

به نام خدا

# کنترل از راه دور ربات ها و وسایل برقی با پیامک

با استفاده از ماژول سیم ۹۰۰

**ROBOT**



**SIM 900**



**CONTROLLING DEVICES VIA SMS**



**مهدی ویسی**

کارشناس ارشد مهندسی فناوری اطلاعات - سیستم های چند رسانه ای

پاییز ۱۳۹۴

## چکیده:

توسعه فناوری های نوین راه را برای ارتباطات بین افراد و وسایل و تجهیزات از راه دور تسهیل نموده است بطوریکه مثلا یک کشاورز بتواند در خانه خود ، موتور مربوط به آبیاری زمین هایش را از فاصله چند کیلومتری خاموش و روشن نماید و یا یک ربات از فاصله بسیار دور پیام ها را دریافت کند و دستور مورد نظر را انجام دهد و بسیاری کاربردهای دیگر. در این مقاله سعی شده است که نحوه ارتباط ماژول SIM900 با میکروکنترلر ATmega32 بیان شود که سیم ۹۰۰ پیام را دریافت می کند و به میکروکنترلر تحویل میدهد و میکروکنترلر بر اساس نوع پیام ، دستور اجرای فرامین خاصی را صادر می کند.

صفحه	فهرست مطالب
۴	مقدمه
۵	سخت افزار پروژه و نیازمندی ها
۱۱	دستورات AT Command
۱۴	نرم افزار پروژه و کدنویسی
۲۳	منابع

## مقدمه

امروزه ارزش وقت و زمان بسیار زیاد است و تقسیم بندی و برنامه ریزی کارها از اتلاف آن جلوگیری می کند و استفاده از فناوری های نوین می تواند ما را یاری کند. یک وسیله برقی یا یک ربات را در نظر بگیرید که کار خاصی را برای ما انجام می دهند و البته نیاز به نگهداری و کنترل و مراقبت هم دارند، اما این کنترل می تواند از راه دور هم صورت گیرد یعنی ما در فاصله دورتر در حال انجام کار دیگری باشیم و همزمان بر عملکرد دستگاه راه دورمان نظارت کنیم و کنترل خاموش و روشن کردن و تنظیم نوع عملکرد و سرعت انجام دادن کاری خاص، و دستورات مختلف دیگر را داشته باشیم. این کار باعث صرفه جویی بسیار در زمان می شود. اگر وسیله مورد نظر در جایی قرار داشته باشد که حضور مستقیم انسان در آنجا دشوار و یا خطرناک باشد، آنگاه ارزش این نوع ارتباط بیشتر هم می شود.

در اینجا ما یک مثال ساده را مورد بحث قرار می دهیم که خواننده محترم با بسط آن می تواند مثال ها و پروژه های پیچیده تر و کاربردی تر را انجام دهد. نحوه ارتباط میکروکنترلر یا کامپیوتر با ماژول سیم ۹۰۰ از طریق پیامک، و کنترل از راه دور یک موتور و ال ای دی و ... را بررسی می کنیم که می تواند بخشی از یک ربات صنعتی یا غیرصنعتی و یا یک وسیله برقی باشد.

در ادامه ما در ابتدا سخت افزار پروژه را مورد بررسی قرار می دهیم و وسایل و امکانات لازم را شرح می دهیم و سپس در بخش بعدی تعدادی از دستورات AT Command مورد نیاز را برای برقراری ارتباط با سیم ۹۰۰ یادآوری می نمایم و در نهایت کد مربوط به پروژه را به زبان برنامه نویسی سی در نرم افزار کدویژن آی وی آر (CodeVisionAVR) بیان می کنیم.

لازم به ذکر است که فرض نویسنده بر این استوار است که خواننده محترم آشنایی لازم را با قطعات الکترونیک، آی سی ها، و میکروکنترلرها و نرم افزارهای مربوطه داراست.

## سخت افزار پروژه و نیازمندی ها

ما به حداقل اجزاء زیر نیاز داریم:

### ۱- ماژول SIM 900 برای ارسال و دریافت پیامک ، همراه با سیم کارت و آنتن آن.

ماژول SIM900 یک GSM MODEM می باشد که دستگاهی است که امکانات ارسال و دریافت پیامک و زنگ زدن به یک شماره و مکالمه ، و استفاده از سرویس اینترنت همراه GPRS را برای ما فراهم می آورد.

### ۲-رگولاتور LM350 برای تنظیم ولتاژ و جریان ورودی به SIM 900

ولتاژ کار باید بین ۳,۴ ولت تا ۴,۵ ولت باشد که البته ۴ ولت خوب و نرمال است (طبق دیتاشیت)، در غیر اینصورت ماژول خاموش می شود. این ماژول به جریان لحظه ای ۲ آمپر نیاز دارد و به همین خاطر باید از منبع تغذیه مناسب بهره برد. شایان ذکر است که حداکثر جریان خروجی رگولاتور LM350 سه آمپر می باشد.

### ۳-میکرو کنترلر ATmega32

میکروکنترلر پیامک را از ماژول سیم ۹۰۰ می گیرد و با توجه محتوای پیام، برنامه ای خاص را اجرا می کند و کاری مرتبط را انجام می دهد. و یا اینکه بالعکس، میکرو کنترلر به ماژول سیم ۹۰۰ دستور می دهد که پیامکی را به شماره ای خاص ارسال کند.

### ۴- LCD شانزده ستونه دو خطی برای نمایش اطلاعات و پیامک و ...

البته اگر LCD بزرگتر باشد بهتر است چون ممکن است همه پیام و ملحقات آن مانند تاریخ و ... در LCD جا نشود.

## ۵- سایر موارد شامل مقاومت ها و خازن و غیره که در شکل آورده شده اند.

قطعات دیگر بستگی به شما دارد که چند وسیله را بخواهید کنترل کنید مثلا درایور موتور L293 برای کنترل یک موتور dc و ... .

## ۶- نرم افزار پروتئوس برای رسم شماتیک پروژه

البته سیم ۹۰۰ بطور معمول در کتابخانه پروتئوس وجود ندارد و باید فایل های مربوط به آن را از اینترنت دانلود کنید و طبق دستورالعمل مربوطه آن را در پوشه نصب پروتئوس قرار دهید. هرچند فقط شکل ظاهری آن نمایش داده می شود و در واقع پروتئوس آن را شبیه سازی نمی کند.

## ۷- نرم افزار کد ویژن آی وی آر برای نوشتن کدهای برنامه

برنامه را به زبان سی در محیط نرم افزار CodeVisionAVR می نویسیم. این برنامه باید نهایتا بر روی میکروکنترلر قرار گیرد.

## ۸- پروگرامر جهت انتقال برنامه نوشته شده از کامپیوتر به روی میکروکنترلر.

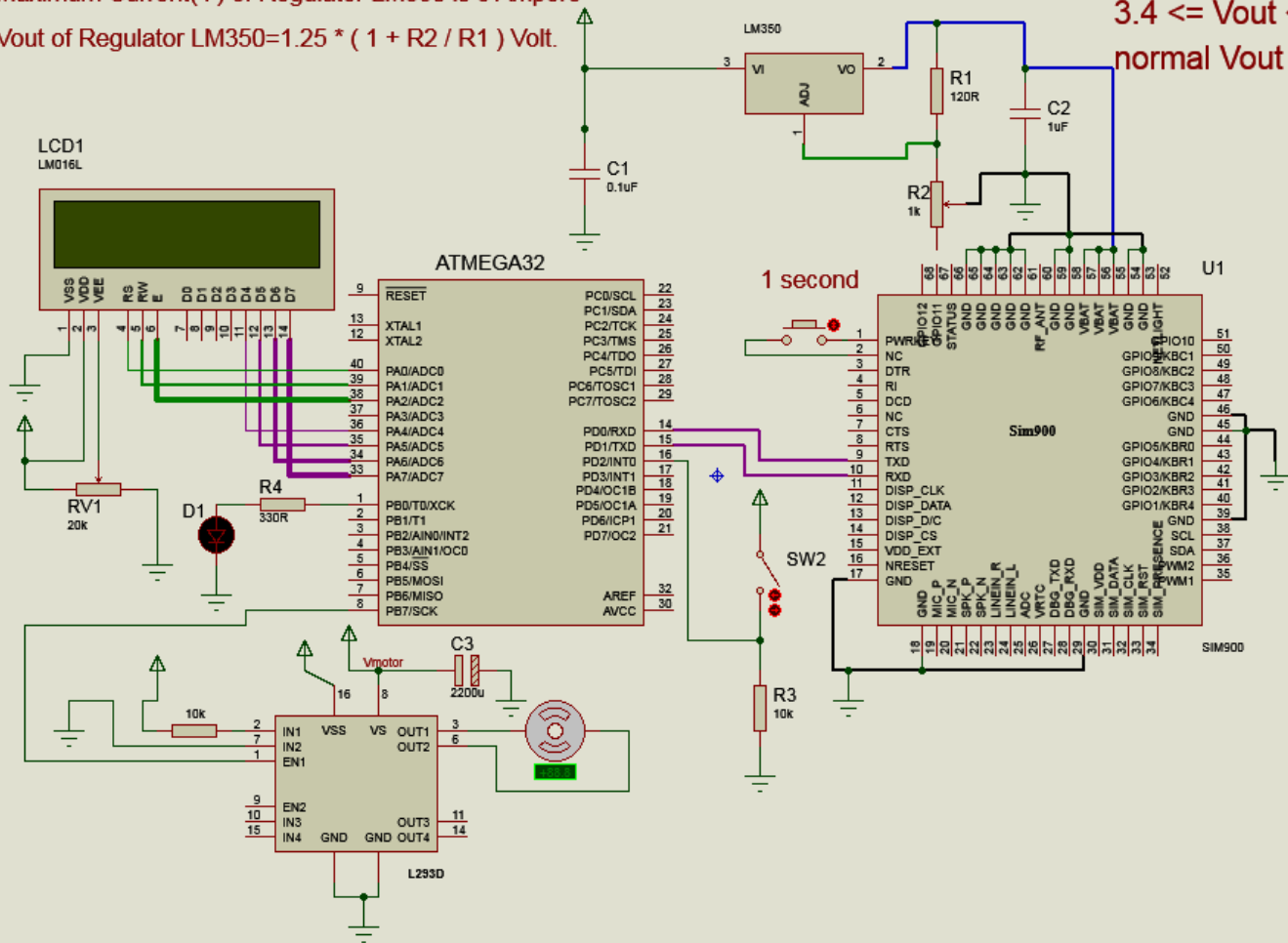
پروگرامر های فراوانی در بازار وجود دارند که بر روی بیشتر سیستم عامل ها نصب می شوند. با استفاده از دستگاه پروگرامر می توانید برنامه نوشته شده در CodeVisionAVR را به میکروکنترلر تان انتقال دهید.

## از پروتئوس 8.2 سرویس پک 2 استفاده کردم

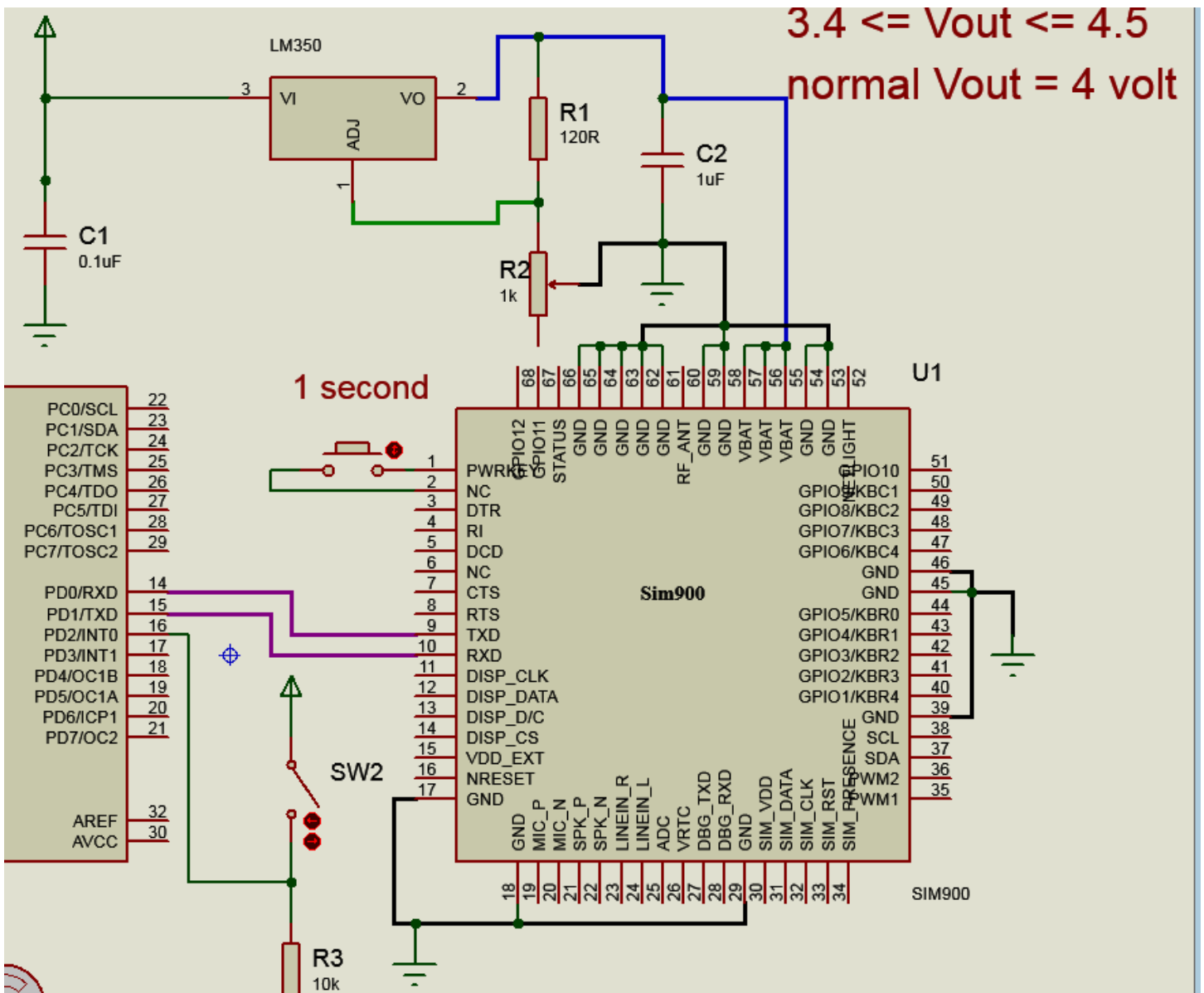
Maximum Current (i) of Regulator LM350 is 3 Ampere

$V_{out}$  of Regulator LM350 =  $1.25 * (1 + R2 / R1)$  Volt.

$3.4 \leq V_{out} \leq 4.5$   
normal  $V_{out} = 4$  volt

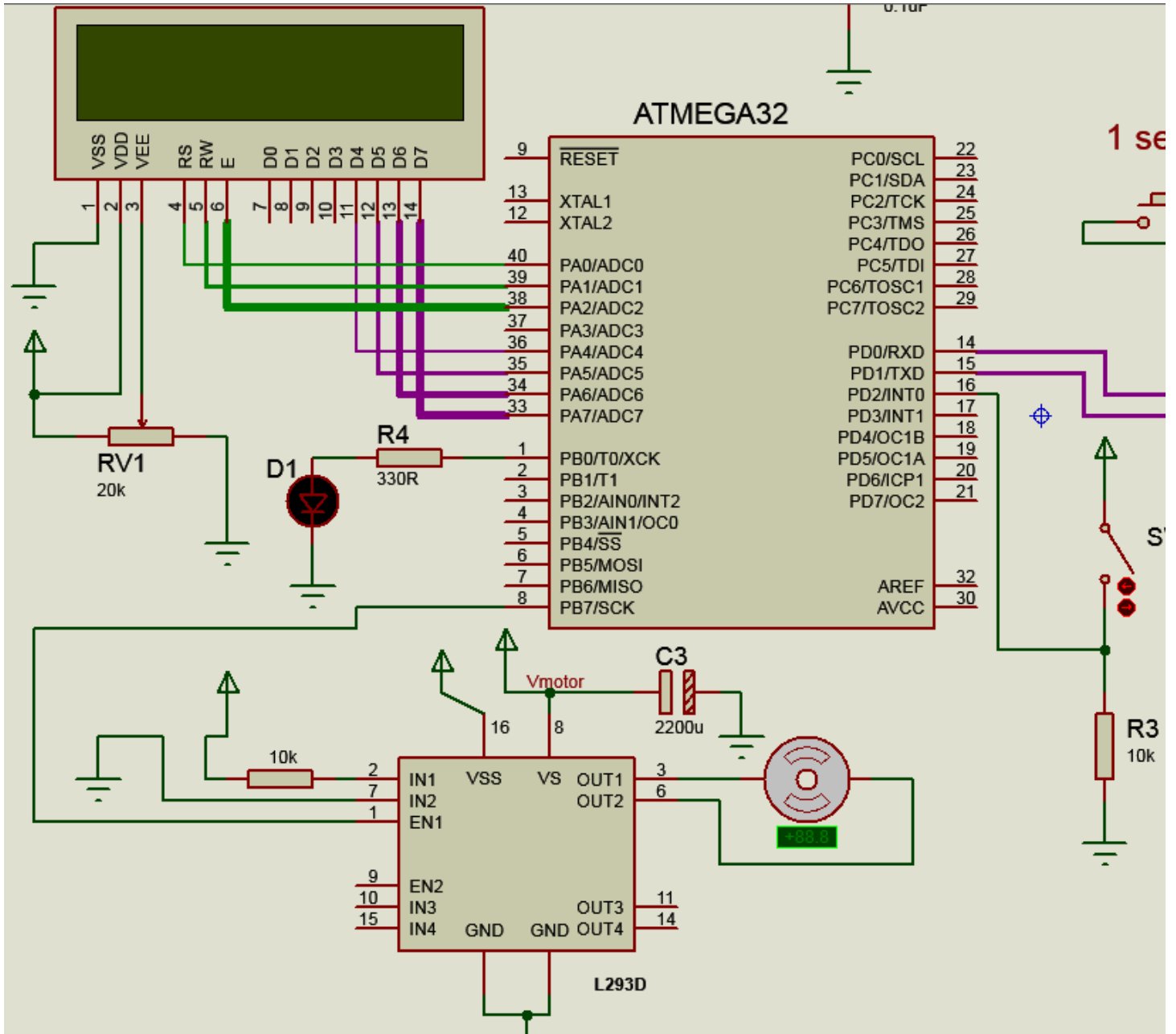


## نمای بزرگتر سیم ۹۰۰ و منبع تغذیه





## نمای بزرگتر میکروکنترلر و ال سی دی و درایور موتور و ...



**نکته:** همانگونه که مشاهده می‌نمایید، پایه‌های RX یا TX میکروکنترلر به پایه دریافت داده یا RX سیم ۹۰۰ متصل شده است و بالعکس پایه RX میکروکنترلر به پایه TX سیم ۹۰۰ متصل شده است.

**نکته:** روش دیگر برای روشن کردن سیم ۹۰۰ این است که به مدت یک ثانیه پایه power را با یک مقاومت یک کیلو اهم به زمین وصل کنیم (و همچنین برای خاموش کردن آن)

**نکته:** همه پایه‌های مثبت (VBAT) باید به مثبت منبع تغذیه متصل باشند (یعنی خروجی رگولاتور) و همچنین همه پایه‌های منفی (GND) هم باید به منفی منبع تغذیه متصل باشند.

**نکته:** بهتر است مدار مربوط به منبع تغذیه را طبق توصیه و راهنمای ماژول سیم ۹۰۰ ببینید.

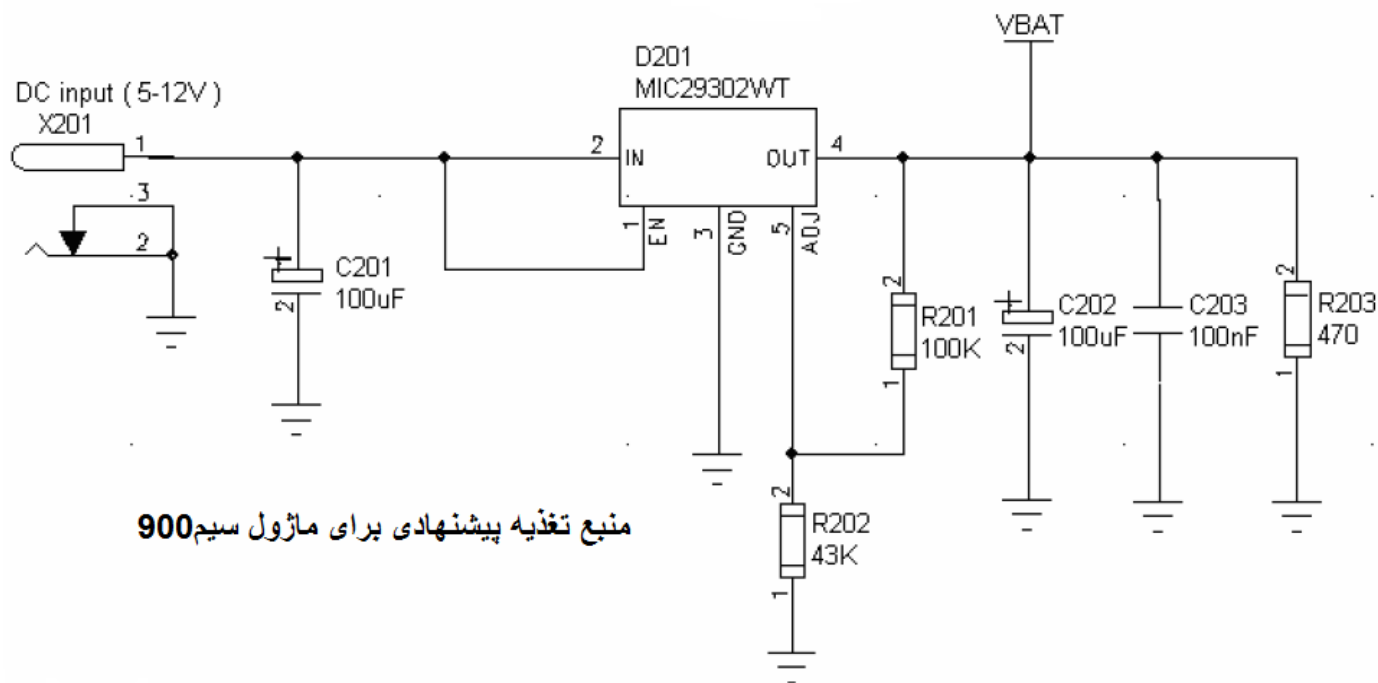


Figure 4: Reference circuit of the source power supply input

## دستورات AT Command

ماژول SIM900 از طریق پورت سریال UART با میکروکنترلر ارتباط برقرار می کند و می تواند بطور اتوماتیک سرعت ارتباط را تشخیص دهد و در چند سرعت مختلف می تواند کار کند که البته پیش فرض ۱۱۵۲۰۰ بیت در ثانیه می باشد. فریم ها ۸ بیتی هستند و یک بیت هم برای توقف و نوع توازن هم غیرفعال است. ماژول سیم ۹۰۰ دستورات را کارکتر به کاراکتر دریافت یا ارسال می کند .

ارتباط بین ماژول SIM900 و میکروکنترلر یا کامپیوتر از طریق دستورات AT Command می باشد. ما صرفا به برخی از این دستورات اشاره می کنیم. ما با استفاده از صدور این دستورات از میکرو کنترلر به ماژول می توانیم پیامک های موجود در حافظه ماژول را بخوانیم (بصورت کاراکتر به کاراکتر) ، و یا سرعت تبادل اطلاعات را تعیین کنیم، پیامک ارسال کنیم و ....

**نکته:** پس از اتمام دستورات ارسالی از میکروکنترلر به ماژول باید کاراکتر کد ۱۳ یا همان ENTER یا return\_carriage هم زده شود. و همچنین در ارسال پیامک باید پایان پیامک به ترکیب کلیدهای ctrl+z یا همان کد ۲۶ ختم شود و شماره موبایل مقصد باید در دو تا دابل کوتیشن یا کد ۳۴ قرار گیرد.

**نکته:** اطلاعاتی دریافتی از ماژول (ارسال از ماژول به سمت میکرو کنترلر) دارای قالب زیر می باشند:

```
<cr><lf>RESPONSE<cr><lf>
```

کداسکی چند کاراکتر مورد نیاز:

None breaking space=Ascci code 255 = space

Double quotation = Ascci code 34 = "

Substitute character = Ascci code 26 = Control + Z keys

Carriage return = Ascci code 13 = Enter key

Line Feed = Ascci code 10

Null character = Ascci code 0

معادل برخی از آنها در زبان سی : \r کلید اینتر \n تعویض خط (لاین فید) "نقل قول دوتایی \0 کاراکتر نال

توضیحات	دستور AT Command
<p>فعال کردن echo یعنی مودم دستورات رسیده از DTE را مجدداً به پورت سریال می فرستد مثلاً جهت نمایش بر روی صفحه نمایش</p> <p>غیرفعال کردن echo یعنی مودم دستورات رسیده از DTE را به پورت سریال اکو نمی کند. برای ارسال پیامک باید از این حالت استفاده کرد.</p>	<p>ATE1</p> <p>ATE0</p>
<p>نرخ باد ریت (نرخ علامت در ثانیه) مثلاً ۹۶۰۰ یا ۱۱۵۲۰۰ و ...</p> <p>البته خود سیم ۹۰۰ بطور اتوماتیک می تواند این نرخ را تشخیص دهد</p>	<p>AT+IPR=x</p>
<p>حالت متنی text mode فعال می شود یعنی محتوای پیامک ها بصورت کاراکترهای متنی قابل خواندن هستند. حداکثر ۱۴۰ حرف با کد اسکی ۸ بیتی و برای کاراکترهای فارسی ۷۰ حرف</p> <p>حالت packet data فعال می شود یعنی محتوای پیامک ها بصورت کدهای هگزادسیمال هستند</p>	<p>AT+CMGF=1</p> <p>AT+CMGF=0</p>
<p>خواندن یک پیامک از صندوق ورودی از حافظه شماره x</p> <p>مثلاً AT+CMGR=1 خواندن اولین پیامک در حافظه جاری</p> <pre>char enter_code=13; printf("AT+CMGR=1%c",enter_code);</pre> <p>دریافت کاراکتر به کاراکتر پیامک...</p> <p>یا</p> <pre>printf("AT+CMGR=1\r");</pre> <p>دریافت کاراکتر به کاراکتر پیامک...</p>	<p>AT+CMGR=x</p>
<p>فرستادن پیامک به یک شماره خاص</p> <p>مثال</p> <pre>char dq=34,enter_code=13,cz=26; printf("AT+CMGS=%c0918928xxxx%c%c",dq,dq,enter_code); delay_ms(100); printf("this sms send by SIM900 %c",cz);</pre> <p>یا</p> <pre>printf("AT+CMGS=\r0918928xxxx\r"); delay_ms(100); printf("this sms send by SIM900 %c",26);</pre>	<p>AT+CMGS=</p> <p>دابل کوتیشن یا کد ۳۴</p> <p>0918xxxxxx</p> <p>دابل کوتیشن یا کد ۳۴</p> <p>کلید Enter یا کد ۱۳</p> <p>محتوای پیامک مورد نظر</p> <p>ترکیب کلیدهای Ctrl+Z</p> <p>یا کد ۲۶</p>

<p style="text-align: center;"><b>ذخیره پیامک در حافظه</b></p> <p style="text-align: center;">مثال</p> <p>AT+CMGW="+85291234567",145,"STO UNSENT"&lt;CR&gt;This is a test for store a message.&lt;Ctrl+z&gt; باز گرداندن مکان ذخیره شده پیامک توسط مازول ، مثلا ۱ +CMGW:1</p> <p style="text-align: center;">۱۴۵ یعنی شماره بین المللی است همراه با علامت + ۱۲۹ یعنی مشخص نیست که شماره بین المللی است یا انواع دیگر</p>	<p style="text-align: center;">AT+CMGW کلید Enter یا کد ۱۳ محتوای پیامک مورد نظر ترکیب کلیدهای Ctrl+Z یا کد ۲۶</p>
<p style="text-align: center;"><b>حذف پیامک ها با استفاده از پارامترهای زیر</b></p> <p style="text-align: center;">حذف خوانده شده ها حذف خوانده نشده ها حذف فرستاده شده ها حذف فرستاده نشده ها حذف پیامک های ورودی حذف همه پیامک ها</p> <p style="text-align: center;">مثال: حذف همه پیام ها بصورت زیر می باشد</p> <pre>printf("AT+CMGDA=%cDEL ALL%c%c",34,34,13);</pre> <p>یا</p> <pre>printf("AT+CMGDA=\\"DEL ALL\\"r");</pre> <p style="text-align: center;"><b>حذف پیامک رسیده در حافظه شماره یک</b></p> <pre>Printf("AT+CMGDA=1r");</pre>	<p style="text-align: center;">AT+CMGDA= "DEL READ"  "DEL UNREAD"  "DEL SENT"  "DEL UNSENT"  "DEL INBOX"  "DEL ALL" کد اینتر یا ۱۳</p> <p style="text-align: center;">AT+CMGDA=1 کد اینتر یا ۱۳</p>

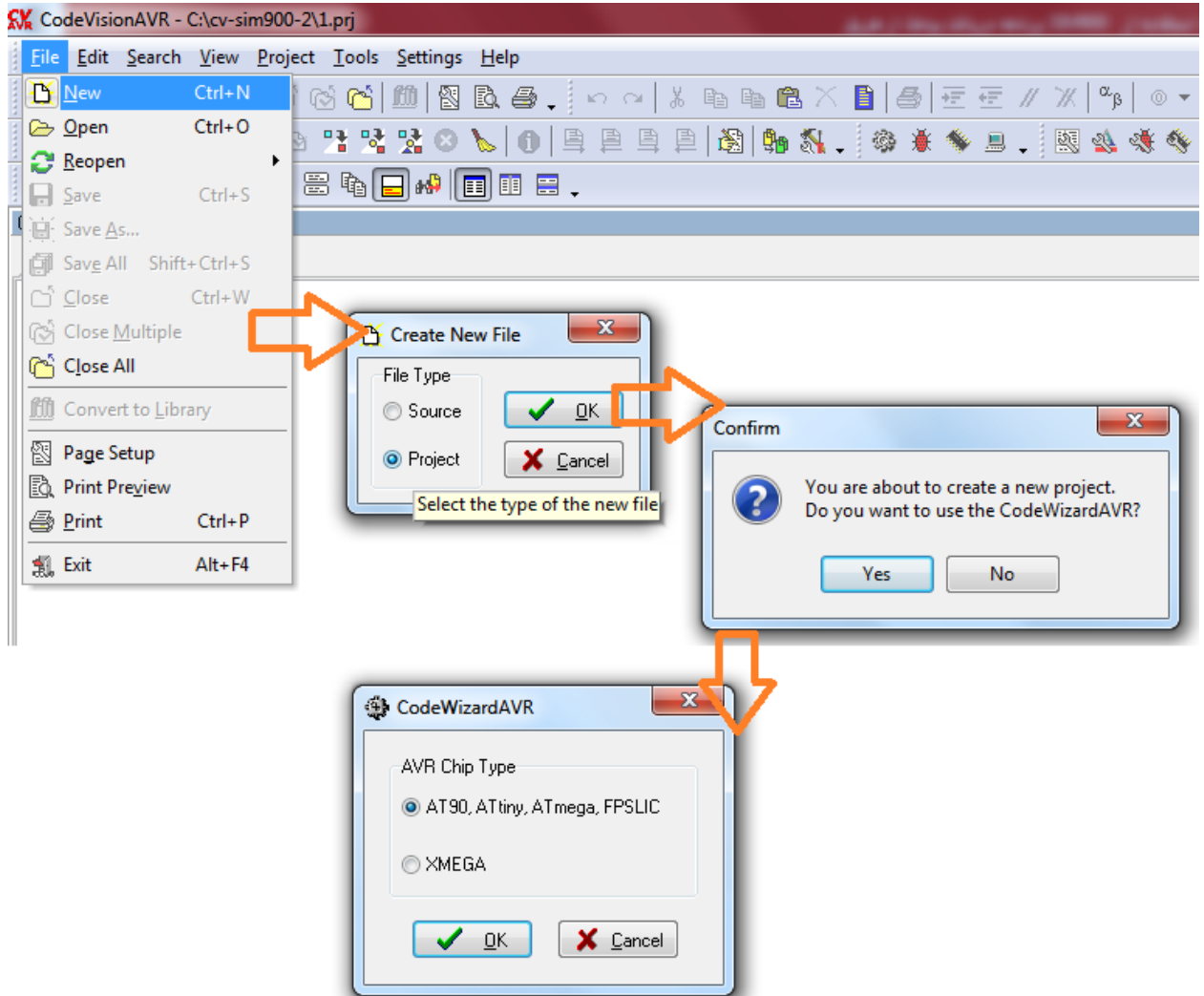
## نرم افزار پروژه و برنامه نویسی

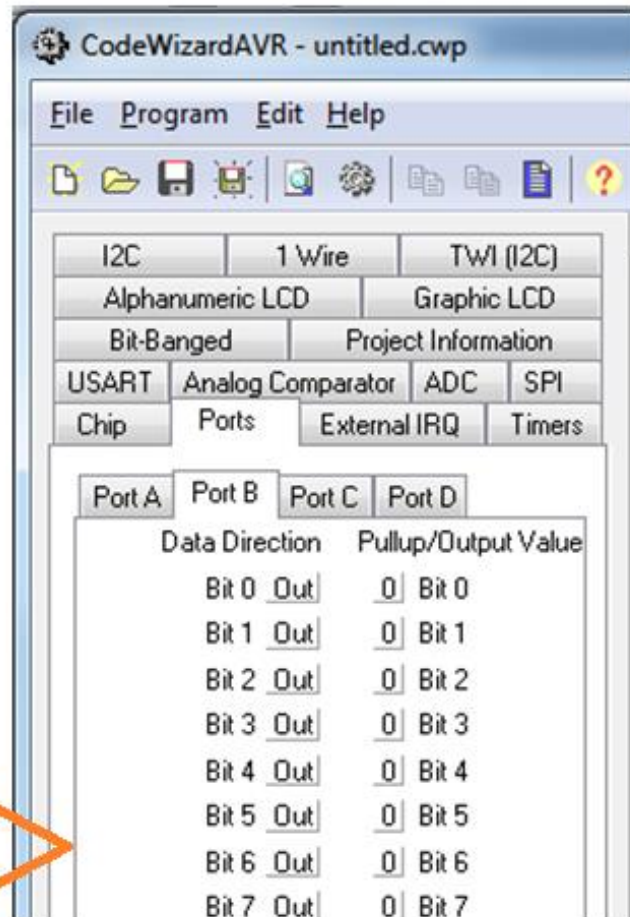
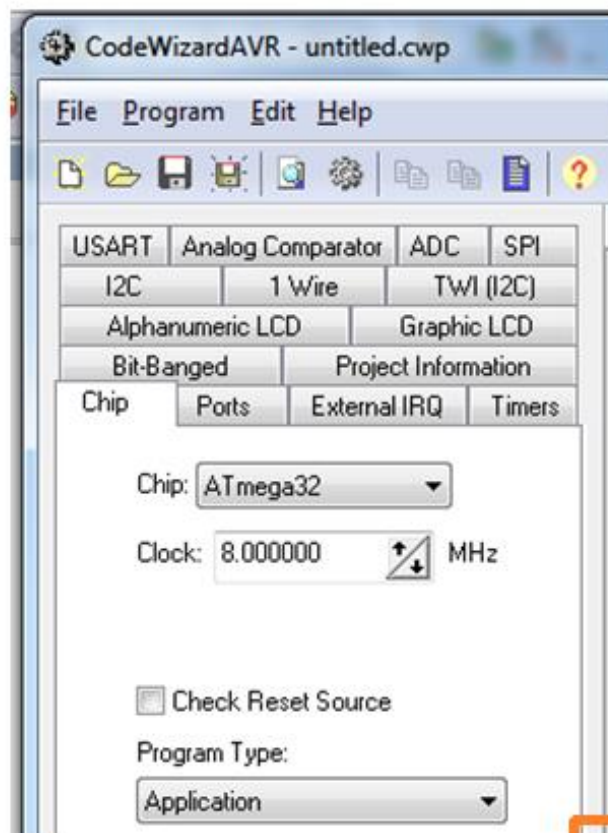
نصب نرم افزار CodeVisionAVR یا نرم افزارهای مشابه جهت نوشتن کدهای برنامه نویسی مربوط به پروژه ، ضروری می باشد.

بطور خلاصه، برنامه ما ابتدا مائول سیم ۹۰۰ را با تنظیمات لازم راه اندازی می کند و همه پیامک های قبلی را از حافظه حذف می نماید و سپس بطور پیوسته پیامک های رسیده (در حافظه شماره ۱) را می خواند و شماره موبایل فرستنده را چک می کند، اگر همان شماره موبایل مورد نظر ما بود آنگاه بررسی می کند که محتوای پیامک چه متنی است و در واقع چه دستوری را باید اجرا کند، و سپس پیامک را از حافظه حذف می کند .

مثلا اگر پیامک شامل متن start-motor باشد، موتور مربوطه را راه اندازی می کند و یک ال ای دی مشخص را نیز روشن می نماید و بر روی ال سی دی پیام روشن شدن موتور را نمایش می دهد و از طرفی یک پیامک برای مدیر مربوطه می فرستد به این عنوان که دستور مورد نظر شما انجام داده شد و یا دستور اشتباه است و ... .  
مراحل انجام کار و کد برنامه به زبان سی ، در ادامه آمده است.

در کدویژن ای وی آر(من از نسخه 2.05.3 استفاده کردم) یک پروژه جدید ایجاد می کنیم و میکرو کنترلر مگا۳۲ را انتخاب می کنیم و تنظیمات ورودی و خروجی پورت ها را با توجه به پروژه خود انجام می دهیم.

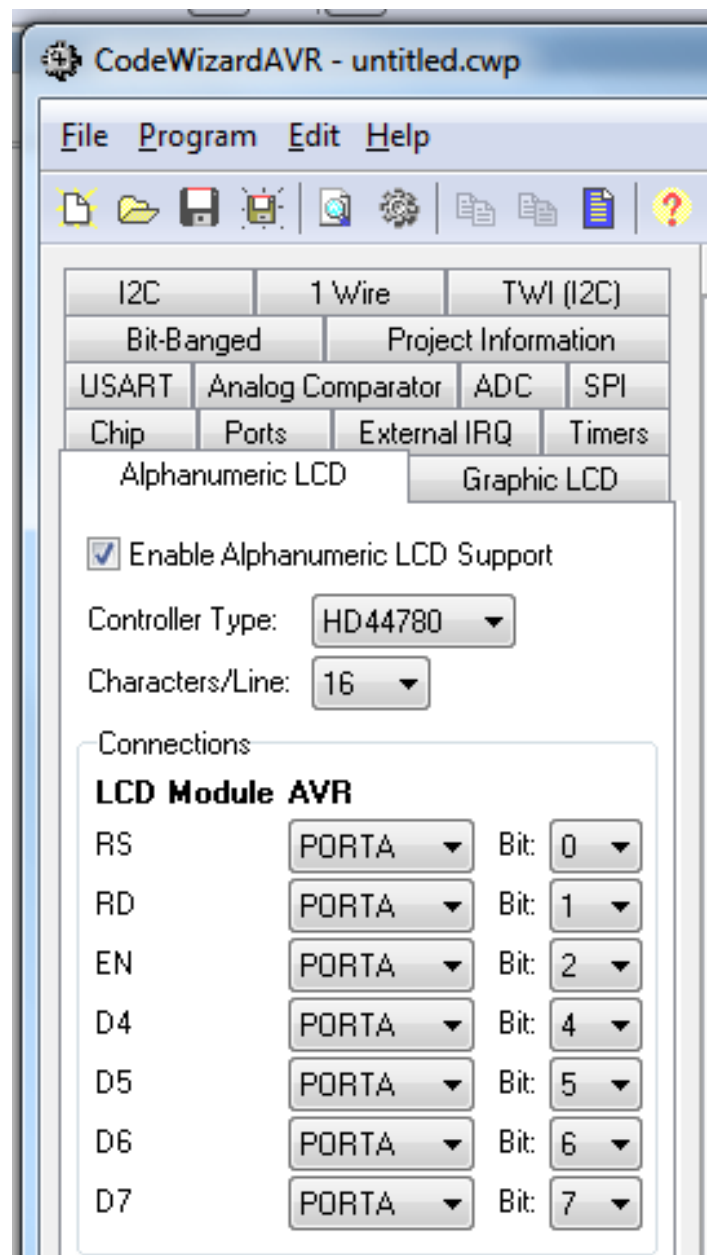




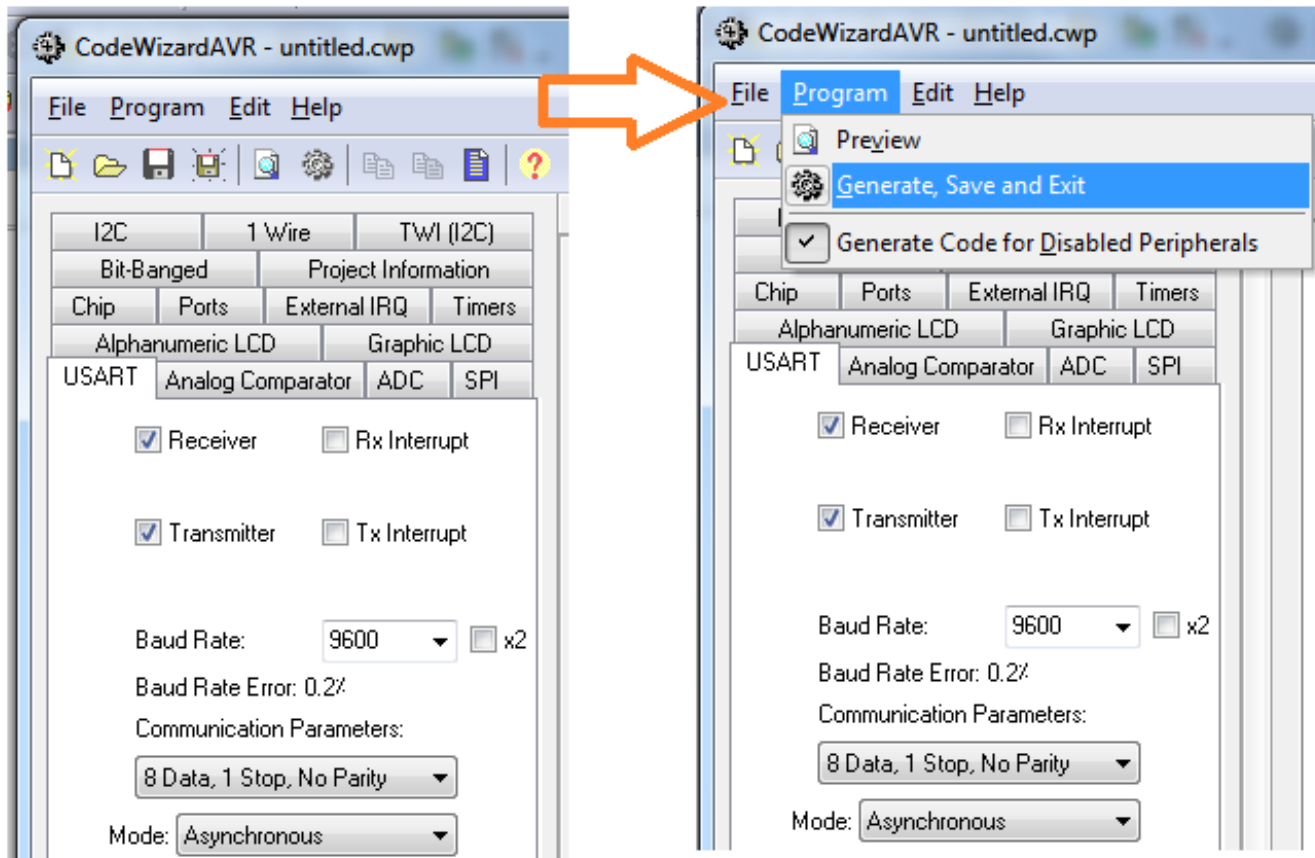
فرض کنید که تو وسط پورت B در حال کنترل چند وسیله هستی مثلاً خاموش و روشن کردن یک LED با پورت B0 و قطع و وصل برق یک موتور با پورت B7 و الا آخر. پس پورت B را باید به عنوان خروجی تعریف کنیم.



برای نمایش پیام دریافتی روی یک LCD شانزده ستونه دوخطه، LCD را به پورت A وصل می کنیم. پس باید در قسمت LCD پورت A انتخاب کنیم. دقت کنید که باید پایه های مربوطه را طبق آنچه که کدویژن برایتان مشخص کرده ، به میکروکنترلر وصل کنید. هرچند میتوان آنها را در همان پنجره مربوطه به صورت دلخواه تغییر داد. همچنین دقت کنید در ورژن های جدید کدویژن دو تب برای LCD وجود دارد (Alphanumeric LCD, Graphical LCD) و در این مثال ما از ال سی دی الفبایی\_عددی شانزده ستونه استفاده می کنیم.



همچنین برای ارتباط این میکروکنترلر با SIM900 نیاز داریم که ارتباط سریال را در میکروکنترلر مگا ۳۲ فعال کنیم و تنظیمات ارسال و دریافت داده را به شیوه زیر انجام دهیم. و سپس پروژه را ایجاد می کنیم.



پس از ذخیره پروژه ، کدویژن یک سری کد را بطور خودکار به برنامه اضافه می کند و ما باید با توجه به نیازمان کدهای مربوط به پروژه مان را اضافه کنیم. در نهایت برنامه به شکل زیر خواهد بود.



C:\us\robot\daneshgah\cv-sim900-2\cv-atmega32-sim900.c

Notes cv-atmega32-sim900.c

```

1
2 #include <mega32.h>
3 #include <alcd.h>
4 #include <stdio.h>
5 #include <delay.h>
6 #include <string.h>
7
8 void main(void)
9 {
10 ///////////////////////////////////////////////////
11 //*****DECLARE VARIABLES*****
12 char enter_code=13;// carriage return = \r
13 char linefeed_code=10;//line feed = \n
14 char null_code=0;//null character = \0
15 char double_quote_code=34;//double quotation = \"
16 char control_z_code=26;//Control+Z keys
17 char message[250];
18 char character;
19 int i;
20 unsigned char *find_substring;//a pointer to found sub string in main string of SMS
21 int end_of_message;
22 int command_done=0;
23 //*****
24
25 PORTA=0x00; DDRA=0x00; PORTB=0x00; DDRB=0xFF;
26 PORTC=0x00; DDRC=0x00; PORTD=0x00; DDRD=0x00;
27
28 TCCR0=0x00; TCNT0=0x00; OCR0=0x00;
29
30 TCCR1A=0x00; TCCR1B=0x00; TCNT1H=0x00; TCNT1L=0x00; ICR1H=0x00; ICR1L=0x00;
31 OCR1AH=0x00; OCR1AL=0x00; OCR1BH=0x00; OCR1BL=0x00;
32
33 ASSR=0x00; TCCR2=0x00; TCNT2=0x00; OCR2=0x00;
34
35 MCUCR=0x00; MCUCSR=0x00;
36
37
38 TIMSK=0x00;
39
40 UCSRA=0x00; UCSRB=0x18; UCSRC=0x86; UBRRH=0x00; UBRL=0x33;
41
42 ACSR=0x80; SFIOR=0x00; ADCSRA=0x00; SPCR=0x00; TWCR=0x00;
43
..

```

```

44
45 /////////////// USER CODES //////////////////////
46
47 //----- LCD initializing and clearing ----
48 lcd_init(16);
49 lcd_clear();
50 lcd_gotoxy(0,0);
51 lcd_puts("Please wait ...");
52 //-----
53
54
55 //===== Initializing and setting SIM900 =====
56 printf("AT\r"); //initializing GSM Modem SIM900
57 delay_ms(1000);
58 printf("ATE0\r"); //Disable echo to terminal
59 delay_ms(500);
60 printf("AT+CMGF=1\r"); //Enable text mode
61 delay_ms(500);
62 printf("AT&W\r"); //Save setting...
63 delay_ms(500);
64 //=====
65
66 lcd_clear();
67 lcd_gotoxy(0,0);
68 lcd_puts("SIM900 IS READEY...");
69
70 //delete all previous messages
71 printf("AT+CMGDA=\"DEL ALL\"\r");
72 delay_ms(500);
73
74 while (1)
75 {
76
77 //***** Array message[] = Null *****
78 for(i=0;i<250;i++)
79 {
80     message[i]=null_code; //null=\0
81 } //end of for
82
83 //*****
84
85     command_done=0;
86

```

```

87
88 //*****
89 //***** Read one SMS character by character *****
90 i=0;
91 end_of_message=0;//FALSE
92 printf("AT+CMGR=1\r");//read a sms in memory[1] of SIM900 MODULE
93 //Read a sms character by character...
94
95 while(end_of_message==0)
96 {
97     character=getchar();
98     if(character==null_code){ //Null character \0
99         //do nothing
100     }
101     else if( (character==enter_code)|| (character==linefeed_code) )&&(message[0]!=null_code)
102     {
103         end_of_message=1;//end_of_message = true
104     }
105     else
106     {
107         if(i>=250){//array message[] is full
108             //do not save rest of sms in array message[]
109         }
110         else{ message[i]=character;
111             i++;
112         }
113     }
114 }
115
116 }//end of while (end_of_message==0)
117
118
119 printf("AT+CMGDA=1\r");//delete sms in memory[1]
120 delay_ms(100);
121 //*****
122 //*****
123
124

```

```

125 //-----
126 //----- if this SMS is from a specific number or not ??? -----
127 //---- function strstr search "0918xxxxxxx" in message ,
128 //---- if find it then returns a pointer to the found substring else returns Null
129
130 find_substring=strstr(message,"0918xxxxxxx");
131 if(find_substring!=0){
132     find_substring=strstr(message,"start-motor");
133     if(find_substring!=0){
134         PORTD.7=1;//START THE MOTOR
135         PORTB.0=1;//LED LIGHTS UP
136         lcd_clear();
137         lcd_gotoxy(0,0);//column 0 , row 0
138         lcd_putsf(" command is : ");
139         lcd_gotoxy(0,1);//column 0 , row 1
140         lcd_putsf("start-motor");
141         command_done=1;//command done =true
142     }//end of if
143
144     find_substring=strstr(message,"turn-off-motor");
145     if(find_substring!=0){
146         PORTD.7=0;//TURN OFF THE MOTOR
147         PORTB.0=0;//TURN OFF LED
148         lcd_clear();
149         lcd_gotoxy(0,0);//column 0 , row 0
150         lcd_putsf(" command is : ");
151         lcd_gotoxy(0,1);//column 0 , row 1
152         lcd_putsf("turn-off-motor");
153         command_done=1;//command done =true
154     }//end of if
155
156
157     //....you can add your specific commands ...
158
159
160     //send sms for manager
161     if(command_done==1){
162         printf("AT+CMGS=\"0912xxxxxxx\"\r");
163         delay_ms(100);
164         printf("your command done!%c",control_z_code);
165         delay_ms(100);
166     }
167     else {
168         printf("AT+CMGS=\"0912xxxxxxx\"\r");
169         delay_ms(100);
170         printf("your command is incorrect%c",control_z_code);
171         delay_ms(100);
172     }
173
174
175     }//end of if
176
177     else {
178         //message is not from our specific number
179         //delete all read messages
180         printf("AT+CMGDA=\"DEL READ\"\r");
181         delay_ms(1000);
182     }
183
184 //-----
185 //-----
186
187
188
189 }//end of while 1
190
191
192 }//end of program
193

```

منابع:

۱- دهقان ، علی ، آشنایی با ماژول SIM900 ، [www.electro2014.ir](http://www.electro2014.ir) ، ۱۳۹۳/۰۸/۱۴ ،

۲- رحیمی ، مرتضی، آموزش GSM (ارتباط SIM300cz با میکروکنترلر ATmega8)،

[www.mrkelectronic.blogfa.com](http://www.mrkelectronic.blogfa.com)

۳- <http://www.developershome.com>

۴- <http://www.eca.ir>

۵- <http://www.diafaan.com>