Strobe	۲۲	پالس مدار الکتریکی: این پایه برنامه ریزی از هر قسمت اختصاصی ضبط را نشان می دهد لبه پایین رونده شروع ضبط در ناحیه را نشان می دهد و لبه بالا رونده نیمه پر بودن ناحیه ضبط را نشان می دهد	مانند ساختار ۱ می باشد	مانند ساختار ۱ می باشد
/CE	۲۳	انتخاب تراشه: با اعمال یک سطح پایین روی این پایه تراشه فعال می شود. معکوس کردن این پایه سازماندهی شکل چند پیغام را به حالت اول برمی گرداند	مانند ساختار ۱ می باشد	مانند ساختار ۱ می باشد
MSEL1	۲۴	ساختار انتخاب ۱: این پایه در ترکیب با M8-Option و MSEL2 ساختار عامل ضبط و پخش را تعیین می کند (به جدول ۱ مراجعه شود)	مانند ساختار ۱ می باشد	مانند ساختار ۱ می باشد
MSEL2	۲۵	ساختار انتخاب ۲: این پایه در ترکیب	مانند ساختار ۱ می باشد	مانند ساختار ۱ می باشد

		با M8-Option و MSEL1 ساختار عامل ضبط و پخش را تعیین می کنند (به جدول ۱ مراجعه شود)		باشد
Extclk	۲۶	پالس خارجی: این پالس می تواند به جای پالس های خارجی برای کنترل برنامه های بزرگتر و بادقت بیشتر استفاده شود وقتی از پالس داخلی استفاده می شود این پایه به زمین مدار وصل می شود	مانند ساختار ۱ می باشد	مانند ساختار ۱ می باشد
/RE	۲۷	فعال کردن ساختار ضبط: این پایه خواندن و نوشتن در تراشه را کنترل می کند با یک کردن این پایه تراشه برای خواندن اطلاعات فعال می شود	مانند ساختار ۱ می باشد	مانند ساختار ۱ می باشد
VccD	۲۸	تغذیه مثبت دیجیتال: این پایه برای مدارای دیجیتالی طراحی شده روی تراشه می باشد و به خط مثبت تغذیه وصل میشود	مانند ساختار ۱ می باشد	مانند ساختار ۱ می باشد

## ارتباط بین میکرو و IC ضبط و پخش صدا

این دستگاه برای تشخیص شش رنگ طراحی شده است. همان طور که قبلاً در بخش مشخصات IC صدا گفته شد در ساختار random access می توان هشت پیام را در هشت پایه ضبط کرد. بنابراین در این دستگاه IC صدا در این ساختار استفاده شده است. قبل از اینکه IC صدا در مدار اصلی قرار بگیرد باید صدا در پایه های آن ضبط شود برای این کار مدار random access در فصل IC صدا را می بندیم و پایه RE و CE را به زمین وصل می کنیم. و با اعمال سطح پائین به هر پایه می توان در آن صدای مورد نظر را ضبط کرد. در عمل این روش به درستی

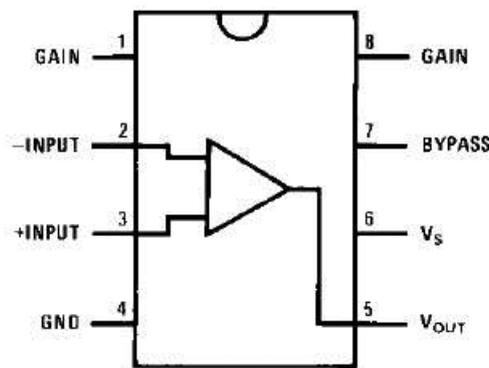


## طبقه تقویت صوت

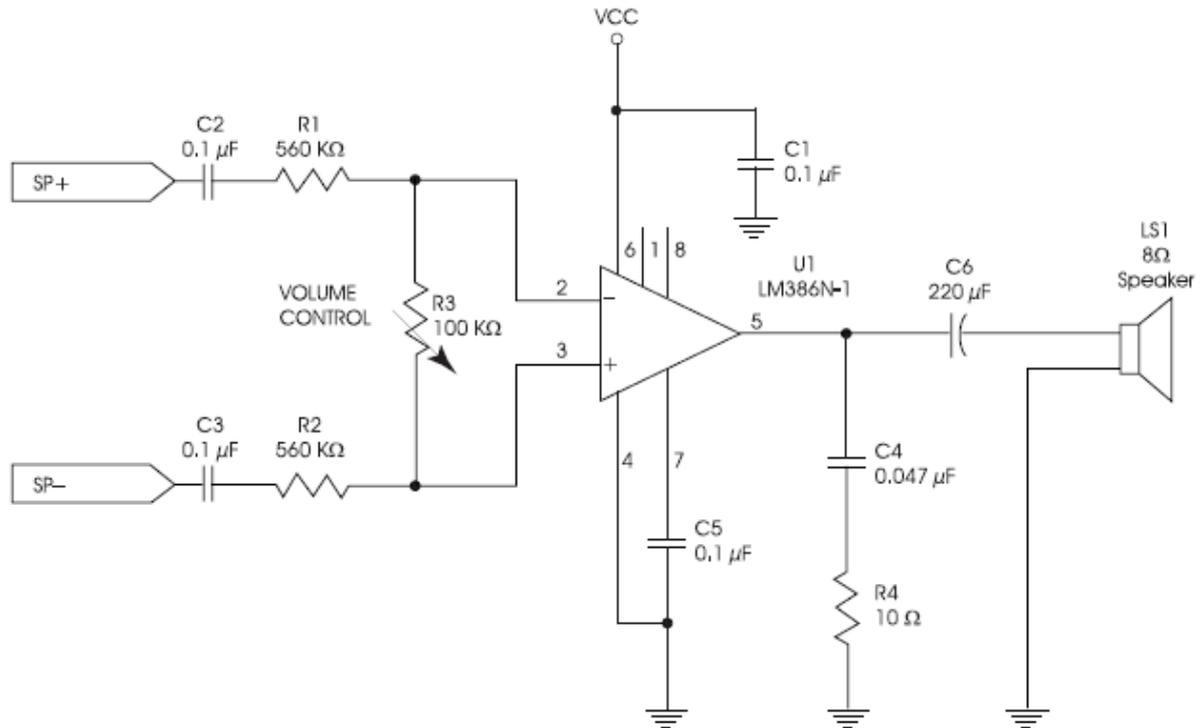
**IC** صدایی که در این دستگاه استفاده شده است خود دارای تقویت کننده صوت می باشد و اگر بلند گو را به خروجی آن متصل کنید صدا به وضوح شنیده می شود و اگر بلندگو را در داخل یک جعبه قرار دهید نیازی به تقویت صدا نخواهید داشت. اما در این پروژه برای این که صدای بلندتری داشته باشیم صدای خروجی **IC** صدا را به وسیله یک طبقه تقویت صوت افزایش می دهیم. در این طبقه از یک **IC** تقویت صوت به نام **LM386** استفاده شده است این **IC** جزو **IC** های معروف و پرکاربرد بازار می باشد.

### آی سی تقویت کننده صدا LM386

یک آی سی تقویت کننده سیگنال صوت می باشد با توان خروجی حداکثر حدود یک وات که در بسته بندی های ۸ پایه **DIP** و **SO** توسط شرکت **National Semiconductor** عرضه شده است. این تقویت کننده برای دستگاه های صوتی با توان پایین ارائه شده است که با ولتاژهای ۴ تا ۱۲ ولت کار می کند و توان مصرفی پایینی دارد و به قطعات جانبی کمی برای کار احتیاج دارد.



پایه های این **IC** به صورت بالا می باشد



طبقه تقویت صوتی که در این پروژه استفاده شده است به صورت بالا می باشد .

### شرح برنامه

هسته اصلی برنامه یک فرکانس متر می باشد . برنامه به این صورت عمل می کند که پایه **B2** به عنوان ورودی

کانتر تعریف شده است که به پایه **VOUT** سنسور متصل می باشد.

پایه **B.3** و **B.4** به عنوان خروجی صفر تعریف شده است که به پایه های **S3** و **S2** سنسور متصل می باشد.

پایه **B.5** به عنوان ورودی یک تعریف شده است.

برنامه در ابتدا خروجی های میکرو را به طوری که در بالا گفته شده ، تعریف می کند سپس مقدار اولیه تایمر را

برابر با ۶ قرار می دهد و در حلقه **WHILE** منتظر صفر شدن پین **B.5** می ماند . بعد از صفر شدن پایه **B.5**

تایمر کانتر را فعال می کند و بعد از ۱۰۰ میلی ثانیه که توسط تایمر ایجاد می شود مقدار پالس های شمارش شده

توسط کانتر را ضربدر ۱۰ می کند و در متغیری به نام **A** می ریزد و سپس متغیر **X** را چک می کند اگر **X=1** باشد

**A** را در مغیر **RED** می ریزد و پایه های **B.3** و **B.4** هر دور را یک می کند ودوباره فرکانس را اندازه می گیرد و

**X=2** قرار می دهد این بار فرکانس را در متغیر **GREEN** می ریزد و پایه های **B3** و **B4** را به ترتیب ۰ و ۰ می کند

و **X=3** قرار داده فرکانس را اندازه گرفته و در متغیر **BLUE** می ریزد. در ادامه این سه متغیر را به سه فرکانس

که قبلا اندازه گرفته شده تک تک مقایسه می کند و اگر فرکانس ما در محدوده فرکانس هر یک از رنگ ما باشد پایه مربوط به آن رنگ صفر می شود . پایه **B6** نیز ۱ و صفر می شود و باعث فعال شدن **IC** صدا و پخش صدای مورد نظر می باشد و بعد از ۲ ثانیه پایه **CE** دوباره ۰ می شود. همانطور که قبلا گفته شد پایه های **D0** تا **D5** میکرو به پایه های ۱ تا ۶ **IC** و پایه **D6** میکرو به پایه **IC CE** صدا وصل می باشد. لازم به ذکر است که این برنامه با کامپایلر **codvision** و به زبان **C** نوشته شده است. در زیر می توانید برنامه را مشاهده کنید.

/\*\*\*\*\*

Project :finding color divice

Version :

Date : 2007/01/29

Author : mahdi

Company : khodabakhsh

Comments:

Chip type : ATmega16L

Program type : Application

Clock frequency : 1.000000 MHz

Memory model : Small

External SRAM size : 0

Data Stack size : 256

\*\*\*\*\*/

```
#include <mega16.h>
```

```
// Alphanumeric LCD Module functions
```

```
#asm
```

```
.equ __lcd_port=0x1B ;PORTA
```

```
#endasm
```

```
#include <lcd.h>
```

```
#include <stdio.h>
#include <delay.h>
int i,c;
long int A,red,green,blue;
unsigned char lcd_buffer[16],B,x;

// Timer 0 overflow interrupt service routine
interrupt [TIM0_OVF] void timer0_ovf_isr(void)
{
// Reinitialize Timer 0 value
TCNT0=0x06;
// Place your code here
i++;
if(i==400)
{
TCCR0=0x00;
TCCR1B=0x00;
A=B*65536;
A=(TCNT1+A)*10;
TCNT1=0;
lcd_clear();
sprintf(lcd_buffer,"%ld",A);
lcd_gotoxy(0,0);
lcd_puts(lcd_buffer);
delay_ms(100);
x++;
B=0;
i=0;

if(x==1)
{
red=A;
PORTB.2=1;
```

```
PORTB.3=1;
TCCR0=0X01;
TCCR1B=0x07;
}
if(x==2)
{
green=A;
PORTB.2=0;
PORTB.3=1;
TCCR0=0X01;
TCCR1B=0x07;
}
if(x==3)
{
blue=A;
x=0;
c=0;
PORTB.2=0;
PORTB.3=0;

if(540<=red&&red<=790&&190<=green&&green<=260&&160<=blue&&blue<=210)
{
PORTD.0=0;
delay_ms(50);
PORTD.6=1;
delay_ms(100);
PORTD.6=0;
delay_ms(2000);
PORTD.6=1;
delay_ms(100);
PORTD.6=0;
delay_ms(50);
PORTD.0=1;
```

```
}
```

```
if(270<=red&&red<=470&&400<=green&&green<=650&&260<=blue&&blue<=400)
```

```
{
```

```
PORTD.1=0;
```

```
delay_ms(50);
```

```
PORTD.6=1;
```

```
delay_ms(100);
```

```
PORTD.6=0;
```

```
delay_ms(2000);
```

```
PORTD.6=1;
```

```
delay_ms(100);
```

```
PORTD.6=0;
```

```
delay_ms(50);
```

```
PORTD.1=1;
```

```
}
```

```
if(190<=red&&red<=270&&200<=green&&green<=275&&310<=blue&&blue<=460)
```

```
{
```

```
PORTD.2=0;
```

```
delay_ms(50);
```

```
PORTD.6=1;
```

```
delay_ms(100);
```

```
PORTD.6=0;
```

```
delay_ms(2000);
```

```
PORTD.6=1;
```

```
delay_ms(100);
```

```
PORTD.6=0;
```

```
delay_ms(50);
```

```
PORTD.2=1;
```

```
}
```

```
if(600<=red&&red<=8400&&280<=green&&green<=390&&190<=blue&&blue<=250)
```

```
{  
  PORTD.3=0;  
  delay_ms(50);  
  PORTD.6=1;  
  delay_ms(100);  
  PORTD.6=0;  
  delay_ms(2000);  
  PORTD.6=1;  
  delay_ms(100);  
  PORTD.6=0;  
  delay_ms(50);  
  PORTD.3=1;  
}
```

```
if(570<=red&&red<=800&&560<=green&&green<=690&&240<=blue&&blue<=290)
```

```
{  
  PORTD.4=0;  
  delay_ms(50);  
  PORTD.6=1;  
  delay_ms(100);  
  PORTD.6=0;  
  delay_ms(2000);  
  PORTD.6=1;  
  delay_ms(100);  
  PORTD.6=0;  
  delay_ms(50);  
  PORTD.4=1;  
}
```

```
if(210<=red&&red<=390&&150<=green&&green<=220&&210<=blue&&blue<=300)
```

```
{  
  PORTD.5=0;  
  delay_ms(50);
```

```
PORTD.6=1;
delay_ms(100);
PORTD.6=0;
delay_ms(2000);
PORTD.6=1;
delay_ms(100);
PORTD.6=0;
delay_ms(50);
PORTD.5=1;
}
}
}

}

// Timer 1 overflow interrupt service routine
interrupt [TIM1_OVF] void timer1_ovf_isr(void)
{
// Place your code here
B++;
TCNT1=0;
}

// Declare your global variables here

void main(void)
{
// Declare your local variables here

// Input/Output Ports initialization
```

```
// Port A initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T
PORTA=0x00;
DDRA=0x00;

// Port B initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T
PORTB=0x10;
DDRB=0x0c;

// Port C initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T
PORTC=0x00;
DDRC=0x00;

// Port D initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T
PORTD=0x3F;
DDRD=0x7F;

// Timer/Counter 0 initialization
// Clock source: System Clock
// Clock value: 1000.000 kHz
// Mode: Normal top=FFh
// OC0 output: Disconnected
TCCR0=0x00;
TCNT0=0x06;
OCR0=0x00;

// Timer/Counter 1 initialization
```

```
// Clock source: T1 pin Rising Edge
// Mode: Normal top=FFFFh
// OC1A output: Discon.
// OC1B output: Discon.
// Noise Canceler: Off
// Input Capture on Falling Edge
// Timer 1 Overflow Interrupt: On
// Input Capture Interrupt: Off
// Compare A Match Interrupt: Off
// Compare B Match Interrupt: Off
TCCR1A=0x00;
TCCR1B=0x00;
TCNT1H=0x00;
TCNT1L=0x00;
ICR1H=0x00;
ICR1L=0x00;
OCR1AH=0x00;
OCR1AL=0x00;
OCR1BH=0x00;
OCR1BL=0x00;

// Timer(s)/Counter(s) Interrupt(s) initialization
TIMSK=0x05;

// Analog Comparator initialization
// Analog Comparator: Off
// Analog Comparator Input Capture by Timer/Counter 1: Off
ACSR=0x80;
SFIOR=0x00;

// LCD module initialization
lcd_init(16);
```

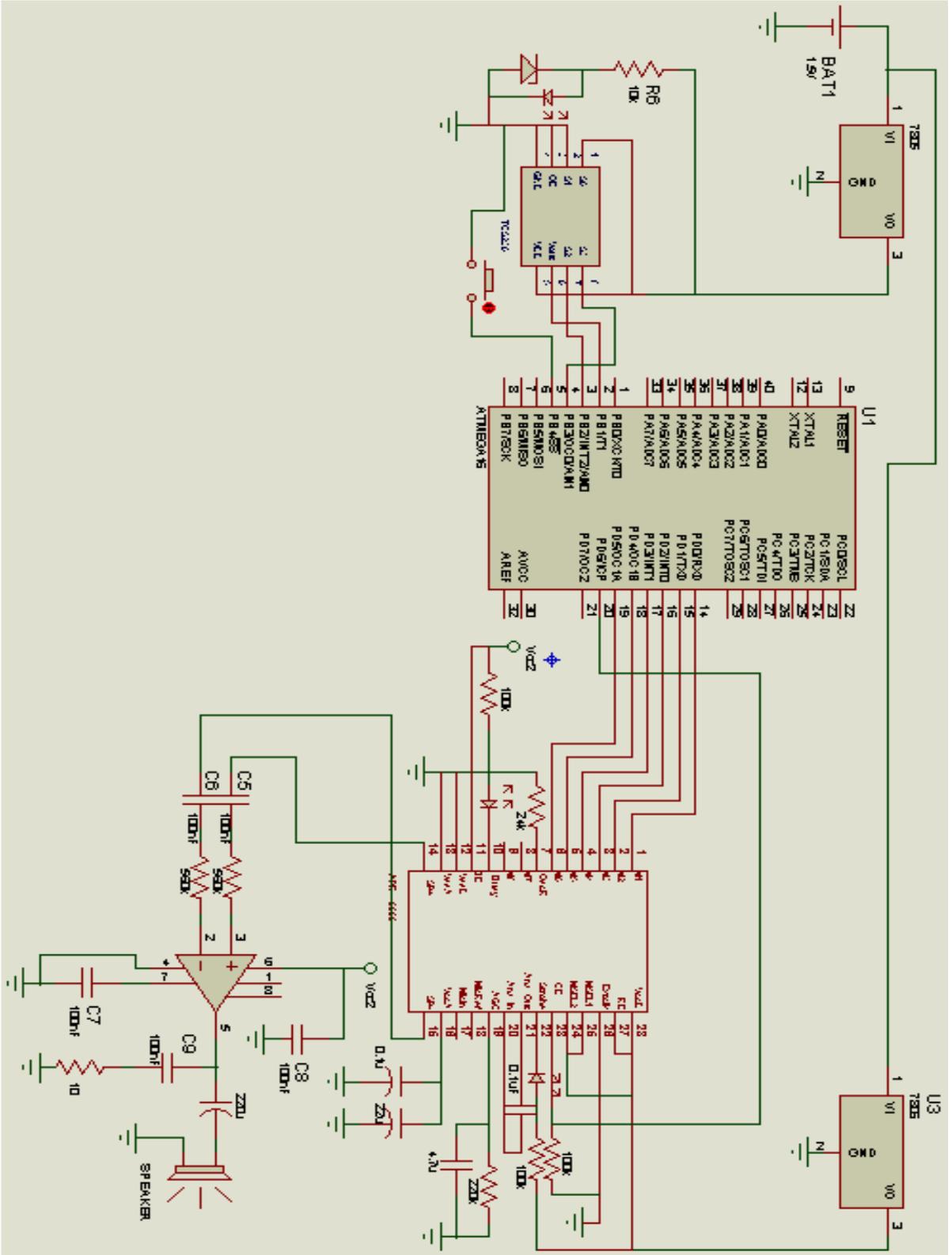
```
i=0;
B=0;
x=0;

// Global enable interrupts
#asm("sei")

while (1)
{
    if(PINB.4==0)
    {
        delay_ms(50);
        if(PINB.4==0)
        {
            TCCR0=0x01;
            TCCR1B=0x07;
        }
    }

};
}
```

مدار کامل پروژه به صورت زیر میباشد.



منابع و ماخذ :

[WWW.TAOSINC.COM](http://WWW.TAOSINC.COM)  
[WWW.APLUSINC.COM.TW](http://WWW.APLUSINC.COM.TW)  
[WWW.DATASHEETCATALOG.COM](http://WWW.DATASHEETCATALOG.COM)

نویسنده : مهدی خدابخش ( سال ۲۰۰۷ )

ویرایش و تکمیل :جهاننیده (سال ۲۰۱۵)

MicroDesigner.ir

زکات علم نشر آن است.  
حضرت علی (ع)