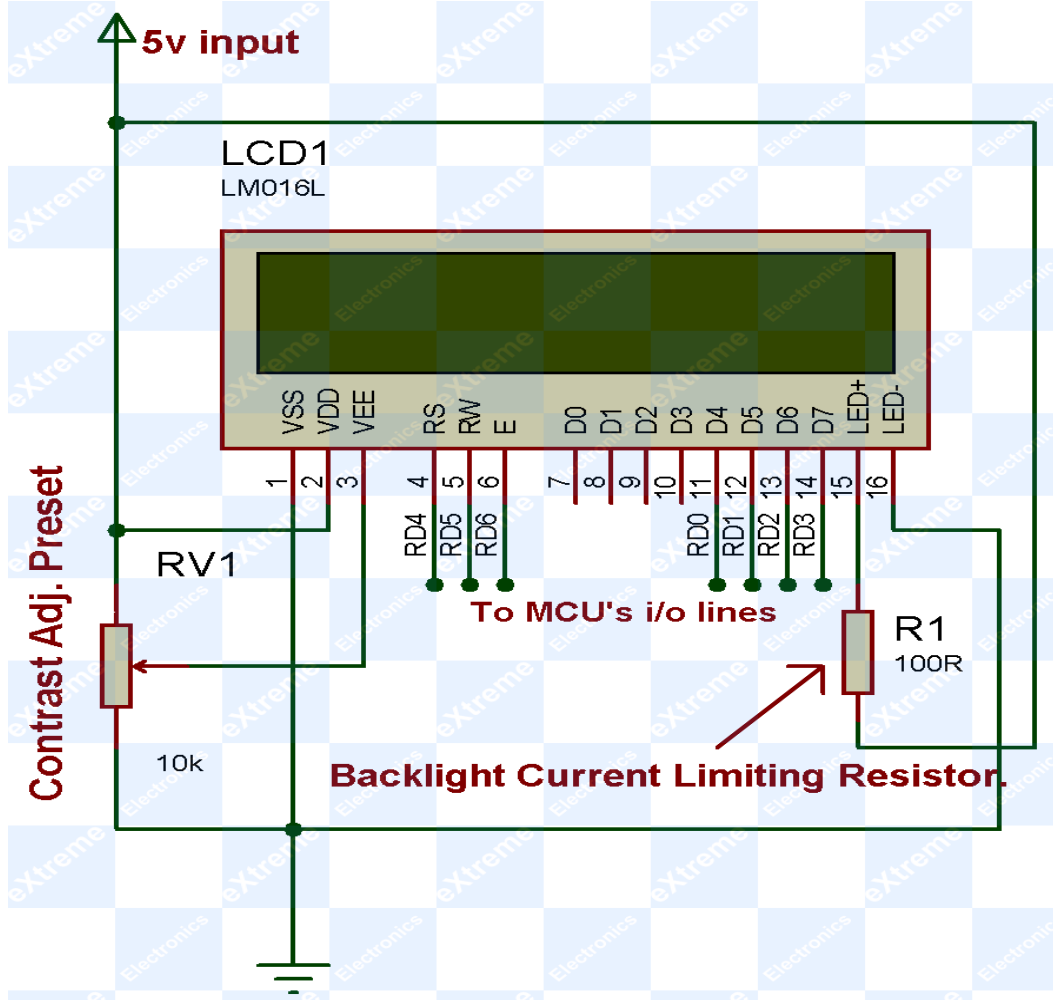


بسم الله الرحمن الرحيم

پرهام کریمی ریکنده 9423090

- 1- در ورودی تمام توابع یک اشاره گر به تایپ `lcd_t` ورودی داده شده است. علت چیست؟
- a. برای استفاده از GPIO نیاز داریم شماره ی پین های متصل به LCD و پورت های آنها را بدانیم:
- i. برای ارسال داده و دستور به ال سی دی نیاز داریم که از GPIO استفاده کنیم؛ چراکه درواقع نوع ارسال داده ها موازی است.
- ii. پایه های کنترلی که در واقع پروتکل ارسال و دریافت را تشکیل می دهند، باید توسط GPIO کنترل شوند.
- iii. در برخی موارد نیاز به خواندن از رجیستر های ال سی دی داریم که برای تنظیم پایه های دیتا به صورت ورودی باز هم به اطلاعات پین های متصل نیاز داریم.
- b. برای تنظیم مد کاری ال سی دی تغییری در این تایپ گنجانده شده
- c. با این کار (تعریف این تایپ) تمامی اطلاعات مذکور را در یک ساختار یکپارچه کرده ایم
- d. چرا اشاره گر؟ ← میتوانیم یک `instance` از این ساختار در بیرون `main` بسازیم تا `global` تعریف شود و سپس تنظیم مقادیر آن را در `main` انجام دهیم. هر بار که در هر تابعی نیاز به استفاده از این ساختار داشتیم، آدرس این `instance` را ارسال میکنیم.
- i. فایده: با این کار `instance` ساخته شده در ناحیه ی `(BSS) data` حافظه قرار خواهد گرفت اما اگر در `main` تعریف شود، از /انجایی که "محلی" است، در `stack` قرار می گیرد. خب که چی؟! ← خب ما ترجیحمون به خالی موندن `stack` هست چون در فراخوانی توابع، متغیر های توابع [که محلی هستند]، در `stack` قرار می گیرند و متغیرهایی مانند این اگر در `main` تعریف شوند، در تمام طول اجرای تابع `main` (در واقع تمام مدت کارکرد میکرو!!) در `stack` باقی میمانند و جای ما رو تنگ میکنند!! پس میایم `global` تعریفش میکنیم.
- به چیز دیگه هم اینکه که وقتی آدرس رو به تابع میفرستیم، فقط داریم به متغیر نهایتاً 32 بیتی رو جابجا میکنیم اما وقتی خودش رو بفرستیم به تابع، داریم به اندازه ی تمام متغیر های داخل این ساختار، جا اشغال میکنیم!
- e. مطالعه ی این سوال و جواب هاش مفیده:
- <https://stackoverflow.com/questions/29708677/difference-between-global-variables-and-variables-declared-in-main-in-arm-c>

2- این نوع lcd 16 پایه دارد. راجب فانکشن آنها توضیح دهید؟
 a. پاسخ این سوال را با توجه به تصویر زیر مینویسم:



b.

c. پایه ی 1 و 2 تغذیه ی ال سی دی هستند

d. پایه ی 3 برای تنظیم کنتراست استفاده می شود

i. با توجه به ساختار درونی ال سی دی (کریستال مایع!)، حساسیت آن نسبت به دما زیاد است. در نتیجه از این قابلیت آن، در کنار مدارهای سنش دما، میتوان عملکرد مناسب را در هر زمان از آن گرفت.

ii. معمولا با یک پتانسیومتر و تقسیم ولتاژ، در

مدارهای آزمایشی، آن را تنظیم میکنند

iii. من بخاطر اینکه حوصله ی مدار بستن نداشتم، وصل

کردم به تایمر میکرو، بهش PWM دادم. (البته

میدونیم که این در واقع یه آپشنه چون با میکرو

میشه کنترلش کرد و این هم اگه حواسمون باشه که

این میکرو ها توی خودشون یه دماسنج دارن، دیگه

خیلی خفنه!!!) (البته در حقیقت این یه اسراف توی

مصرف پایه های میکرو محسوب میشه ولی خب حالا الان

که کار دیگه ای نداریم با میکرو!!)

- e. پایه ی 4 ، پایه ی RS است که برای اعلام این به میکرو هست که دارم بهت دستور میدم یا داده میدم.
- i. صفر بشه ← دستور
 - ii. یک بشه ← داده
- f. پایه ی 5، پایه ی R/W هست که میگیریم بهش داریم میخوانیم یا داریم مینویسیم (چون میشه رجیستر های ال سی دی رو خونند!)
- i. صفر بشه ← نوشتن
 - ii. یک بشه ← خواندن
 - iii. از اونجایی که توی صورت سوال، این رو بین پین ها شما نداشتید، اینطوری توجیهش کردیم که چون کلا میخوایم بنویسیمش، پس دیگه یه پایه ی میکرو رو حروم نکردین و در نتیجه ما باید دستی خودمون صفرش کنیم، پس وصل کردم به زمین.
- g. پایه ی 6، پایه ی EN هستش که برای ارتباط با ال سی دی، نقش مهمی داره؛ خلاصش این میشه که کلا هر وقت چیزی خواستیم بفرستیم، یه لبه ی پایین رونده روی این ایجاد میکنیم و یه مقدار صبر میکنیم (توی دیتاشیت برای این ال سی دی نوشته t_H حداقل 10 نانو ثانیه) بعدش داده روی ال سی دی نشسته!
- i. برای حالت 4 پایه، برای هر بخش از داده باید این کار رو بکنیم.
- h. پایه ی 7 تا 14، پایه های ارتباطی داده هستند.
- i. با اعمال صفر و یک روی این پایه ها، داده/دستور/آدرس رو به میکرو میدیم
 - ii. در کاربرد 4 پایه، باید پایه های 7 تا 10 رو زمین کنیم (یا کاریشون نداشته باشیم (فلوت)) و پایه های 11 تا 14 رو به میکرو متصل کنیم.
- i. پایه ی 15 و 16 که مربوط به تغذیه ی نور پس زمینه ی ال سی دی هستند و اینم میتونیم با یه پتانسیومتر و تقسیم ولتاژ، تنظیمش کنیم.
- i. اینم مثل کنتراست، به PWM میکرو وصل کردم چون حوصله ی مدار بستن نداشتم. (البته آپشن محسوب میشه چون با میکرو میتونیم تنظیمش کنیم!!!)
- ژ. در نهایت چندتا عکس دیگه هم میذارم از یه کتاب که برای AVR بوده ولی خوب توضیح داده:

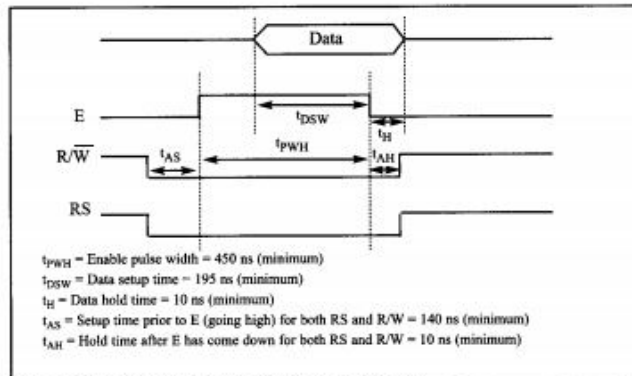
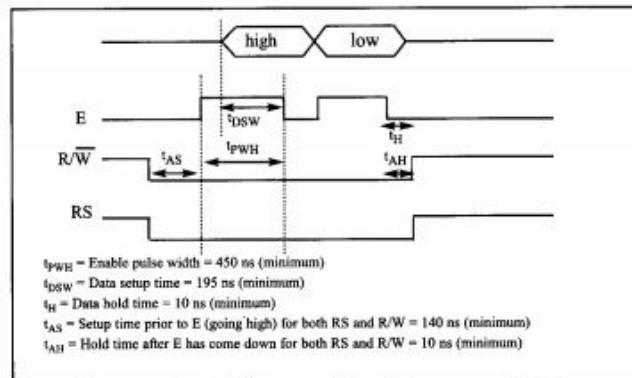
Table 12-2: LCD Command Codes

Code (Hex)	Command to LCD Instruction Register
1	Clear display screen
2	Return home
4	Decrement cursor (shift cursor to left)
6	Increment cursor (shift cursor to right)
5	Shift display right
7	Shift display left
8	Display off, cursor off
A	Display off, cursor on
C	Display on, cursor off
E	Display on, cursor blinking
F	Display on, cursor blinking
10	Shift cursor position to left
14	Shift cursor position to right
18	Shift the entire display to the left
1C	Shift the entire display to the right
80	Force cursor to beginning of 1st line
C0	Force cursor to beginning of 2nd line
28	2 lines and 5 × 7 matrix (D4–D7, 4-bit)
38	2 lines and 5 × 7 matrix (D0–D7, 8-bit)

Note: This table is extracted from Table 12-4.

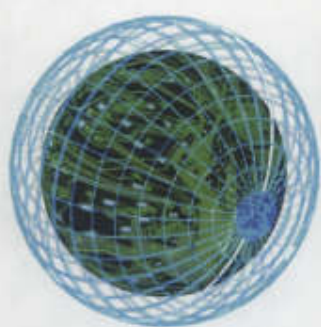
Table 12-1: Pin Descriptions for LCD

Pin	Symbol	I/O	Description
1	V _{SS}	--	Ground
2	V _{CC}	--	+5 V power supply
3	V _{EE}	--	Power supply to control contrast
4	RS	I	RS = 0 to select command register, RS = 1 to select data register
5	R/W	I	R/W = 0 for write, R/W = 1 for read
6	E	I/O	Enable
7	DB0	I/O	The 8-bit data bus
8	DB1	I/O	The 8-bit data bus
9	DB2	I/O	The 8-bit data bus
10	DB3	I/O	The 8-bit data bus
11	DB4	I/O	The 8-bit data bus
12	DB5	I/O	The 8-bit data bus
13	DB6	I/O	The 8-bit data bus
14	DB7	I/O	The 8-bit data bus

**Figure 12-5. LCD Timing for Write (H-to-L for E line)****Figure 12-6. LCD Timing for 4-bit Write**

the avr microcontroller and embedded system

using assembly and c



MUHAMMAD ALI MAZIDI
SARMAD NAIMI
SEPEHR NAIMI



.1

1. راستی توی این کتاب کد رو با فرض این نوشته که همه ی پایه های داده ی ال سی دی به یک پورت وصل شدن و پشت هم هستن (همونطوری که خودش شماتیک رو کشیده) ولی با توجه به صورت سوال شما، همیشه اینطوری کد رو نوشت. (کدی که از آقای نجفی مدرس مجتمع فنی هم در دست هست، همین فرض رو کرده!)

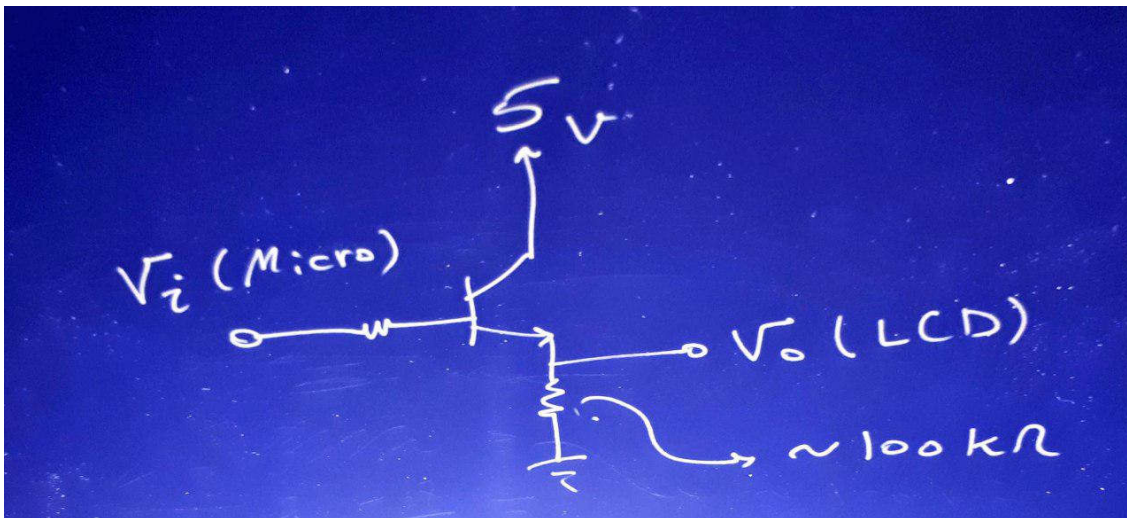
3- محدود تغذیه آن از چند ولت تا چند ولت می باشد؟

a. این دوتا سوال آخر رو با توجه به این جدول دیتاشیت

دارم جواب میدم:

Item	Symbol	Condition	Standard value			Unit	Applicable terminal
			Min.	Typ.	Max.		
Power voltage	V _{dd}		4.5	5.00	5.5	V	V _{dd}
Input H - level voltage	V _{IH}		2.2	—	V _{dd}	V	RS, R/ \overline{W} , E DB0~DB7
Input L - level voltage	V _{IL}		-0.3	—	0.6	V	
Output H - level voltage	V _{OH}	- I _{OH} = 0.205mA	2.4	—	—	V	DB0~DB7
Output L - level voltage	V _{OL}	I _{OL} = 1.2mA	—	—	0.4	V	
I/O leakage current	I _{IL}	V _{in} = 0~V _{dd}	-1	—	1.0	μ A	RS, R/ \overline{W} , E DB0~DB7
Supply current	I _{dd}	V _{dd} = 5V	2	—	—	mA	V _{dd}
LCD operating voltage	V _{LCD}	V _{dd} - V _O	3.0	—	11.0	V	V _O

- b. محدوده ی ولتاژ کاریش از 4 و نیم تا 5 و نیم هست.
- 4- فرض کنید میکروکنترلر با ولتاژ تغذیه 2 ولت کار می کند. در این حالت یک شماتیک راجع به نحوه اتصال lcd به میکروکنترلر مورد نظر ارائه دهید؟
- a. چندتا داده ای که مهمه:
- i. جریان نشتی I/O ها:
 1. نهایتا 1 میکرو آمپر
 - ii. حد ولتاژ ورودی سطح High:
 1. 2.2 ولت
 - iii. حد ولتاژ ورودی سطح Low:
 1. 0.6 ولت
 - iv. این دو حد در خروجی مهم نیستن (در این تمرین!)
- چون ما فقط میخوایم به ال سی دی ورودی بدیم
- b. خب حالا با توجه به اینا این شماتیک رو پیشنهاد میکنم:



- c. وقتی میکرو صفر میده، ترانزیستور قطعه، پس توی شاخه ی امیتر جریان نداریم، پس خروجی که وصله به ال سی دی، صفره، البته ی کمی جریان نشتی داریم که برای خود ال سی دی و خود ترانزیستور در حالت قطع، هستن ولی اونا با توجه به دیتاشیت در حدود میکرو آمپر هستن ← بخاطر همین هم مقاومت امیتر باید در همین حوالی 100 کیلو باشه (اگه محاسبه کنیم اگه فقط جریان نشتی ال سی دی رو در نظر بگیریم (یعنی 1 میکرو آمپر)، باید مقاومتمون

کمتر از 600 کیلو اهم باشد تا افت ولتاژش کمتر از 0.6 ولت بشه تا سطح منطقی Low رو بشناسه)
d. وقتی میکرو 1 میده، یعنی 2 ولت میشه ورودی، ترانزیستور اشباع میشه، یعنی ولتاژ خروجی میشه 5 ولت تغذیه منهای ولتاژ اشباع یعنی حدود 0.2 ولت، یعنی 4.8 ولت که همچنان از حد ال سی دی (2.2 ولت) خیلی بالاتره!!
i. در این حالت هرچی مقاومت امیتر بزرگ تر باشه بهتره!! چون توان مصرف مدارمون کمتر میشه! پس حد پایین مقاومت امیترمون رو هم این حالت تعیین میکنه.

ii. اگر همون 100 کیلو اهم بذاریم، میشه حدود 48 میکرو آمپر که عدد خیلی خوبی نیست (خود ال سی دی 1 میکروئه یعنی این الان 48 برابر نشسته!!!) ولی وحشتناکم نیست! (تازه! میکرو که همیشه 1 نمیده! دقیقاً برعکس! اکثراً صفره مگه اینکه داده ای بخواد ارسال بشه!)

5- در نهایت یه توضیحاتی درباره ی ال سی دی هست که برای یکی از دوستانم ویس فرستادم، اونا رو هم آپلود میکنم!

6- کد این تمرین رو هم 2 سال قبل تو عید برای AVR زده بودم، با کمک همین کتابه، همونو تبدیلش کردم به جواب این تمرین!

7- **با توجه به زیاد بودن حجم فایل ها، (مخصوصاً فولدر گیت که گفته بودید بفرستیم!) فقط فایل های اصلی به همراه گزارش ارسال شده اند. پروژه ی گیت به صورت عمومی تعریف شده و همه (و در واقع شما) میتونن بهش دسترسی داشته باشن؛ آیدی گیت من:**

Parhamk75 .a

8- لینک مربوط به این تمرین:

https://github.com/parhamk75/MICRO_HW2 .a