به نام خدا

تمرین سری دوم میکرو

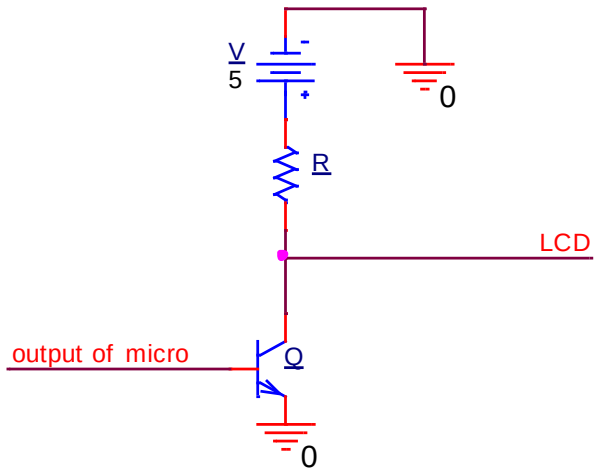
سید محمد مظفری 9423106

LCD دارای 16 پایه است که کاربرد هر کدام از آن ها به قرار زیر می باشد:

- V_{CC} که ولتاژ 5+ را برامان تامین خواهد کرد.
- V_{SS} که زمین LCD را تعیین میکند.
- V_{EE} که میزان contrast صفحه ی LCD را تعیین میکند.
- R_S انتخاب کننده‌ی رجیستر است. (Register select) دو رجیستر مهم در LCD وجود دارد که پین R_S این دو رجیستر را انتخاب خواهد کرد به این صورت که اگر $R_S = 0$ آنگاه رجیستر instruction command code انتخاب خواهد شد که به کاربر اجازه می‌دهد دستورهایی از قبیل clear display, cursor at home و غیره را به LCD بدهد. اگر $R_S = 1$ باشد آنگاه data register انتخاب می‌شود که به کاربر اجازه می‌دهد data را برای نمایش بروی LCD به آن ارسال کند.
- R/W(read/write) به کاربر اجازه‌ی خوادن اطلاعات از LCD و نوشتن اطلاعات بر روی LCD را می‌دهد به این صورت که $R/W = 1$ برای خواندن و $R/W = 0$ برای نوشتن.
- E(enable) پین enable بدین منظور به‌کار می‌رود که data های قرار گرفته بر روی data pin ها را latch کند (یا به عبارتی دیگر نگه دارد). برای این کار هنگامی که اطلاعاتی بر روی پین های ورودی اطلاعات LCD قرار گرفت، یک پالس پایین رونده (از سطح بالا به پایین) باید به پین enable اعمال کنیم تا LCD اطلاعات را Latch کند. طول این پالس باید حداقل $450n_S$ باشد.
- D0-D7 پین‌های data برای LCD هستند که برای فرستادن اطلاعات به LCD و خواندن محتوای رجیسترهای داخلی آن به کار می‌روند. برای نمایش دادن کلمات و اعداد بر روی LCD کدهای ASCII مربوط به حروف A-Z و a-z و اعداد

0-9 را در حالتی که پین $R_S = 1$ است (یعنی data register انتخاب شده است) به این پین‌ها ارسال می‌کنیم.

می‌توان در خروجی میکرو از ساختار یک ترانزیستور که ورودی LCD از درین آن گرفته می‌شود استفاده کرد در این صورت اگر فرض کنیم خروجی میکرو در حدود 2 ولت در سطح بالا و نزدیک به 0 در سطح پایین باشد و همچنین اگر درین ترانزیستور قرار گرفته در خروجی میکرو را به 5 ولت وصل کنیم، آنگاه هنگامی که خروجی میکرو high است، ترانزیستور به اشباع رفته و ورودی LCD صفر خواهد شد و نیز هنگامی که خروجی میکرو low است ورودی LCD به 5 ولت رسیده و high خواهد شد. در واقع با این روش ولتاژ کم خروجی میکرو را به یک ولتاژ با سطح بالاتر تبدیل کرده‌ایم.



در مورد ورودی توابع که به صورت پوینتر داده شده بود، علت آن است که هنگامی که یک struct تعریف می‌کنیم و می‌خواهیم آن را به عنوان ورودی به توابع بدهیم، اگر به صورت void این ورودی‌ها را داده و از متغیرهایی که به صورت struct تعریف کرده

استفاده کنیم، آنگاه تمام این متغیرها یکبار کپی شده و سپس مورد استفاده قرار می-گیرد در صورتی که این کار اتلاف حافظه و source است. ولی اگر ورودی توابع را به صورت pointer بدهیم آنگاه خود متغیرها به توابع رفته و مورد استفاده قرار می‌گیرند.

لینک زیر، لینک git برای تمرین‌های کد است.

https://github.com/seyyedmm/arm