گزارش کار آزمایش 5 قسمت الف

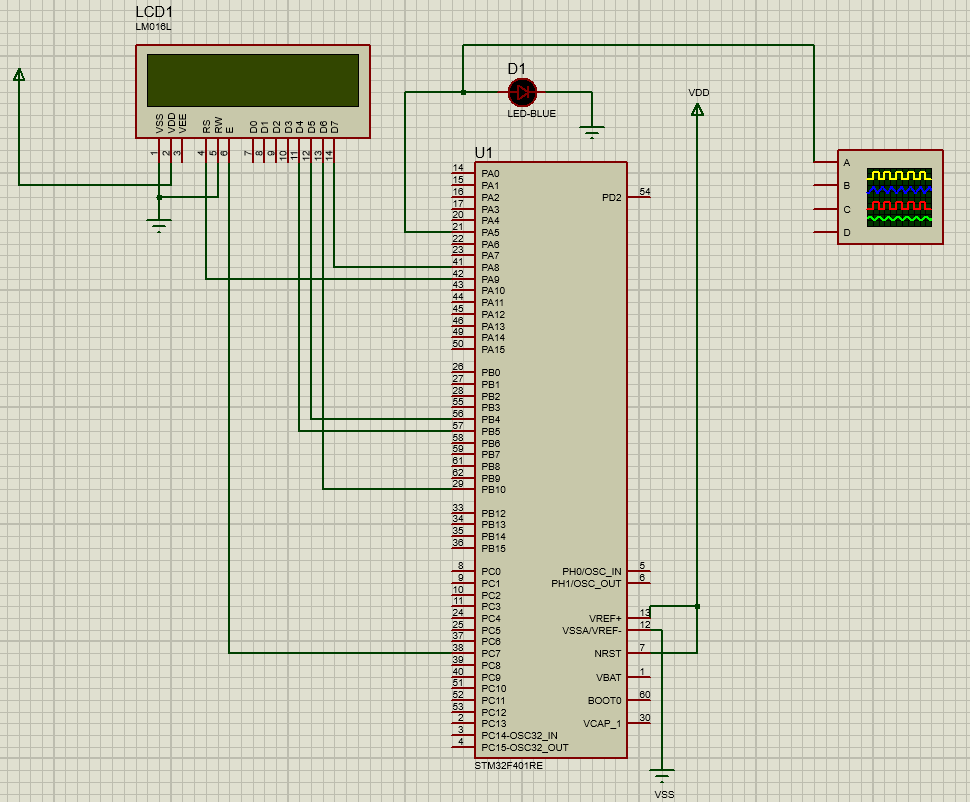
امین احسانی­مهر / شماره دانشجویی: 99243009

نگار هنرور صدیقیان / شماره دانشجویی: 99243076

سوالات تحلیلی:

1. همانطور که می­دانیم، شیلد LCD دارای 8 بیت دیتا برای منتقل کردن داده و یک پین RW برای خواندن/نوشتن، یک پین RS برای انتخاب command/data register، یک پین E (enable) یک پین VEE برای افزایش contrast صفحه نمایش است. تمامی این مقادیر باید به LCD داده شود. پس در نتیجه پورت­های nucleo باید پورت­های خروجی باشد.

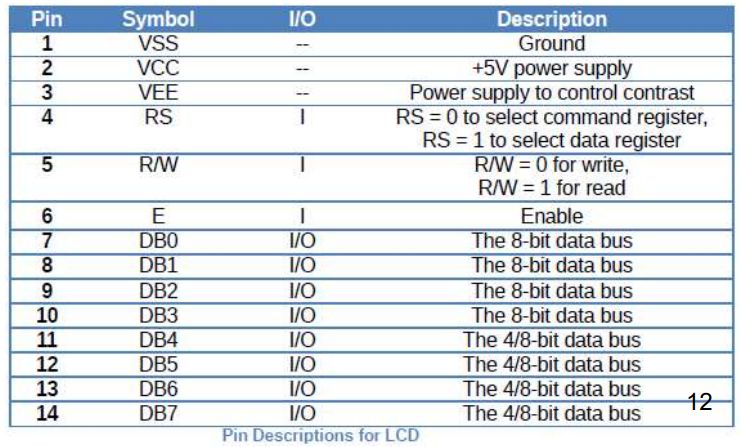
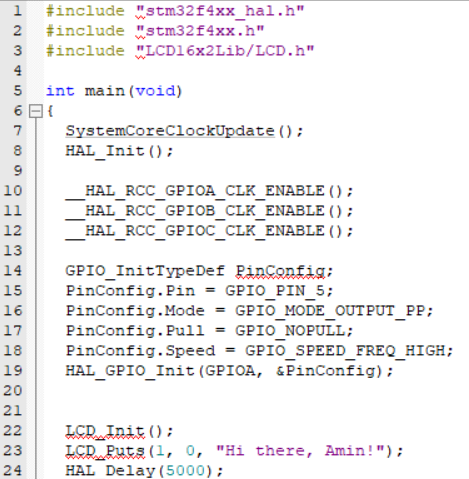
2. برای راه­اندازی LCD، ابتدا تابع SystemCoreClockUpdate را صدا می­کنیم که فرکانس میکروکنترولر را ست کند تا در ادامه از آن استفاده کند. سپس تابع HAL\_INIT() صدا می­شود تا بتوانیم در ادامه از توابع کتابخانه HAL استفاده کنیم. کلاک­های هر پورت را ست می­کنیم. سپس مشخص می­کنیم که کدام پورت مدنظر ما قرار است به عنوان خروجی از آن استفاده شود. در آخر با فراخوانی تابع LCD\_INIT() که باعث راه اندازی شیلد LCD می­شود و با تابع LCD\_PUTS()، مقدار مورد نظر خود را روی LCD نمایش می­دهیم.

گزارش کار:

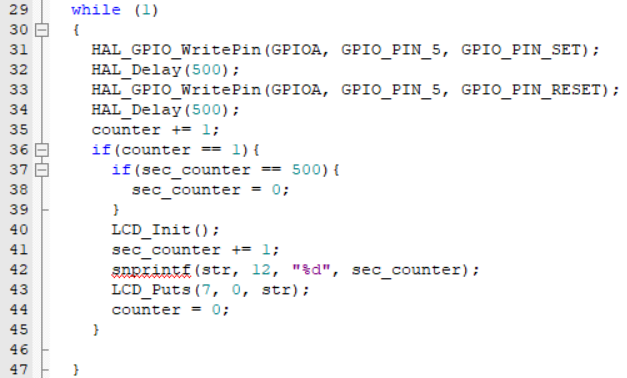
ابتدا از شماتیک شروع می­کنیم. قرار است از 4 بیت دیتای LCD استفاده کنیم. پس در نتیجه، از D4 تا D7 را به میکروکنترولر متصل می­کنیم و همینطور مقدار E و RW و RS و VEE را به آن وصل می­کنیم. همچنین VDD و VSS را به power و ground متصل می­کنیم. همچنین از یک بیت پورت A برای ایجاد سیگنال موج مربعی استفاده می­کنیم. این سیگنال را به oscilloscope متصل می­کنیم تا سیگنال خروجی را به صورت گرافیکی مشاهده کنیم.

شکل 1 – شماتیک مدار و نحوه اتصال میکروکنترولر به LCD و oscilloscope

پین­های LCD به صورت زیر تعریف می­شوند:

شکل 2 – پین­های شیلد LCD و کارکرد آن­ها

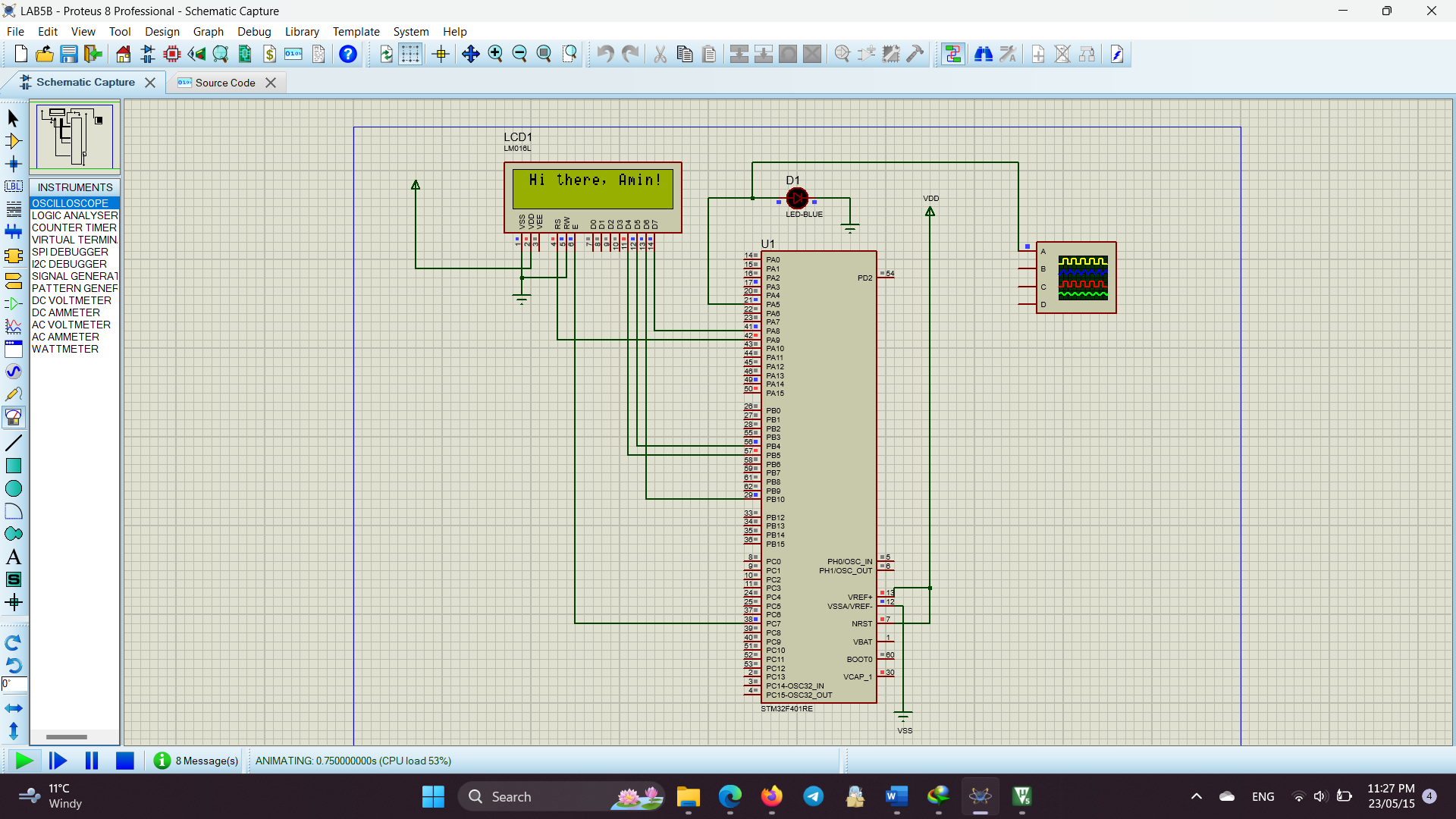
حال به سراغ کد می­رویم. درون برنامه پس از include کردن کتابخانه­های مورد نیاز و مواردی که در سوال دوم بخش تحلیلی ذکر شد، با دادن رشته مورد نظر به تابع LCD\_PUTS() و مشخص کردن محل شروع cursor، آن رشته را روی LCD نمایش می­دهیم. این مقدار تا 5 ثانیه باید روی صفحه باشد، پس با تابع HAL\_DELAY() یک دیلی 5000 میلی ثانیه­ای قرار می­دهیم.

حال در بخش بعد، باید یک سیگنال موج مربع با فرکانس 500 هرتز ایجاد کنیم و به ازای هر 500 پالس به وجود آمده، تعداد پالس در ثانیه که روی LCD نمایش داده می­شود را آپدیت کنیم. پس به ازای هر 500 پالس به وجود آمده، یک واحد روی LCD اضافه می­کنیم. برای اینکار، یک متغیر counter داریم که وظیفه آن بررسی آن است که 500 پالس انجام شده است یا خیر. اگر بود، مقدار آن را 1 می­کنیم تا وارد حلقه شرط بشویم تا عدد مدنظر را روی LCD نمایش بدهیم. این مقدار که با متغیر sec\_counter مقداردهی شده است، اگر از حدی (که در اینجا برابر 500 قرار دادیم) بیشتر شد، این مقدار را reset می­کنیم.

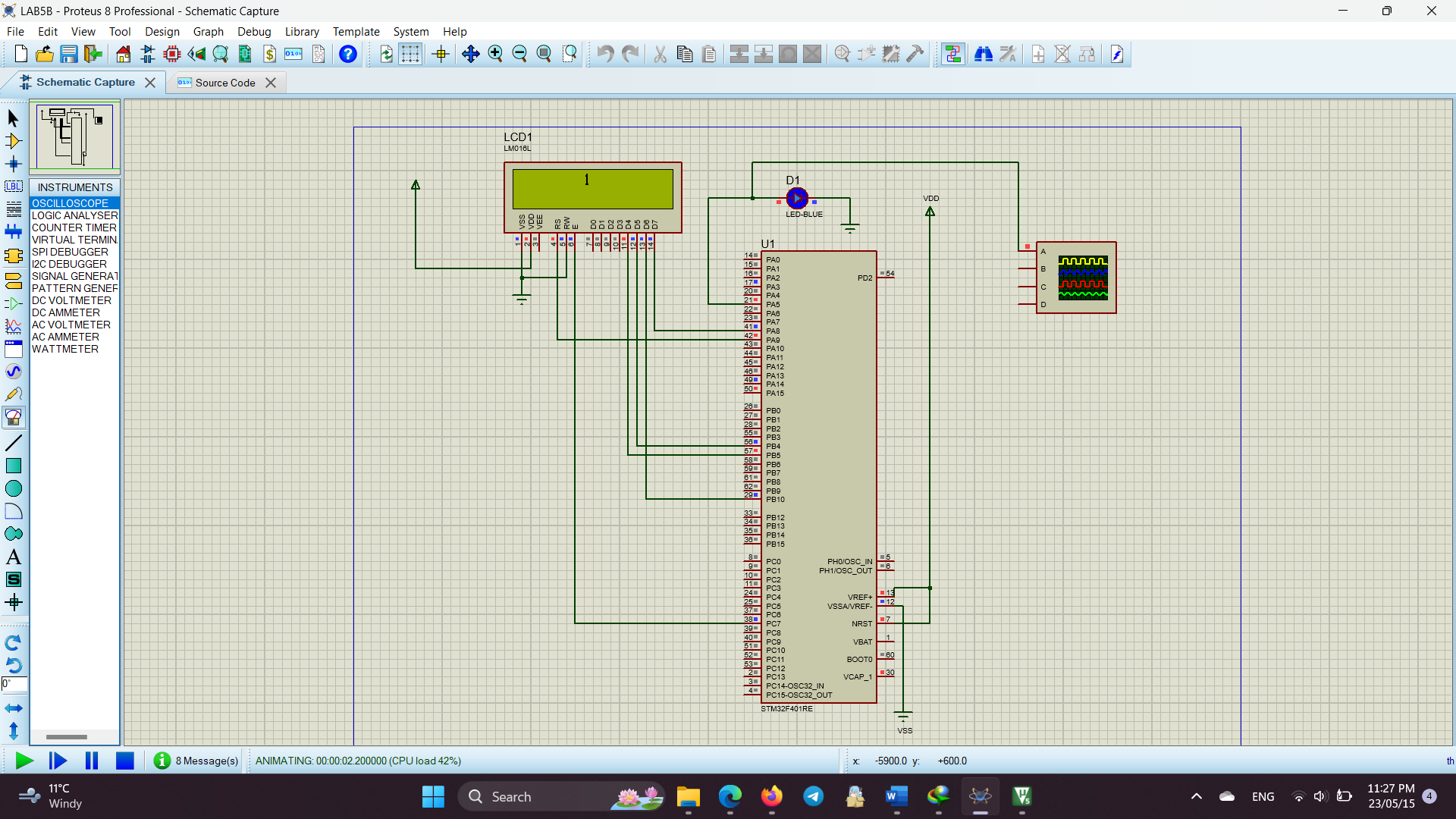
شکل 4 – نمایش دادن تعداد پالس زده شده در ثانیه / 500

شکل 3 – فراخوانی SystemCoreClockUpdate و HAL\_Init() و مشخص کردن کلاک پروت­ خروجی بودن پین A و در آخر، نمایش خوش­آمدگویی خواسته شده

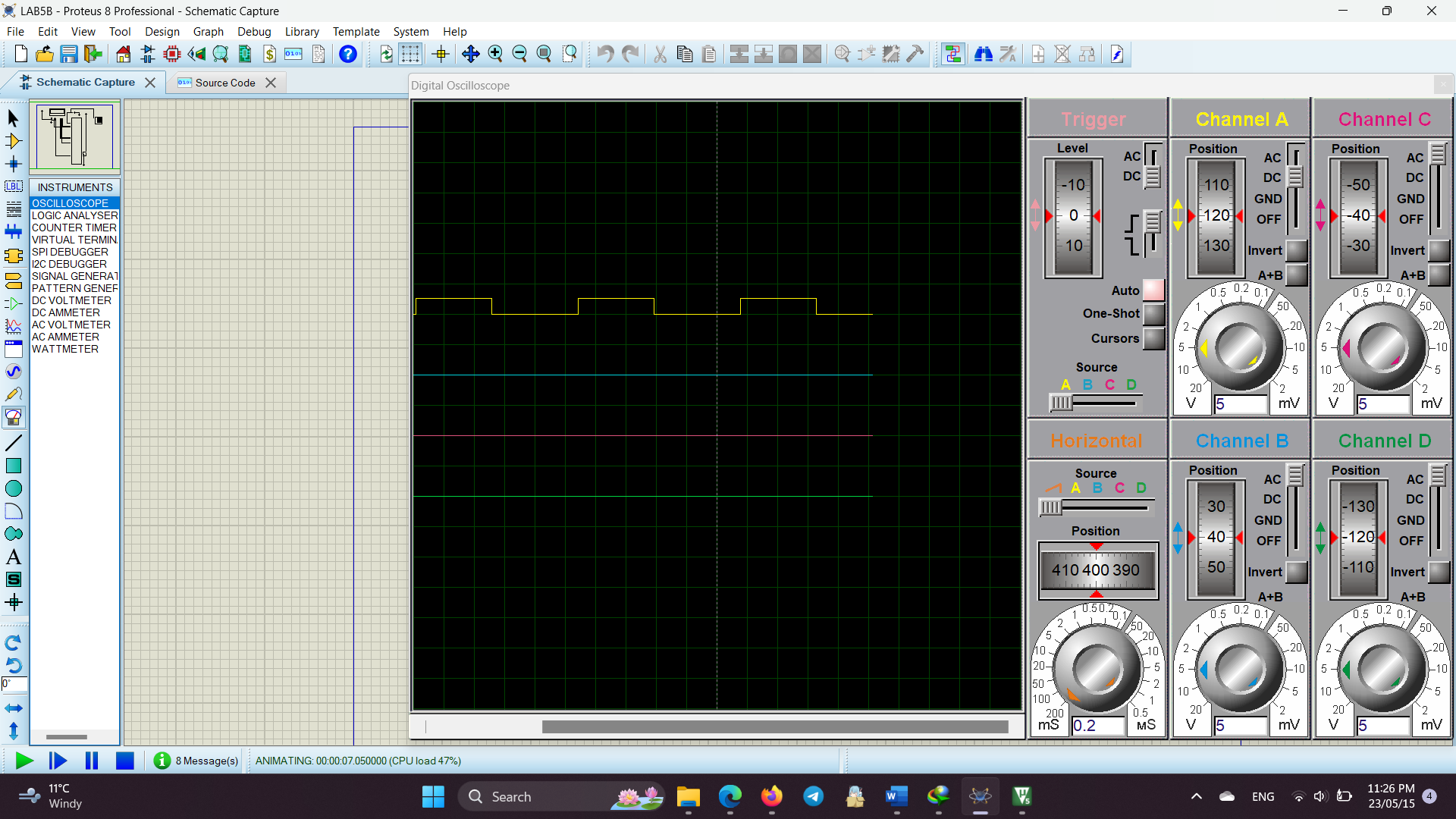
مدار بسته شده را در proteus شبیه­سازی می­کنیم. خروجی به دست آمده به صورت زیر خواهد شد:



شکل 5 – نمایش خوش­آمدگویی



شکل 6 – تعداد پالس­های انجام شده در واحد زمان



شکل 7 – پالس مربعی تولید شده، نمایش داده شده بر روی oscilloscope