

آزمایش ۵: آشنایی با نمایشگرها

هدف آزمایش: آشنایی با LCD کاراکتری^۶ و رنگی و نحوه راهاندازی و به کارگیری آنها

مدت زمان آزمایش: ۸ ساعت



شکل ۱۲ - نمایی از LCD کاراکتری

بخش اول، LCD متنی


در سیستم‌های میکروکنترلی، ممکن است نیاز باشد که پیامی به کاربر نشان داده شود و برنامه‌نویس یا کاربر از تغییرات متغیری در طول روند برنامه آگاهی یابد. بدین منظور از نمایشگرها استفاده می‌شود. یکی از ساده‌ترین نمایشگرها، LCD کاراکتری است. LCDهای کاراکتری معمولاً از روی تعداد سطر و ستون‌های خود شناخته می‌شوند و دارای ۱۴ یا ۱۶ پایه هستند که در نمونه‌های ۱۶ پایه‌ای، ۲ پایه برای نور پس زمینه در نظر گرفته شده است. LCD کاراکتری را می‌توان به دو صورت ۸ بیتی و ۴ بیتی راهاندازی نمود.

ماژول‌های LCD کاراکتری دارای کنترلی به شماره HD44780 هستند. کنترلر واسط میان پانل کریستال مایع و میکروکنترلر است. این چیپ فرامین ارسال شده از میکروکنترلر را دریافت می‌کند و بر اساس آنها پیکسل‌های پانل کریستال مایع را تغییر روشن یا خاموش می‌کند. HD44780 به دلیل واسط پارالل خود به راحتی با میکروکنترلر ارتباط برقرار می‌کند. این کنترلر توانایی نمایش اعداد، حروف و تعدادی کاراکتر خاص در اندازه ۵×۷ پیکسل را دارد، همچنین کاربر می‌تواند ۸ کاراکتر دلخواه را برای LCD تعریف کند. برای کار

^۶ character LCD یا Alphanumeric LCD

با کنترلر لازم است یک برنامه واسط راه انداز برای آن بنویسیم. برای بسیاری از میکروکنترلرها این واسطهای آماده به صورت یک فایل کتابخانه‌ای با پسوند h. موجود است و در صورت عدم دسترسی، باید با توجه به اطلاعات برگه راهنمای کنترلر HD44780 برای راه اندازی آن برنامه مناسب را نوشت.

نمایشگر را می‌توان با استفاده از ۴ یا ۸ بیت باس داده و ۳ خط وضعیت (RS, E, R/W) کنترل کرد. پهنای باس داده در طول پیکربندی اولیه نمایشگر توسط میکروکنترلر انتخاب می‌شود. لذا این امکان وجود دارد که با انتخاب پهنای باس ۴ بیتی، تعدادی از پایه‌های میکروکنترلر را برای مصارف آتی نگه داشت. در این حالت ارسال یا دریافت هر فرمان/داده نیازمند دو سیکل ساعت است. لذا اندکی نسبت به حالت ۸ بیتی کندتر عمل خواهد کرد اما در بسیاری از کاربردها این کند بودن قابل توجه نخواهد بود. با هر لبه پایین‌رونده در پایه E داده‌های قرار داده شده بر روی باس داده به LCD منتقل یا از آن خارج می‌شوند. در حالت ۴ بیتی برای ارسال داده ابتدا نیل (نیم بایت) بالا ارسال می‌شود و سپس نسبت به ارسال نیل پایین اقدام می‌گردد. تقریباً ارسال هر دستور به LCD نیازمند ۴۰ میکروثانیه جهت اجرا شدن است. اما دستوراتی مانند پاک نمودن صفحه و ریست یا انتقال مکان‌نما به محل معین نیازمند ۱/۴ میلی ثانیه جهت اجرا شدن هستند. این زمان‌بندی برای نمایشگرهایی است که توسط کلاک داخلی ۲۵۰ کیلوهرتز کلاک می‌خورند. با این حال این امکان وجود دارد که در هر لحظه با بررسی پرچم مشغول بودن^۷ مشخص شود که آیا نمایشگر آماده دریافت دستور بعدی است یا نه.

 **تحقیق ۱:** چگونه می‌توان به وضعیت فعلی پرچم مشغول بودن در LCD کاراکتری دست یافت؟

شرح پایه های این LCD در ادامه آورده شده است.

جدول ۴ - پایه‌های LCD متنی

پایه	نماد	I/O	شرح
۱	VSS	---	زمین
۲	VDD	---	تغذیه ۵ ولتی
۳	VEE	---	تنظیم کنتراست نمایش
۴	RS	I	اگر صفر باشد رجیستر دستور اگر یک باشد رجیستر داده
۵	R/W	I	اگر صفر باشد برای نوشتن اگر یک باشد برای خواندن از LCD
۶	E	I	فعال‌ساز LCD
۷ تا ۱۴	D0 – D7	I/O	باس داده
۱۵	BLA	---	آند LED جهت نور پس‌زمینه در LCD
۱۶	BLK	---	کاتد LED جهت نور پس‌زمینه در LCD

⁷ Busy flag


هر کاراکتر در LCD دارای یک کد هشت بیتی است که برای نمایش آن استفاده می‌شود. کاراکترهایی که می‌توان به طور معمول در LCD نشان داد در جدول زیر آورده شده است.

Char. code															
	0	0	0	0	0	0	1	1	1	1	1	1	1	1	
	0	0	0	1	1	1	1	0	1	1	1	1	1	1	
	0	1	1	0	0	1	1	1	1	0	0	1	1	1	
	0	0	1	0	1	0	1	0	1	0	1	0	1	1	
xxxx0000				0	@	P	`	P		-	9	≡	α	p	
xxxx0001				!	1	A	Q	a	q	.	7	ç	4	ä	q
xxxx0010				"	2	B	R	b	r	'	ι	τ	×	β	θ
xxxx0011				#	3	C	S	c	s]	υ	̄	ε	ω	
xxxx0100				\$	4	D	T	d	t	\	ε	ι	†	μ	Ω
xxxx0101				%	5	E	U	e	u	.	σ	†	ι	σ	Ü
xxxx0110				&	6	F	V	f	v	7	カ	ニ	ヨ	ρ	Σ
xxxx0111				'	7	G	W	g	w	7	キ	ヲ	ラ	q	π
xxxx1000				(8	H	X	h	x	ι	ク	ネ	リ	ル	ア
xxxx1001)	9	I	Y	i	y	ウ	ケ	ル	ル	"	U
xxxx1010				*	:	J	Z	j	z	エ	コ	ハ	レ	i	チ
xxxx1011				+	;	K	[k	[オ	サ	ヒ	ロ	*	ア
xxxx1100				,	<	L	¥	l	¥	ハ	シ	フ	ワ	Φ	Α
xxxx1101				-	=	M]	m]	ユ	ズ	ン	ム	÷	
xxxx1110				.	>	N	^	n	^	ヨ	セ	ホ	ン	ñ	
xxxx1111				/	?	O	_	o	_	ツ	ソ	マ	°	ö	■

شکل ۱۳ - جدول کد کاراکترهای LCD متنی

به طور مثال اگر لازم باشد کارکتر ? نمایش داده شود باید ۸ بیت ارسال شده به صورت زیر باشد که هنگام برنامه‌نویسی به زبان C این کار توسط کامپایلر انجام می‌گیرد اما برنامه‌نویس می‌تواند به صورت مستقیم هر کاراکتر را بدین صورت نمایش دهد:

00111111


برای آشنایی با LCD و نحوه نمایش کاراکتر و عملکرد آن می‌توانید به فایل راهنمای کاربری آن  (Dot matrix lcd user manual) و فایل هدر lcd.h مراجعه نمایید و یا سایر مستندات موجود در اینترنت را مطالعه کنید.


؟ سؤال ۱: با توجه به خطوط ابتدایی نوشته شده در فایل کتابخانه `lcd.h`^۸ پایه‌های مناسب میکروکنترلر جهت اتصال به LCD را بیابید (در صورت نیاز می‌توانید آن‌ها تغییر دهید) و شماتیک نحوه اتصال آن به میکروکنترلر STM32F429ZIT را با توجه به آن‌چه در `lcd.h` آورده شده است ترسیم کنید.

برای استفاده از این کتابخانه، باید آن را با استفاده از دستور پیش پردازنده `#include` به برنامه خود اضافه کنید. شرح تعدادی از توابع این کتابخانه در زیر آمده است.


جدول ۵ – دستورات کتابخانه LCD کاراکتری

تابع	شرح
<code>lcd_init()</code>	برای پیکربندی اولیه LCD لازم است ابتدا این تابع فراخوانی شود.
<code>lcd_gotoxy(x,y)</code>	مکان نما را به سطر <code>x</code> و ستون <code>y</code> می‌برد.
<code>lcd_print(*str)</code>	رشته <code>str</code> را بر روی LCD نمایش می‌دهد.
<code>lcd_clear()</code>	صفحه LCD را پاک می‌کند.
<code>lcd_putchar(c)</code>	کاراکتر <code>c</code> را بر روی LCD نمایش می‌دهد.
<code>lcd_cmd_write(x)</code>	برای ارسال دستور <code>x</code>
<code>lcd_data_write(x)</code>	برای ارسال داده <code>x</code>
<code>lcd_display_off()</code>	خاموش کردن صفحه نمایش
<code>lcd_define_char</code>	تعریف کاراکتر دلخواه
<code>lcd_cursor_on()</code>	نمایش مکان نما


 **تحقیق ۲:** فایل کتابخانه `lcd.h` را توسط یک ادیتور متن باز کنید و شرح کامل و جزئی تابع `lcd_init` را با توجه به آنچه که در برگه‌ی راهنمای LCD آورده شده است، تحقیق و گزارش نمایید. در صورتی که ورودی یک تابع عددی معین قرار داده شده است، با توجه به آن‌چه در دیتاشیت کنترلر این LCD آورده شده است علت آن را بیان کنید.

 **آزمایش ۱:** ب.ب.ک. شماره دانشجویی شما را در سطر اول LCD و تعدادی از حروف یونانی مانند سیگما و پی را در خط دوم نمایش دهد. با پیاده‌سازی برنامه خود بر روی برد آموزشی، صحت آن را بررسی نمایید.

^۸ از سایت آزمایشگاه قابل دریافت است.

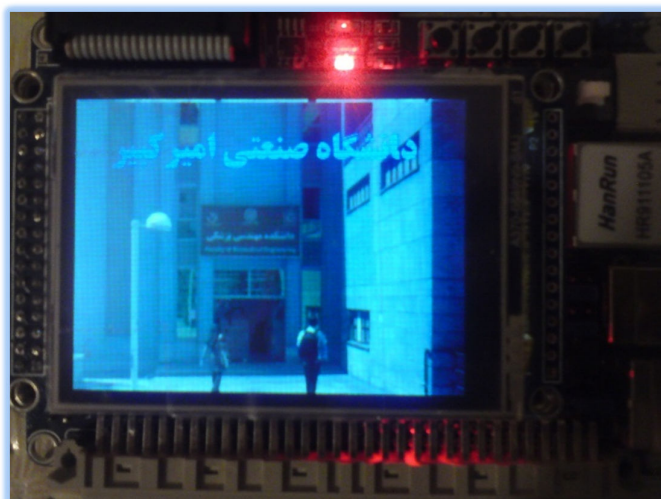
 **آزمایش ۲:** ب.ب.ک. ابتدا واژه‌ای دلخواه روی LCD نوشته و سپس با هر بار فشردن کلید روی برد، بدون نوشتن مجدد، این واژه را یک کاراکتر به سمت راست شیفت دهد (شیفت به چپ و راست جزء دستورات کنترلی LCD کاراکتری است).

این امکان وجود دارد که ۸ کاراکتر توسط کاربر به صورت دلخواه تعریف شوند. دیتای پیکسلی این کاراکترها در حافظه تولید کاراکتر LCD ذخیره می‌شود. تعریف شدن هر کاراکتر نیازمند ۸ بایت است که بیت ۰ تا ۴ هر بایت برای تعریف ۵ پیکسل در هر سطر استفاده می‌شود. ابتدایی‌ترین بایت مربوط به بالاترین سطر و آخرین بایت در تعریف کاراکتر مربوط به انتهایی‌ترین سطر است. اگر چه تولید کد نمایش هر کاراکتر به صورت دستی بسیار ساده است اما معمولاً برای تولید کدهای آماده برای کاراکترهای دلخواه از نرم‌افزارهای تولیدکننده این کدها استفاده می‌کنند.

 **آزمایش ۳:** ب.ب.ک. به زبان فارسی نام خود را بر روی LCD نمایش دهید.

بخش دوم، TFT LCD

در قسمت دوم آزمایش می‌خواهیم با LCD رنگی و نحوه به کارگیری آن آشنا شویم. LCDهای رنگی توانایی تفکیک رنگ و نمایش تصویر با رزولوشن بالاتر را فراهم می‌کنند. مانند LCD کاراکتری، بر روی پنل این LCDها نیز یک چیپ کنترلر قرار دارد که فرامین ارسالی از طریق میکروکنترلر را دریافت و تفسیر کرده و بر روی پنل LCD اجرا می‌کند. یکی از محدودیت‌های کار با LCDهای رنگی نیاز به سرعت بالا در ارتباط با آن است. در غیر این صورت تصاویری که پشت سر هم نمایش داده می‌شوند به صورت فریم-فریم ظاهر شده و گسستگی در نمایش به وجود می‌آید که این به دلیل حجم بالای اطلاعات تبدالی بین میکروکنترلر و LCD است. در برخی میکروکنترلرها و اکثر پردازنده‌های پیشرفته (مانند پردازنده‌های Cortex-A که در موبایل‌ها و تبلت‌ها استفاده می‌شوند) یک واحد جانبی برای کارهای گرافیکی و تبادل اطلاعات در نظر گرفته شده است تا CPU کم‌تر درگیر راه‌اندازی LCD شود.



شکل ۱۴ - LCD گرافیکی رنگی

در میکروکنترلرهای STM32F429xx امکانات جانبی مختلفی برای راه اندازی بهینه LCD درگیر هستند:

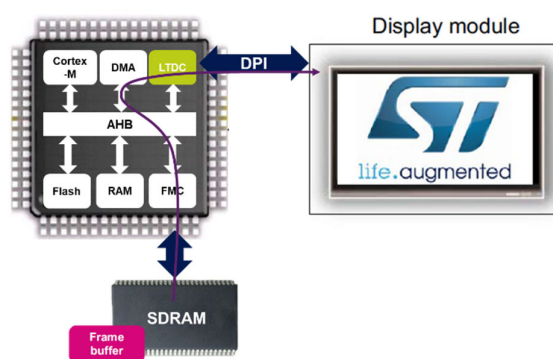
- واحد LTDC (LCD-TFT Controller): یک بخش اختصاصی برای ارتباط با LCDهای گرافیکی است که کلاک های لازم برای رفرش شدن درست LCD (کلاک کلی، سیگنال سنکرونی افقی و سیگنال سنکرونی عمودی) به همراه باس داده های رنگ پیکسل ها (باس رنگ قرمز، سبز و آبی یا RGB) را تأمین می کند.
- واحد FMC (Flexible Memory Controller): این واحد واسط بین میکروکنترلر با انواع حافظه های خارجی است و زمان بندی و قالب اطلاعات برای خواندن/نوشتن از حافظه خارجی را مدیریت می کند. در راه اندازی LCD، به کمک این واحد می توان با حافظه گرافیکی LCD که اطلاعات تصویر را در خود نگه می دارد، ارتباط پیدا کرد.
- واحد DMA (Direct Memory Access): DMA واحدی است که برای انتقال مستقیم اطلاعات بین دو قسمت از حافظه داخلی یا بین حافظه داخلی و peripheralها و یا حتی بین دو peripheral مختلف استفاده می شود. بدون بهره گیری از DMA، انتقال اطلاعات باید مستقیماً توسط CPU انجام شود که بخش زیادی از توان محاسباتی آن را به خود اختصاص می دهد. اما DMA با داشتن باس داده اختصاصی و کلاک مجزا از CPU، کار انتقال اطلاعات را بدون درگیر کردن CPU انجام می دهد. در STM32F429 یک DMA اختصاصی به نام DMA2D تعبیه شده است که به صورت تخصصی برای انتقال داده های تصاویر رنگی کاربرد دارد.

با مراجعه به فایل راهنمای کاربری، بخش های LTDC، FMC و DMA2D (فصول ۱۶، ۳۷ و ۱۱) مشاهده کنید و با امکانات و هدف کلی این بخش ها آشنا شوید.

LCD رنگی مورد استفاده در آزمایشگاه، با رزولوشن 240×320 پیکسل^۹ (QVGA)، توانایی تفکیک ۲۶۲۱۴۴ رنگ مختلف را دارد. معمولاً بر روی پنل LCD صفحه‌ی لمسی قرار داده می‌شود که بر روی LCD مورد استفاده در آزمایشگاه نیز پنل لمسی از نوع مقاومتی تعبیه شده است. درایور این LCD، ILI9341 نام دارد.

با مراجعه به برگه اطلاعاتی AN4861، فصل اول را مطالعه کنید تا با اصول کارکرد LCDهای گرافیکی آشنا شوید.

LCD موجود در بردهای آموزشی آزمایشگاه از نوع DPI (MIPI Display Parallel Interface) بدون حافظه بافر داخلی است. شکل ۱۵ نحوه انتقال اطلاعات در طراحی این بردها را نشان می‌دهد که در آن یک SDRAM خارجی نقش فریم بافر یا همان حافظه گرافیکی را بازی می‌کند. این حافظه به کمک واحد FMC خوانده شده و محتوای آن از طریق واحد DMA2D به طور مستقیم روی LTDC می‌رود تا اطلاعات رنگ هر پیکسل ساخته شده و صفحه نمایش به روز شود.




شکل ۱۵ - ساختار ارتباط میکروکنترلر با TFT

تحقیق ۳: با توجه به تعداد پیکسل‌ها و عمق رنگ LCD مورد استفاده در بردها، حداقل فضای لازم برای حافظه گرافیکی چه مقدار است؟ مدل و ظرفیت حافظه SDRAM تعبیه شده روی برد چیست؟ اگر این دو مقدار به یکدیگر نزدیک نیستند، چه توجیهی برای عدم تناسب آن‌ها می‌توانید ارائه دهید؟

راه‌اندازی LCD گرافیکی اگر چه به نظر پیچیده می‌رسد ولی خوشبختانه شرکت ST همراه با برنامه STMCube مجموعه‌ای از کتابخانه‌ها در اختیار کاربران قرار داده است که کار با LCD برد دیسکآوری را بسیار ساده کرده است. برای این کار به پیوست ۷ مراجعه کنید و مراحل ساخت پروژه را مطابق آنچه گفته شده است، طی نمایید.

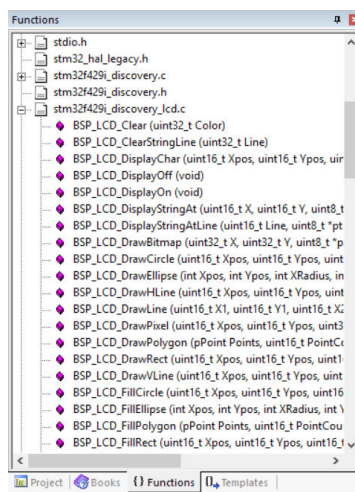
^۹ پیکسل کوچک‌ترین جزء نمایش دهنده در تصویر است.

 **آزمایش ۴:** ب.ب.ک. یک مستطیل توپر رسم کند و با هر فشردن کلید مستطیل از توپر به توخالی و برعکس تغییر شکل دهد.


توجه ۱: بخش LTCD برای تصویر دو لایه در نظر می‌گیرید. بنابراین پیش از شروع کار با LCD، اولاً باید این لایه‌ها را تنظیم نمایید و در ثانی تعیین کنید تغییراتی که می‌دهید، در کدام لایه باشد. برای نمونه، می‌توانید کد زیر را بررسی کنید.


```
BSP_LCD_Init();
BSP_LCD_LayerDefaultInit(LCD_BACKGROUND_LAYER, LCD_FRAME_BUFFER);
BSP_LCD_LayerDefaultInit(LCD_FOREGROUND_LAYER, LCD_FRAME_BUFFER);
BSP_LCD_SelectLayer(LCD_FOREGROUND_LAYER);
BSP_LCD_DisplayOn();
```

توجه ۲: لیست دستورات کتابخانه BSP برای کار با LCD را می‌توانید در قسمت Functions و در بخش stm32f429i_discovery_lcd مشاهده کنید.




شکل ۱۶ - دستورات کتابخانه BSP برای LCD گرافیکی

 **تحقیق ۴:** در مورد چگونگی آماده‌سازی و نمایش یک عکس دلخواه بر روی LCD رنگی تحقیق و گزارش کنید.

 **آزمایش ۵:** ب.ب.ک. تصویر خود را بر روی LCD نمایش دهد و در انتهای تصویر شماره دانشجویی و نامتان را به ترتیب با رنگ‌های قرمز و آبی و با دو سایز مختلف بنویسد. با پیاده‌سازی برنامه خود بر روی سخت‌افزار آزمایشگاه، صحت آن را بررسی کنید.

? سؤال ۲: با بررسی فایل‌های موجود در پوشه Fonts، نحوه استفاده از فونت دلخواه را هنگام نمایش یک متن تشریح کنید. هر کاراکتر در این فونت چگونه ساخته می‌شود؟

 **آزمایش ۶:** در ابتدای برنامه آزمایش ۵، دستور `ili9341_WriteReg(0x21)` را اضافه کرده و نتیجه را مشاهده کنید.

? سؤال ۳: با مراجعه به دیتاشیت ili9341 بررسی کنید، فرمان ارسال شده در آزمایش ۶، موجب چه تغییری در LCD شده است؟