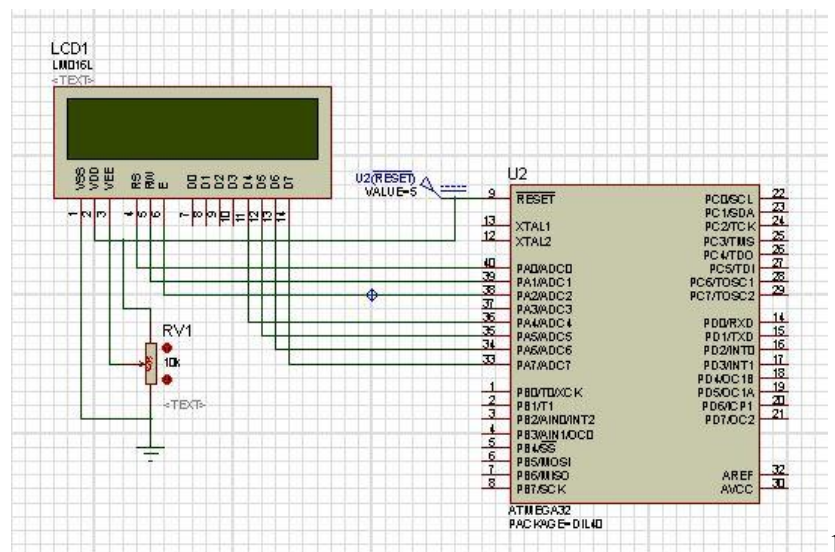


## آزمایش ۱: نمایشگر LCD

### ۱-۱ مقدمه

هدف از این آزمایش راه اندازی نمایشگر متنی می باشد. نمایشگر استفاده شده یک LCD 2×16 بوده که پیکربندی اتصال آن به میکرو در شکل ۱-۱ نشان داده شده است. در شکل ۱-۱ از پایه های پورت A میکرو برای راه اندازی نمایشگر استفاده شده است که می تواند در برنامه نوشته شده توسط هر گروه تغییر داده شود. نمایشگرهای 2\*16 دارای ۱۶ پایه می باشند..



شکل ۱-۱: شماتیک مدار راه انداز نمایشگر متنی.



پایه شماره ۱ و ۲ به ترتیب پایه های منفی و مثبت تغذیه داخلی نمایشگر است که باید به زمین و ۵ ولت تغذیه وصل گردند. پایه شماره ۳ پایه کنترل درخشندگی نمایشگر می باشد که بین ۰ ولت (حداکثر درخشندگی) و ۵ ولت (حداقل درخشندگی) متغیر می باشد. این پایه باید با استفاده از یک پتانسیومتر به ولتاژ مورد نظر وصل گردد. پایه های ۱۵ و ۱۶ به ترتیب پایه های مثبت و منفی تغذیه LED پس زمینه نمایشگر می باشند که اگر بخواهیم نمایشگر به صورت دید در شب کار کند باید آن را روشن نماییم.

پایه های ۷ تا ۱۴ (DB0 تا DB7) پایه های داده نمایشگر می باشد. کنترلر داخلی نمایشگر از طریق این ۷ پایه با میکروکنترلر تبادل داده می کند. به عبارتی تمامی تبادل داده ای که بین نمایشگر و میکروکنترلر انجام می شود با استفاده از این پایه ها می باشد. داده های دودویی که روی این پایه ها قرار می گیرند ممکن است به دو گونه تعبیر شوند: ۱- به صورت دستورالعمل؛ مانند پاک کردن صفحه نمایشگر، روشن کردن نمایشگر، انتقال مکان نما به مکان خاص و غیره، ۲- داده؛ یعنی همان کاراکترهایی که قرار است نمایش داده شوند. تمایز بین این دو حالت هم توسط پایه شماره ۴ تشخیص داده می شود. پایه شماره ۴ پایه انتخاب رجیستر (RS) می باشد. توضیح اینکه کنترلر داخلی این نوع نمایشگرها دارای دو رجیستر داخلی می باشد: رجیستر دستورالعمل (IR) یا Instruction Register برای ذخیره سازی موقت دستورالعمل هایی که به نمایشگر وارد می شوند و رجیستر داده (DR) یا Data Register برای ذخیره کردن موقت داده هایی که برای نمایش به نمایشگر وارد می شوند. حال اگر پایه RS به زمین وصل شده باشد پایه های ۷ تا ۱۴ به عنوان دستورالعمل تلقی شده و دستورالعمل ورودی به صورت موقت

برای اجراشدن داخل رجیستر IR ذخیره می شود اما اگر RS به یک منطقی وصل شده باشد این پایه ها به عنوان داده تلقی شده و داده ورودی به صورت موقت در داخل رجیستر DR ذخیره می شود.

پایه ۵ پایه انتخاب خواندن/نوشتن (R/W) می باشد. کنترل داخلی نمایشگر دارای انواع مختلف از حافظه های داخلی می باشد که در بخش های بعد با آن ها آشنا می شویم. کارکرد پایه ۵ این می باشد که ما با استفاده از آن انتخاب می کنیم که می خواهیم روی حافظه داخلی نمایشگر بنویسیم ( $R/W=0$ ) و یا از روی آن بخوانیم ( $R/W=1$ ).

وقتی داده روی پایه های ۷ تا ۱۴ قرار گرفت و با پایه RS رجیستر مورد نظر را انتخاب کردیم نمایشگر از این وضعیت با خبر شده و بداند که چه زمانی اطلاعات قابل برداشتن از این پایه های می باشد. این کار توسط شماره ۶ (E) انجام می شود. نمایشگر در لحظه ای که یک لبه پایین رونده روی این پایه ایجاد می شود داده روی پایه های ۷ تا ۱۴ را می خواند و روی رجیستر IR یا DR ذخیره می کند. در اکثر کارکردهای معمولی هدف نوشتن اطلاعات روی حافظه نمایشگر جهت نمایش روی آن می باشد کارکردهای خاصی وجود دارند که لازم باشد از روی حافظه نمایشگر داده ای بخوانیم. بنابراین در کارکردهای معمولی می توان این پایه را مستقیم به زمین وصل کرده و به میکرو متصل نکرد.

برای کار با LCD نیاز به فراخوانی کتابخانه مربوطه و استفاده از دستورات موجود می باشد. دستورات کار با LCD در ادامه در چند آزمایش مختلف بررسی خواهند شد.

## ۲-۱ مراحل اجرای آزمایش

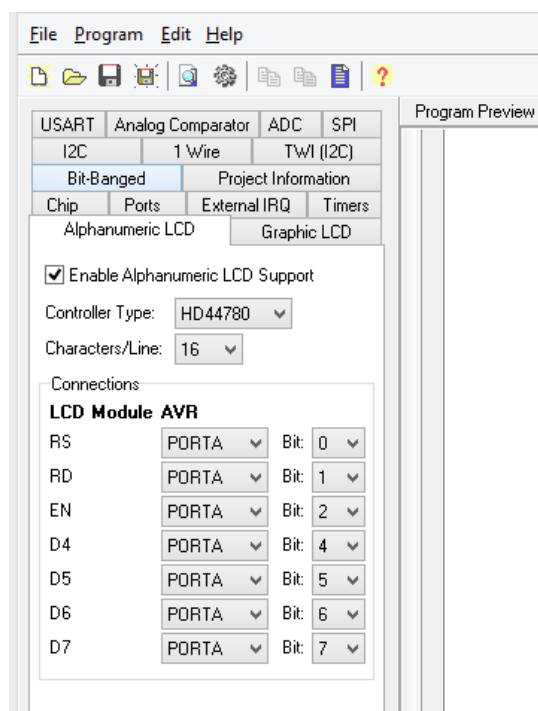
در ادامه متداولترین دستورات کار با LCD بررسی خواهند شد.

### ۱-۲-۱ چاپ تک کارکتر

این دستور یک کاراکتر را بر روی LCD چاپ می نماید. دقت شود که داده ورودی به این دستور باید از نوع کارکتر بوده و موقع چاپ کد اسکی آن چاپ خواهد شد. یک پروژه جدید بدون استفاده از CodeWizard در نرم افزار ایجاد نموده و برنامه نوشته شده در زیر را به آن اضافه نمایید. سپس در منوی project به بخش Configure رفته و مطابق با شکل ۲-۱ تنظیمات مربوط به اتصالات LCD به میکرو انجام دهید.

```
#include <mega32.h>
#include <delay.h>
#include <alcd.h>
char Data;
void main(){
    lcd_init(16);
    lcd_clear();
    while(1){
        lcd_gotoxy(0,0);
        lcd_putchar(Data+48);
        if (Data == 9) Data = 0;
        else Data++;
        delay_ms(1000); }}

```



شکل ۱-۲: تنظیمات LCD.

پس از اجرای برنامه، فایل Hex نهایی را بر روی برد آزمایشگاه پروگرام نمایید. دیده خواهد شد که اعداد ۰ تا ۹ بر روی نمایشگر چاپ می‌شوند.

## ۱-۲-۲ چاپ داده با استفاده از دستورات تبدیل

مطابق مرحله قبل یک پروژه جدید ایجاد نموده و برنامه زیر را به آن اضافه نمایید. در این برنامه از دستور `sprintf` برای تبدیل رشته، و از دستور `lcd_puts` برای چاپ رشته تولید شده استفاده شده است. دستور `sprintf`

داده ورودی را بر اساس فرمت انتخابی تبدیل به یک رشته نموده و در آرایه کارکتری ذخیره می‌نماید. این آرایه در SRAM میکرو ذخیره شده و توسط دستور lcd\_puts می‌تواند بر روی LCD چاپ شود.

```
#include <mega32.h>
#include <delay.h>
#include <alcd.h>
#include <stdio.h>
char Data;
char Buf[32];
void main(){
    lcd_init(16);
    lcd_clear();
    lcd_puts("Start");
    delay_ms(5000);
    lcd_clear();
    while(1){
        sprintf(Buf,"%d",Data);
        lcd_puts("Counting");
        lcd_gotoxy(0,1);
        lcd_puts(Buf);
        Data++;
        delay_ms(1000);
        lcd_clear();}}
```

فایل Hex نهایی را بر روی برد آزمایشگاه پروگرام نموده و نتایج را مشاهده نمایید. دیده خواهد شد که در خط دوم LCD، داده به فرم دسیمال چاپ می‌شود.

علاوه بر دستور lcd\_puts دستور lcd\_putsf نیز وجود دارد که رشته ذخیره شده در حافظه FLASH میکرو را بر روی LCD چاپ می‌نماید.

تمرین ۱: با استفاده از کدهای بخش‌های قبل یک counter صعودی /نزولی با دقت یک صدم ثانیه بسازید که توسط یک ورودی نوع آن تعیین می‌شود.

تمرین ۲: برنامه‌ای نویسید که عدد counter صعودی از سمت چپ LCD شروع و در هر ثانیه یک واحد به راست جابجا شود و پس از رسیدن به انتها به صورت معکوس (نزولی و از راست به چپ) عمل کند.