

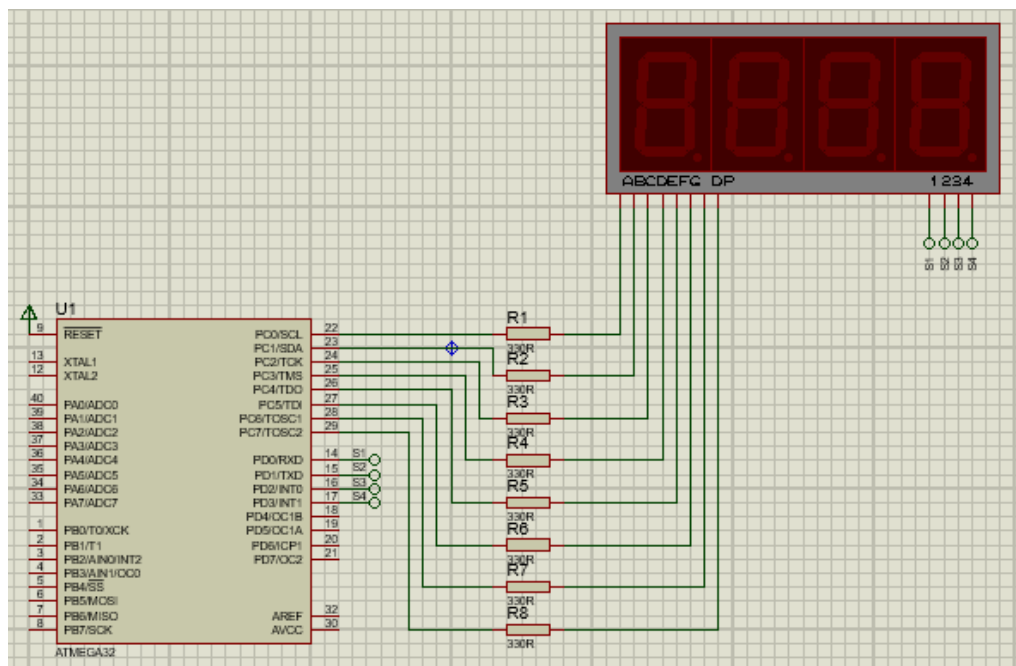
به نام خدا

تمرین ششم درس ریزپردازنده

غزل زمانی نژاد

۹۷۵۲۲۱۶۶

1. در پیاده سازی بخش سخت افزاری این سوال از ATMEGA32، قطعه ای دارای 4 SEVEN SEG و چند مقاومت استفاده میکنیم. بیت های S1 تا S4 شمارنده برای enable کردن هر کدام از 7seg ها به کار میروند و به D0 تا D3 میکروکنترلر متصل هستند.

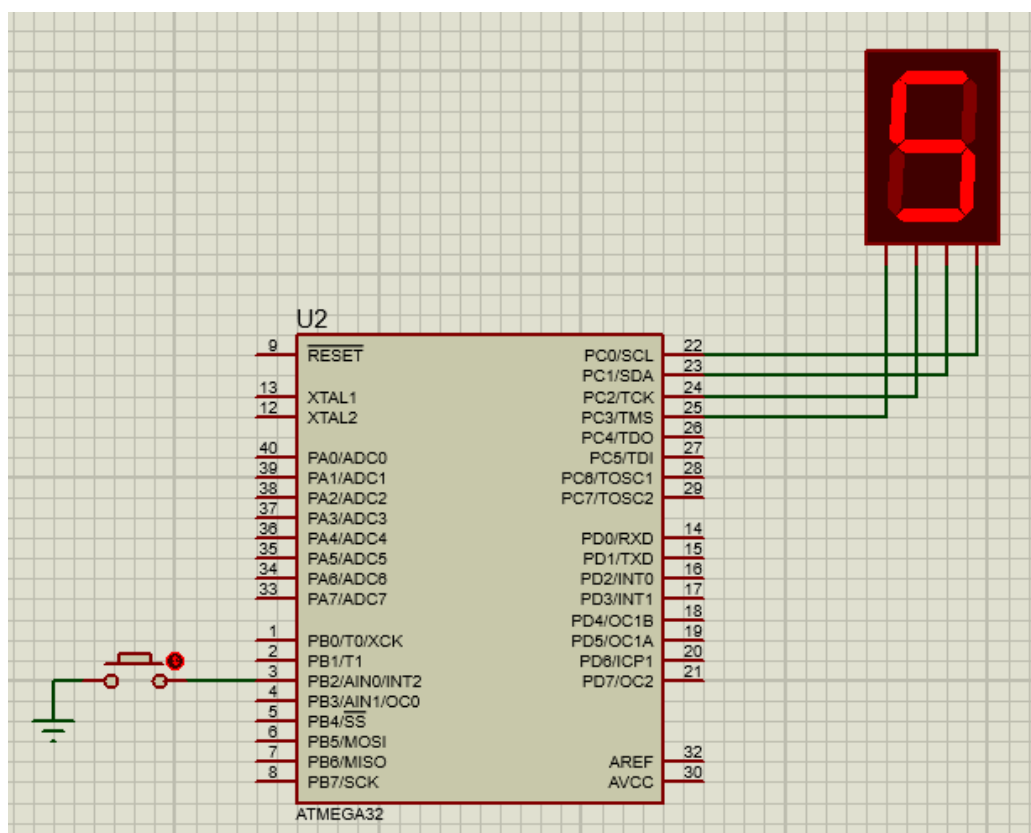


برای پیاده سازی کد اسمبلی این سوال، جهت پین های C,D را به صورت خروجی قرار میدهیم. از رجیسترهای R20,R21 برای ذخیره عدد فعلی و رجیسترهای R22,R23 برای عدد قبلی در دنباله فیبوناچی استفاده می کنیم. در لیبل CALCULATE، مقدار عدد فیبوناچی جدید را با جمع کردن دو مقدار قبلی موجود در دنباله محاسبه می کنیم. سپس مقدار یکان را با 10 compare میکنیم. اگر بزرگتر باشد نیاز به تغییر دارد. آن را 0 میکنیم و به رقم دهگان یکی اضافه میکنیم. در غیر این صورت برچسب LOW_YEKAN را اجرا میکنیم. همین عملیات را برای رقم دهگان نیز انجام میدهیم. در صورتی که دهگان از 10 بیشتر شود، باید عدد را به مقدار 01 ریست کنیم.

در آخر با استفاده از DISPLAY، ارقام محاسبه شده را نمایش می دهیم. باید هر کدام از ارقام را به گونه ای تغییر دهیم که در 7SEG قابل نمایش باشند، به همین دلیل CONVERT را صدا میزنیم. برای اینکه اعداد به درستی با چشم قابل مشاهده باشند، بعد از نمایش آنها از DELAY استفاده میکنیم.

2. .

3. در پیاده سازی بخش سخت افزاری این سوال از ATMEGA32، یک چراغ SEVEN SEG و یک دکمه برای متوقف کردن شمارش استفاده میکنیم.



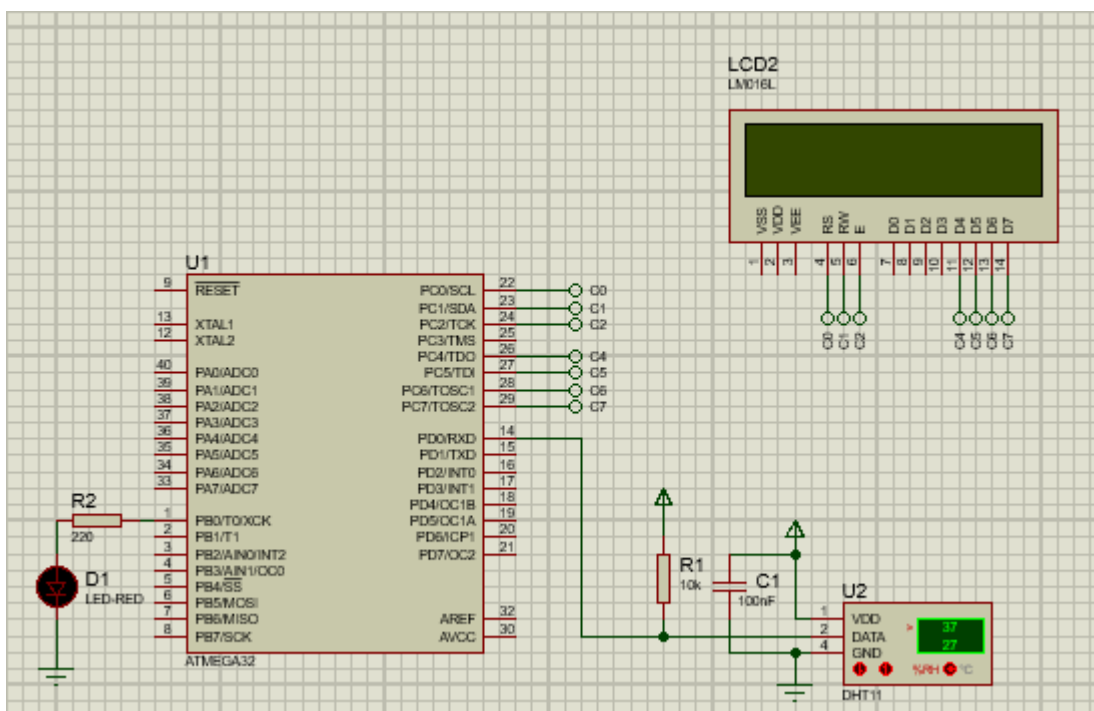
در اینجا دکمه به پین INT2 میکروکنترلر متصل است. برای وقفه های از نوع external interrupt به کار میرود. هر وقت کاربر بر روی دکمه کلیک کند، متد ext_int2_isr صدا زده می شود. چون وظیفه این دکمه متوقف کردن شمارش است، در کد آن مقدار pause را ست میکنیم تا شمارش متوقف شود. برای فرایند شمارش، از تابع timer1_compa_isr استفاده میکنیم. در این متد، اگر مقدار pause صفر نباشد، یعنی باید به شمارش ادامه دهیم پس متغیر number را افزایش میدهیم.

در بخش main برنامه، پین های C را به عنوان خروجی ست میکنیم. رجیسترهای به خصوصی را به گونه ای تغییر میدهیم که وقفه های خارجی ایجاد شود. همچنین کانفیگ های تایمر را به گونه ای ست می کنیم که در هر ثانیه وقفه ایجاد کند و کار شمارش را انجام دهد. سپس با اجرای دستور

```
#asm("sei")
```

وقفه ها را enable میکنیم. در پایان در یک حلقه بی نهایت، عدد فعلی را روی پورت C میفرستیم.

4. در پیاده سازی بخش سخت افزاری این سوال از ATMEGA32، قطعه DHT11 برای تشخیص میزان رطوبت، یک LCD برای نشان دادن میزان رطوبت و یک LED برای هشدار استفاده میکنیم.



در پیاده سازی بخش نرم افزاری، ابتدا دو متد response, request را پیاده سازی میکنیم تا سنسور تشخیص رطوبت برای تشخیص میزان رطوبت آماده باشد. سپس در متد receive_data مقدار رطوبتی که سنسور تشخیص داده را دریافت می کنیم.

در بخش main برنامه ابتدا پورت B را خروجی ست میکنیم تا بتوانیم چراغ هشدار را در مواقع نیاز روشن کنیم. بعد در یک حلقه بی نهایت از سنسور اطلاعات دریافت میکنیم. در صورتی که جمع مقادیر دریافتی با check_sum برابر نباشد، به معنای وجود خطا در مقادیر است، به همین دلیل مقادیر را دوباره دریافت میکنیم. در صورت معتبر بودن مقادیر، آن را داخل LCD نمایش میدهیم.

سپس شرط خواسته شده را بررسی می کنیم. اگر رطوبت کمتر از 40 یا بیشتر از شود، پین صفرم از پورت B را 1 میکنیم تا LED روشن شود. در غیر این صورت به آن مقدار 0 میدهیم تا LED خاموش باشد.