

## به نام خدا

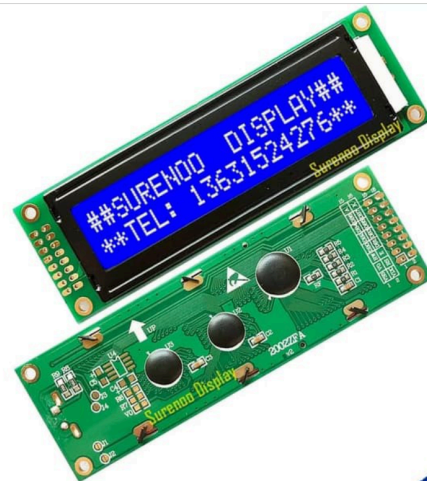
## آشنایی با LCD

صفحه نمایش کریستال مایع و یا همان LCD ها ابزاری برای نمایش اطلاعات می باشند که بر خلاف 7Seg ها که فقط توانایی نمایش کاراکتر های خاصی را دارند، می توانند هر نوع کاراکتری، شامل حروف، اعداد و علائم و یا حتی کاراکترهای تعریف شده توسط کاربر را نمایش دهند. البته این نوع تنوع نمایش باعث شده که کارکردن با LCD ها کمی پیچیده تر از 7Seg شود و به همین دلیل در مداراتی که شامل میکرو کنترلر نباشد، امکان استفاده از LCD سخت و محدود می باشد.

چیزی که از آن بعنوان LCD یاد می شود درواقع یک صفحه نمایشگر LCD مانند صفحه ماشین حساب است که با آی سی کنترلر، مدارهای جانبی و عموماً یک لامپ پشت صفحه در یک بسته پیش ساخته همراه شده است.

## LCD2

LM032L



شکل 1. نمونه ای از LCD

در شکل بالا یک نمونه LCD نشان داده شده است. این LCD از نوع ۲۰ کاراکتری و دو خطی است (دو خط برای نمایش دارد). همان طور که در شکل سمت راست ملاحظه می شود، LCD مجهز به یک IC کنترلر است که در واقع یک CPU کوچک است که کلیه کارهای مربوط به روشن و خاموش شدن پیکسل ها را کنترل می کند و برای نمایش یک متن فقط لازم است که از طریق خط فرمان این CPU کاراکتر هایی که باید نمایش داده شود به شکل رشته ای از حروف داده شود.

معمولاً پایه های LCD ها که درواقع راه حل تبادل داده با CPU مربوط به LCD است به شکل زیر تعریف شده است

## ■ پایه های تغذیه

1.  $V_{CC}$  ( $V_{DD}$ ) : تغذیه مثبت
2.  $GND$  ( $V_{SS}$ ) : زمین
3.  $V_O$  ( $V_{EE}$ ) : درخشندگی

4. LED<sup>+</sup> و LED<sup>-</sup>: تغذیه پس زمینه

■ پایه های کنترلی

5. EN: با لبه منفي اطلاعات خوانده مي شود

6. R/W: صفر يعني نوشتن و يك يعني خواندن

7. RS: صفر يعني ثبات دستور و يك يعني ثبات داده

■ گذرگاه داده

8. DB0 ~ DB7

از طريق پایه های LCD می توان دستورات متفاوتی را به LCD وارد کرد. لیست این دستورات نیز در جدول يك نشان داده است.

جدول 1- دستورات LCD

Instruction	Code										Execution Time (max.) fosc = 250KHz
	R S	R/ W	D B7	D B6	D B5	D B4	D B3	D B2	D B1	D B0	
Clear Display	0	0	0	0	0	0	0	0	0	1	1.64 mS
Cursor At Home	0	0	0	0	0	0	0	0	1	*	1.64 mS
Entry Mode Set	0	0	0	0	0	0	0	1	I/ D	Sh	40 μS
Display On/Off Control	0	0	0	0	0	0	1	D	C	B	40 μS
Cursor/Display Shift	0	0	0	0	0	1	S/C	R/ L	*	*	40 μS
Function Set	0	0	0	0	1	D L	N	F	*	*	40 μS
CGRAM Address Set	0	0	0	1	ACG						40 μS

DDRAM Address Set	0	0	1	ADD	40 $\mu$ S
Busy Flag/ Address Read	0	1	BF	AC	0 $\mu$ S
CGRAM/ DDRAM Data Write	1	0		WRITE DATA	40 $\mu$ S
CGRAM/ DDRAM Data Read	1	1		READ DATA	40 $\mu$ S

## Set Cursor Move Direction:

I/D - Increment the Cursor After Each Byte Written to Display if Set  
Sh - Shift Display when Byte Written to Display

## Set Interface Length

DL - Set Data Interface Length 8(1)/4(0)  
N - Number of Display Lines 1(0)/2(1)  
F - Character Font 5x10(1)/5x7(0)

## Enable Display/Cursor

D - Turn Display On(1)/Off(0)  
C - Turn Cursor On(1)/Off(0)  
B - Cursor Blink On(1)/Off(0)

## Poll the "Busy Flag"

BF - This bit is set while the LCD is processing

## Move Cursor to CGRAM/Display

A - Address

## Move Cursor/Shift Display

SC - Display Shift On(1)/Off(0)  
RL - Direction of Shift Right(1)/Left(0)

## Read/Write ASCII to the Display

D - Data

## مراحل کار

۱. در این قسمت برنامه معروف Hello Word را روی LCD اجرا می کنید. برای این کار می توانید از کتاب خانه LCD که کلیه فرمان های LCD را داخل خود دارد و از طریق توابع مناسب امکان نمایش متن را میسر می کند استفاده کنید. پروژه خود را به عنوان part1 در پوشه ای به همین نام در گیت ذخیره کنید.

۲. با استفاده از تایمر دو یک ساعت دیجیتالی طراحی کنید. ساعت شما باید نمایش ساعت و دقیقه و ثانیه را داشته باشد و به شکل خیلی دقیق (بدون استفاده از تابع delay) کار کند. اضافه کردن دکه های تنظیم ساعت ضرورتی ندارد و در صورتی که این کار را انجام دهید به عنوان پروژه اضافه در نظر گرفته می شود. پروژه خود را به عنوان part2 در پوشه ای به همین نام در گیت ذخیره کنید.

۳. با استفاده از تایمر یک، ساختار آلارم را در ساعت خود پیاده سازی کنید. آلارم در واقع یک زمان سنج است که پس از طی یک دوره زمانی که کاربر مشخص می کند یک متن را روی LCD نمایش می دهد و یک LED را روشن می کند. آلارم شما باید قابلیت تنظیم داشته باشد. پروژه خود را به عنوان part3 در پوشه ای به همین نام در گیت ذخیره کنید.

برای هر دو بخش بالا یک فایل پروتئوس و یک فایل hex و همچنین برنامه C را ایجاد کنید و روی گیت به شکل جداگانه قرار دهید.