

آموزش استفاده از ماژول های برنامه نویسی

(ماژول CHARACTER_LCD)

آخرین تغییرات :

Sunday, January 13, 2019

نگارش و گردآوری : مجید درهم بخش (Majid Derhambakhsh)

نگارنده کد و پشتیبان : مجید درهم بخش (Majid Derhambakhsh)

[E-mail](#)

فهرست

۳	راهنمای استفاده از ماژول CHARACTER_LCD
	تنظیمات LCD
	تابع Lcd_Init
	تابع Lcd_SendCommand
	تابع Lcd_PutChar
	تابع Lcd_PutString
	تابع Lcd_Clear
	تابع Lcd_GotoXY
	تابع Lcd_ShiftToLeft
	تابع Lcd_ShiftToRight
۵	نیازمندی ها
۶	نکات مهم
۷	مثال های استفاده از ماژول
۸	میکرکنترلر های پشتیبانی شده
۹	تغییرات نسخه

راهنمای استفاده از ماژول CHARACTER_LCD

در مرحله نخست برای استفاده از این ماژول باید طبق دستور زیر آن را به برنامه اضافه کنید:

```
#include "location/CHARACTER_LCD.h"
```

جهت استفاده از LCD ابتدا باید آن را تنظیم نمایید. برای اینکار کفایت به روش زیر عمل کنید:

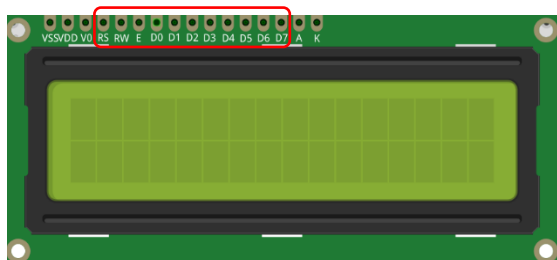
Lcd_Config.x

x : ویژگی مورد نظر و دارای بخش های زیر است

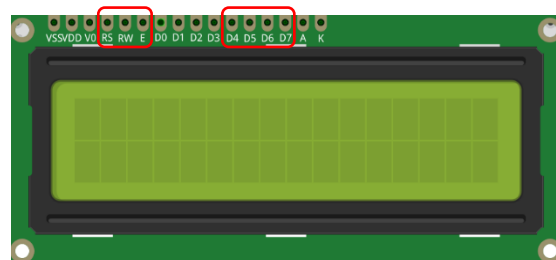
Mode : جهت تنظیم حالت کاری می باشد و دارای آرگومان های زیر است.

_4BIT_INTERFACE : جهت انتخاب مود ۴ بیتی.

_8BIT_INTERFACE : جهت انتخاب مود ۸ بیتی.



پایه های مورد نیاز در مود ۸ بیتی



پایه های مورد نیاز در مود ۴ بیتی

Font : جهت تنظیم فونت می باشد و دارای آرگومان های زیر است.

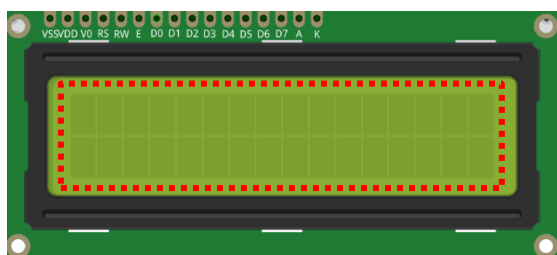
_FONT_5X7 : جهت انتخاب فونت ۵ * ۷ .

_FONT_5X10 : جهت انتخاب فونت ۵ * ۱۰ .

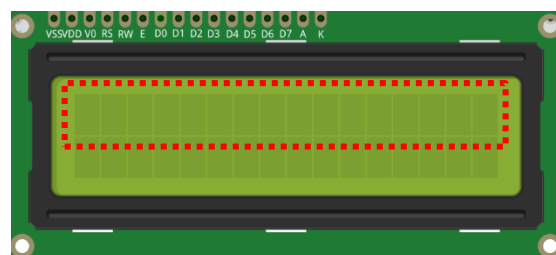
NumberOfLine : جهت تنظیم تعداد سطر های مورد نیاز می باشد . دارای آرگومان های زیر است.

_1LINE_DISPLAY : جهت انتخاب حالت تک خطی.

_2LINE_DISPLAY : جهت انتخاب حالت دو خطی.



حالت دو خطی



حالت تک خطی

مثال :

```
Lcd_Config.Mode = _4BIT_INTERFACE;
```

```
Lcd_Config.Font = _FONT_5X10;
```

```
Lcd_Config.NumberOfLine = _2LINE_DISPLAY;
```

سپس برای استفاده از توابع باید طبق راهنمای زیر آن ها را فراخوانی کنید:

A. تابع **Lcd_Init** :

- این تابع جهت اعمال تنظیمات اولیه و راه اندازی LCD می باشد و مانند مثال زیر باید قبل از دیگر توابع آن را فراخوانی کنید.

```
Lcd_Init();
```

B. تابع **Lcd_SendCommand** :

- این تابع جهت ارسال دستورات می باشد و به صورت زیر استفاده می شود.

```
Lcd_SendCommand( your command );
```

your command دارای مقادیر زیر می باشد :

_CLEAR_DISPLAY	→	پاک کردن ال سی دی
_RETURN_HOME	→	برگشتن به خانه اول
_SHIFT_CURSOR_TO_LEFT_AFTER_WRITE_CHARACTER	→	شیفت دادن مکان نما به خانه سمت چپ بعد از چاپ یک کاراکتر
_SHIFT_CURSOR_TO_RIGHT_AFTER_WRITE_CHARACTER	→	شیفت دادن مکان نما به خانه سمت راست بعد از چاپ یک کاراکتر
_SHIFT_CURSOR_TO_LEFT	→	شیفت دادن مکان نما به خانه سمت چپ
_SHIFT_CURSOR_TO_RIGHT	→	شیفت دادن مکان نما به خانه سمت راست
_SHIFT_TO_LEFT	→	شیفت دادن متن ال سی دی به سمت چپ
_SHIFT_TO_RIGHT	→	شیفت دادن متن ال سی دی به سمت راست
_DISPLAY_OFF	→	خاموش کردن نمایشگر
_DISABLE_LCD_CURSOR	→	غیرفعال کردن مکان نما
_BLINK_LCD_CURSOR	→	تنظیم مکان نما روی حالت چشمک زن
_FIXED_LCD_CURSOR	→	تنظیم مکان نما روی حالت ثابت

مثال :

```
Lcd_SendCommand( _DISABLE_LCD_CURSOR );
```

```
Lcd_SendCommand( _SHIFT_TO_RIGHT );
```

C. تابع **Lcd_PutChar** :

- این تابع جهت نوشتن و نمایش یک کاراکتر روی ال سی دی به صورت زیر استفاده می شود :

```
Lcd_PutChar( your character );
```

your character کاراکتر مورد نظر.

مثال :

```
Lcd_PutChar( 'H' );
```

```
Lcd_PutChar( 'L' );
```

D. تابع **Lcd_PutString** :

- این تابع جهت نوشتن و نمایش یک رشته روی ال سی دی به صورت زیر استفاده می شود :

```
Lcd_PutString( your string );
```

your string رشته مورد نظر.

مثال :

```
Lcd_PutChar( "Hello" );
```

```
Lcd_PutChar( "Majid Derhambakhsh" );
```

E. تابع **Lcd_Clear** :

- این تابع جهت پاک کردن و رفتن به خانه اول نمایشگر استفاده می شود مانند مثال زیر:

```
Lcd_Clear();
```

F. تابع **Lcd_GotoXY** :

- این تابع جهت رفتن به خانه خاصی از نمایشگر به صورت زیر استفاده می شود :

```
Lcd_GotoXY( column , row );
```

column : ستون مورد نظر.

Row : سطر مورد نظر.

مثال :

```
Lcd_GotoXY( 0 , 5 );
```

```
Lcd_GotoXY( 1 , 0 );
```

G. تابع **Lcd_ShiftToLeft** :

- این تابع جهت شیفت دادن متن ال سی دی به سمت چپ با تعداد و زمان دلخواه به صورت زیر استفاده می شود :

```
Lcd_ShiftToLeft( n , t );
```

n : تعداد شیفت.

t : زمان هر شیفت به صورت میلی ثانیه.

مثال :

```
Lcd_ShiftToLeft( 10 , 0 );
```

```
Lcd_ShiftToLeft( 3 , 1000 );
```

H. تابع **Lcd_ShiftToRight** :

- این تابع جهت شیفت دادن متن ال سی دی به سمت راست با تعداد و زمان دلخواه به صورت زیر استفاده می شود :

```
Lcd_ShiftToRight( n , t );
```

n : تعداد شیفت.

t : زمان هر شیفت به صورت میلی ثانیه.

مثال :

```
Lcd_ShiftToRight ( 7 , 250 );
```

```
Lcd_ShiftToRight ( 2 , 500 );
```

نیازمندی ها

ماژول **GPIO.h** برای میکروکنترلر های **AVR**

کتابخانه **Hal** جهت استفاده در میکروکنترلر های **ARM** شرکت **ST**

نکات مهم

کلیه دستورات و تنظیمات با `_` شروع می شوند.

کلیه توابع به صورت کامل کیس نوشته شده اند. (شروع هر کلمه با حرف بزرگ)

جهت تنظیم پورت و پین های LCD از فایل [CHARACTER_LCD_CONFIG.h](#) و به صورت زیر استفاده نمایید.

AVR:

```
#define _LCD_CTRL_PORT &PORTB
#define _LCD_DATA_PORT &PORTA

#define _LCD_D0_PIN 0
#define _LCD_RS_PIN 5
#define _LCD_RW_PIN 6
#define _LCD_EN_PIN 7
```

ARM:

```
#define _LCD_CTRL_PORT GPIOB
#define _LCD_DATA_PORT GPIOA

#define _LCD_D0 0
#define _LCD_RS 5
#define _LCD_RW 6
#define _LCD_EN
```

کد های مورد استفاده جهت راه اندازی و استفاده در میکروکنترلر های مختلف تفاوتی ندارند.

برای استفاده از توابع تاخیر زمانی می توانید از تابع `_Delay_Ms(time)` استفاده نمایید.

مثال های استفاده از ماژول

```
#include "CHARACTER_LCD.h"
```

```
Lcd_Config.Font = _FONT_5X10;
```

```
Lcd_Config.Mode = _4BIT_INTERFACE;
```

```
Lcd_Config.NumberOfLine = _2LINE_DISPLAY;
```

```
Lcd_Init();
```

```
Lcd_Clear();
```

```
Lcd_PutChar('A');
```

```
_Delay_Ms(1000);
```

```
Lcd_Clear();
```

```
Lcd_GotoXY(5,0);
```

```
Lcd_PutString("Hello");
```

```
Lcd_ShiftToLeft(11,500);
```

```
_Delay_Ms(1000);
```

نکته : کد های نگارش شده نیازی به تغییر در کامپایلر ها و میکروکنترلر های مختلف ندارند.

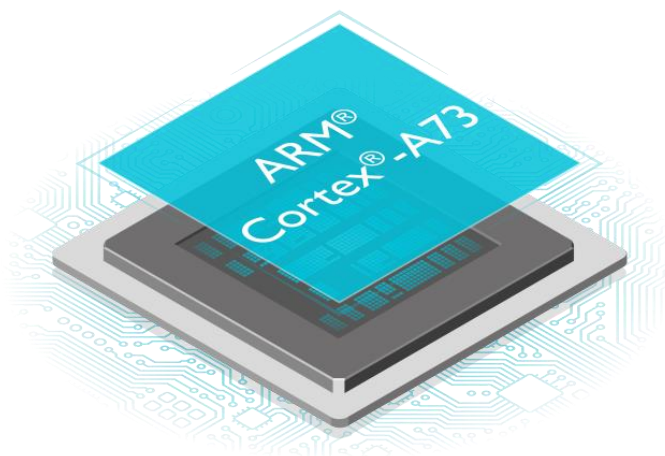
میکروکنترلر های پشتیبانی شده



میکروکنترلر های AVR سری tiny – mega توسط توابع GPIO

کامپایلر های Codevision و GNUC نظیر AtmelStudio

codevision



کلیه میکروکنترلر های ARM سری STM توسط توابع HAL

کلیه کامپایلر های ARM



IAR Embedded Workbench

تغییرات نسخه

Version 0.1.0 :

- * **Add** : Disable 'unused function' warning.
 - * **Improve** : Function speed
-

Version 0.0.5 :

- * **Improve** : Code structure
-

Version 0.0.4 :

- * **Improve** : Library ckeck
-

Version 0.0.3 :

- * **Stable version**
-