

در تصویر بالا با فعال کردن تیک مربوط به پکیج و زدن گزینه Install now پکیج مورد نظر دانلود و نصب خواهد شد. (میتوانید ورژن جدیدتر 1.8.6 را دانلود کنید).

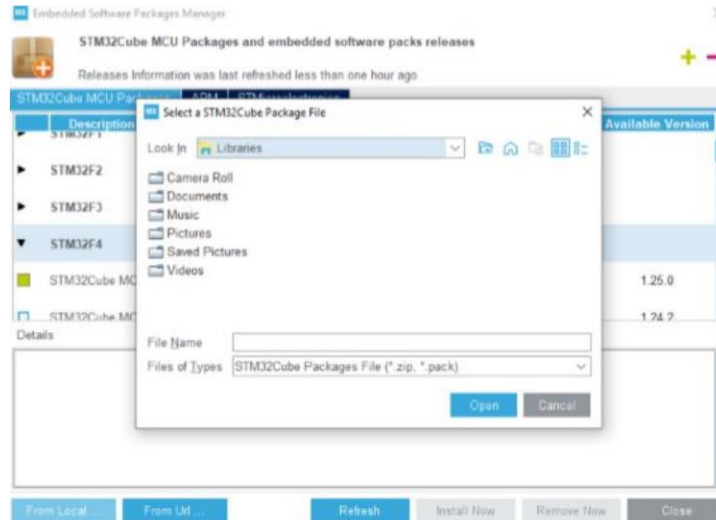
روش دوم)

نرم افزار را از سایت [soft 98](#) دانلود و نصب کنید.

سپس پکیج STM32F1 را از سایت [sisoog](#) دانلود کنید. برای اینکار در سایت گفته شده فایل "دانلود پکیج F1" را دانلود کنید. (شکل زیر)



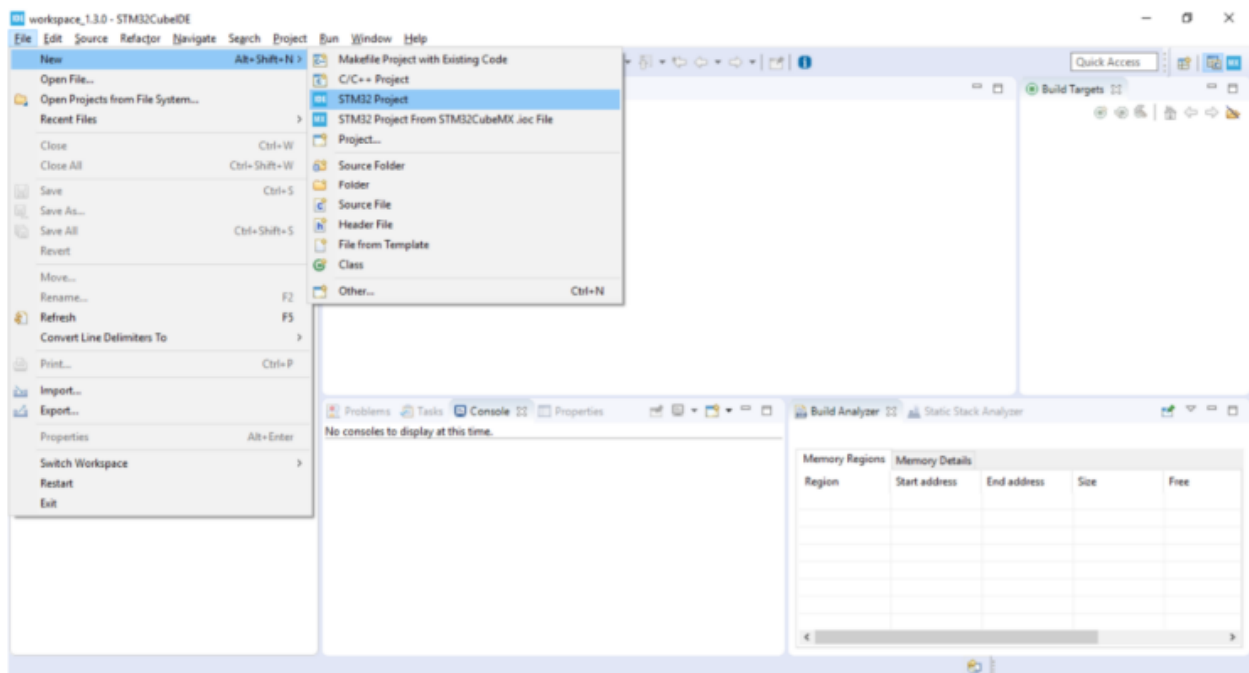
بعد از دانلود پکیج، همانند روش قبلی به قسمت Manage embedded software package برید و از قسمت From Local فایل دانلود شده را انتخاب کنید.



حال نرم افزار آماده ی ساخت پروژه میباشد.

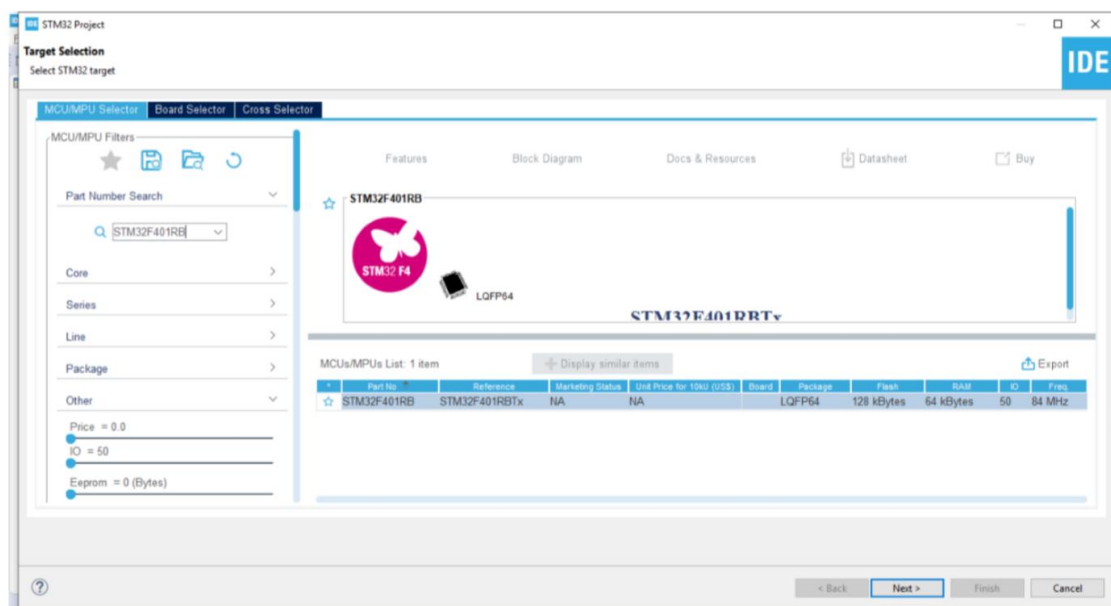
❖ ساخت پروژه

از مسیر شکل زیر یک پروژه جدید بسازید. (جهت افزایش سرعت نرم افزار، پیشنهاد میشود که در این قسمت اینترنت خود را خاموش کنید)

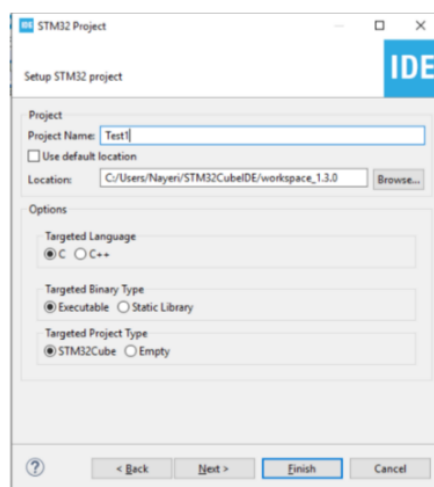


سپس صفحه ای به صورت شکل زیر باز خواهد شد. در قسمت Part Number Search میکروکنترلر مطابق جدول زیر را تایپ کرده و در پنجره وسط آن را انتخاب کنید و بر روی Next کلیک کنید.

| میکروکنترلر | باقیمانده شماره دانشجویی بر ۳ |
|-------------|-------------------------------|
| STM32F103C6 | ۰ |
| STM32F103R6 | ۱ |
| STM32F103T6 | ۲ |

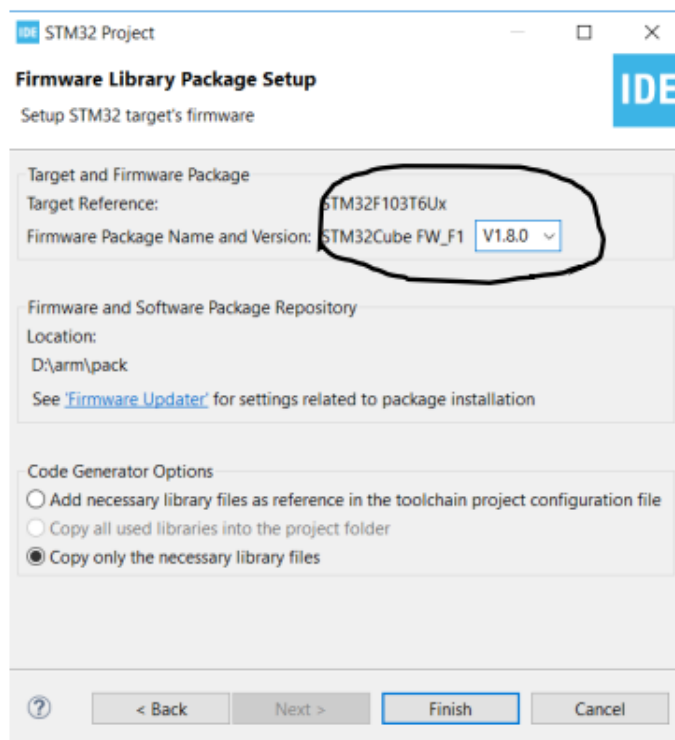


سپس پنجره ای به صورت شکل زیر باز خواهد شد که در آن میبایست یک نام برای پروژه انتخاب کرده و سپس مسیری را که پروژه در آن ذخیره می شود انتخاب کنید



توجه: پس از انتخاب نام و آدرس پروژه بر روی Next کلیک کنید. در قسمت علامت زده ی شکل زیر دقت کنید که باید دقیقا همان ورژن پکیجی انتخاب شود که بر روی نرم افزار داللود شده است. که اگر از روش دوم در

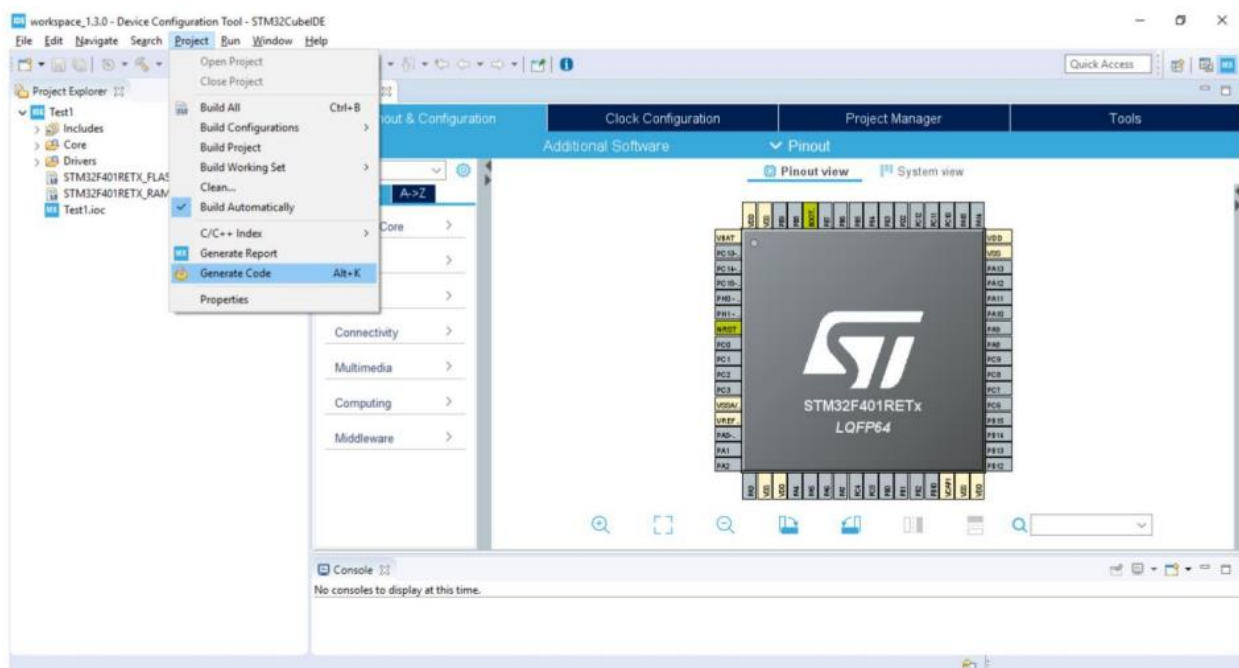
نصب نرم افزار استفاده کرده باشید باید 1.8.0 را انتخاب کنید در غیر این صورت پروژه با موفقیت ساخته نمیگردد.



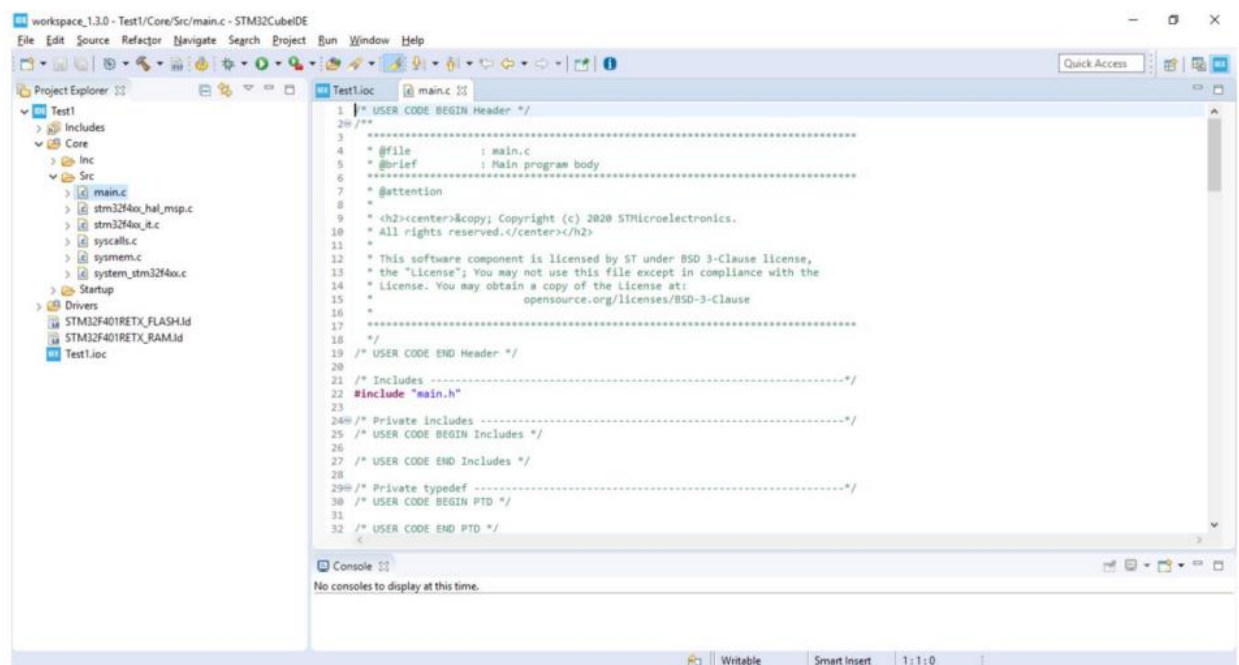
سپس بر روی Finish کلیک کنید. در ادامه محیطی مثل نرم افزار CubeMX ظاهر خواهد شد که به راحتی میتوانید تنظیمات مربوط به بخش های مختلف میکروکنترلر را روی آن انجام دهید.



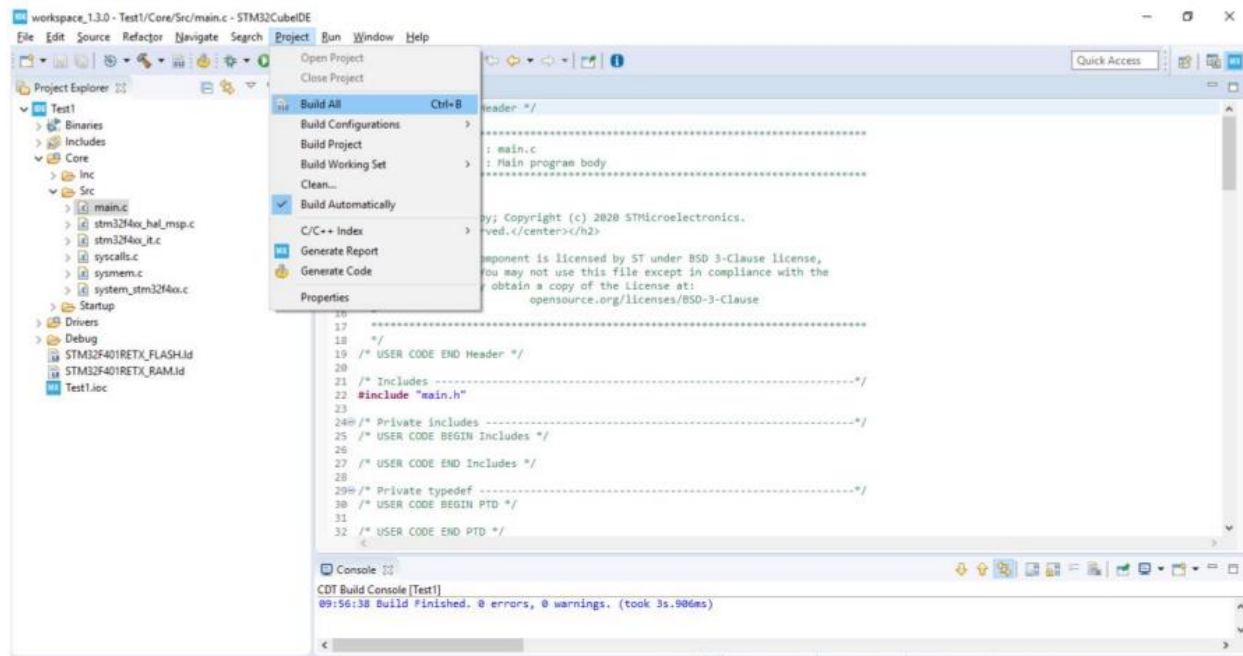
پس از انجام تغییرات مورد نظر، همانند شکل زیر بر روی Generate Code کلیک کنید تا فایل های مربوط به برنامه نویسی میکروکنترلر تولید شود.



سپس مطابق شکل زیر از سمت چپ فایل main.c را انتخاب کنید. صفحه ای مطابق شکل زیر باز خواهد شد که بدنه اصلی برنامه ست و میتوانید بر روی آن کدهای خود را بنویسید.

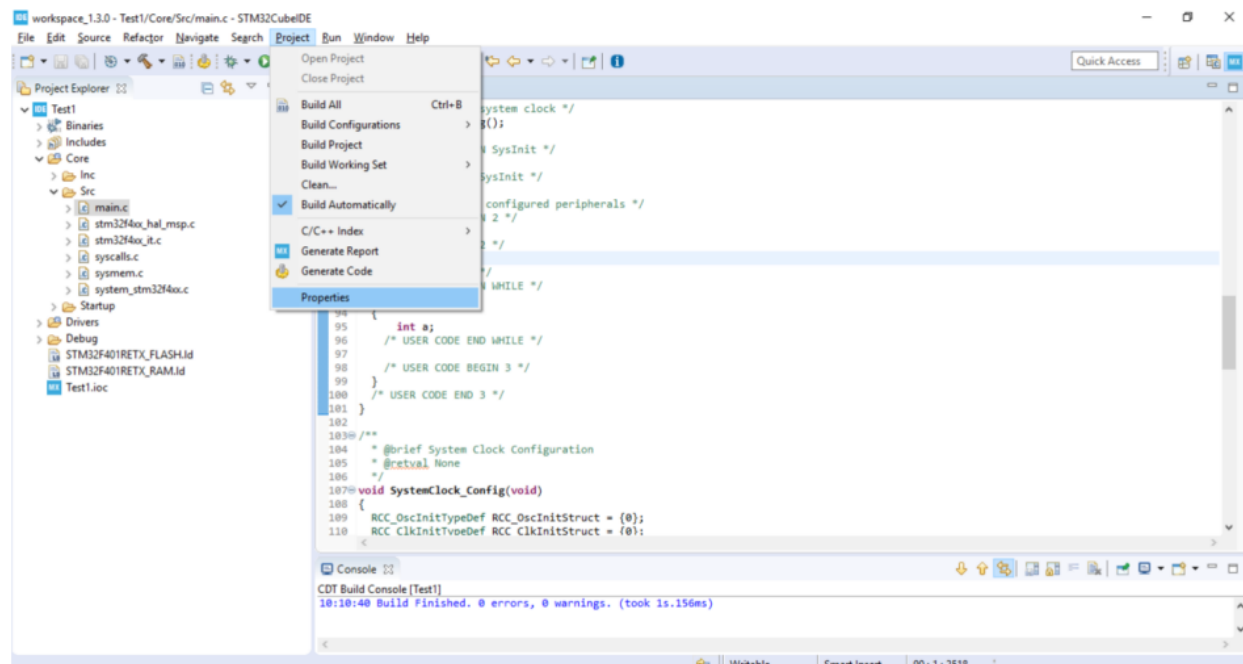


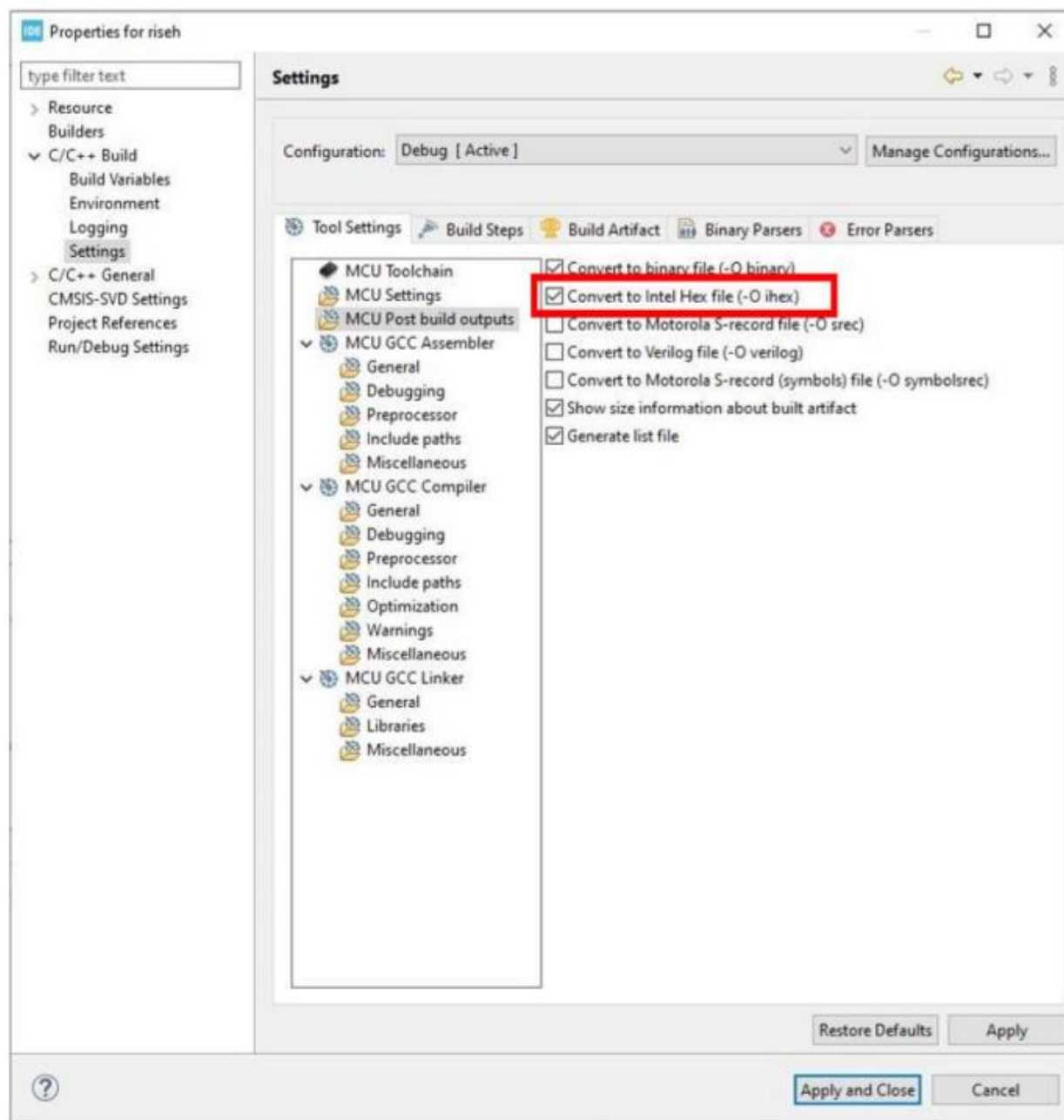
بعد از نوشتن کد های مدنظرتان، همانند شکل زیر باید کد های مد نظر را Build all کنید تا از خطا ها و ارور های احتمالی که در باکس console مشاهده میشود مطلع گردید.



❖ ساخت فایل hex.* جهت استفاده در پروتئوس

مطابق شکل های زیر عمل کنید:





پس از انتخاب گزینه داخل باکس عکس بالا، Apply and Close را بزنید و برنامه را دوباره Build All کنید.

برای دسترسی به فایل *.hex کافیست که به آدرس زیر در محل پروژه خود بروید. (در پوشه Debug)

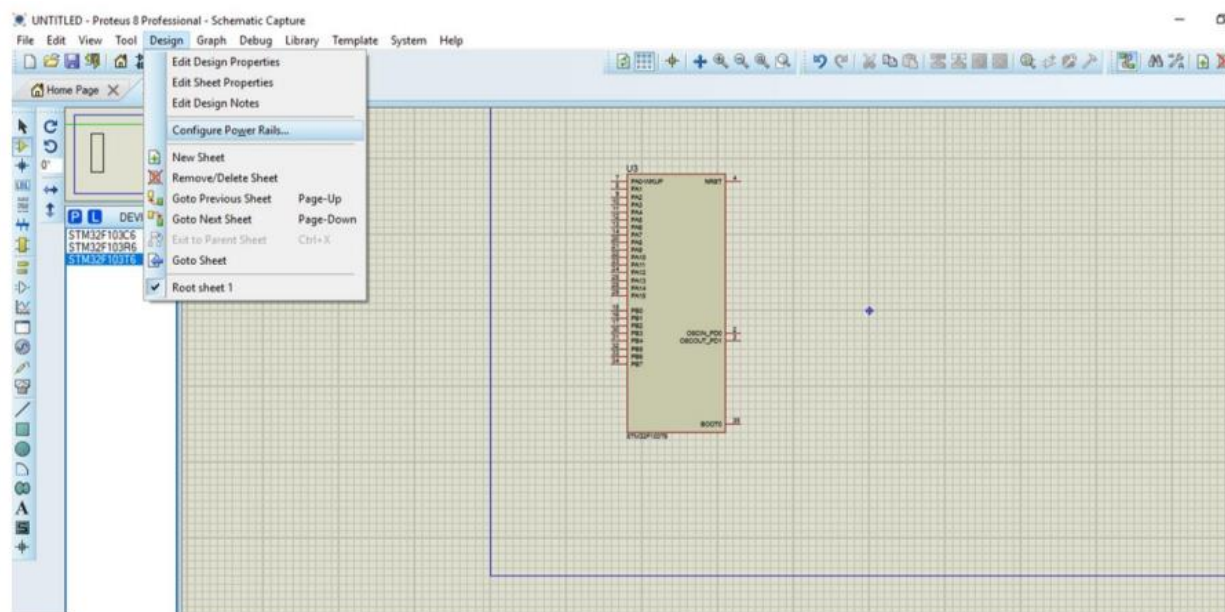
| C:\Users\Nayeri\STM32CubeIDE\workspace_1.3.0\Test1\Debug | | | | |
|----------------------------------------------------------|--------------------|-------------|--------|--|
| Name | Date modified | Type | Size | |
| Core | 7/16/2020 9:56 AM | File folder | | |
| Drivers | 7/16/2020 9:56 AM | File folder | | |
| makefile | 7/16/2020 10:10 AM | File | 3 KB | |
| objects.list | 7/16/2020 10:10 AM | LIST File | 2 KB | |
| objects.mk | 7/16/2020 9:56 AM | MK File | 1 KB | |
| sources.mk | 7/16/2020 10:10 AM | MK File | 1 KB | |
| Test1.bin | 7/16/2020 10:03 AM | BIN File | 5 KB | |
| Test1.elf | 7/16/2020 10:03 AM | ELF File | 750 KB | |
| Test1.hex | 7/16/2020 10:10 AM | HEX File | 13 KB | |
| Test1.list | 7/16/2020 10:03 AM | LIST File | 112 KB | |
| Test1.map | 7/16/2020 10:03 AM | MAP File | 236 KB | |

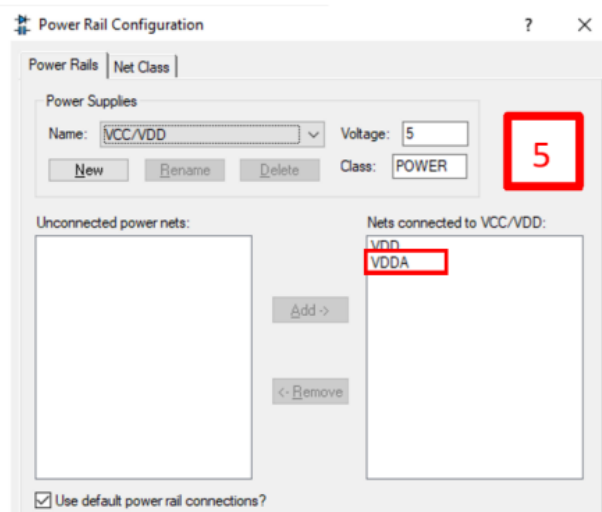
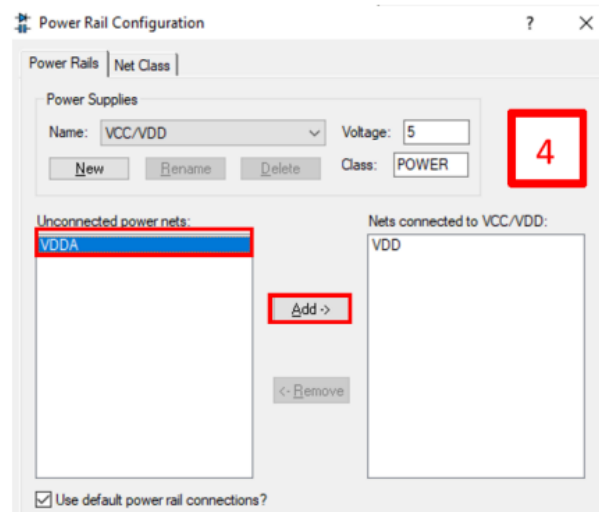
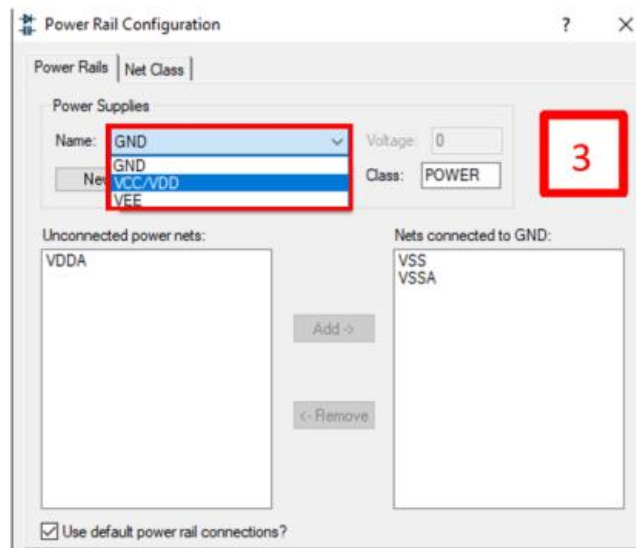
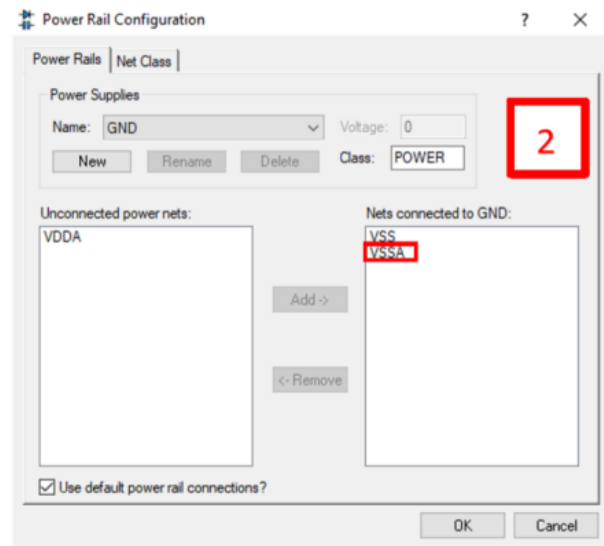
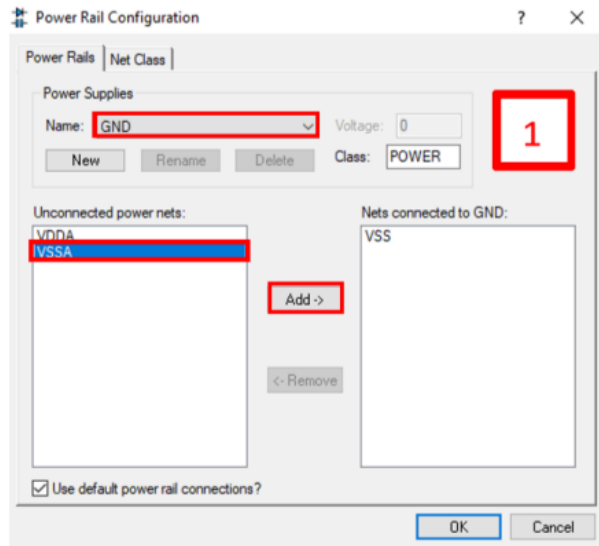
❖ راه اندازی نرم افزار پروتئوس

نرم افزار پروتئوس را نصب کنید (ترجیحا ورژن های بالاتر از 8.8)

توجه: باتوجه به اینکه این نرم افزار کرک شده است و بسته به کامپیوتر شما و نوع کرک، ممکن است پس از مدتی که نرم افزار باز است، خود به خود بسته شود بنابراین پس از هر تغییر در محیط شبیه سازی بلافاصله پروژه را Save کنید تا تغییرات از دست نروند.

برای شبیه سازی نیاز است که پایه های VDDA و VSSA را به ترتیب به تغذیه و زمین میکروکنترلر متصل کنید. برای این منظور مطابق شکل های زیر عمل کنید.





ابتدا مقدار کلاک اصلی میکرو کنترلر را به صورت زیر تنظیم کنید.

| باقیمانده شماره دانشجویی بر ۹ | $HCLK(MHz)$ |
|-------------------------------|-------------|
| ۰ | 10 |
| ۱ | 15 |
| ۲ | 20 |
| ۳ | 26 |
| ۴ | 30 |
| ۵ | 36 |
| ۶ | 40 |
| ۷ | 44 |
| ۸ | 52 |
| ۹ | 60 |

سوال 1

میخواهیم یک چراغ راهنمایی درست کنیم که دارای 3 وضعیت قرمز، زرد و سبز است. با اتصال یک Push Button به صورت Pull Up به میکرو و سه LED به رنگ های گفته شده برنامه ای بنویسید که با هر بار فشردن کلید رنگ چراغ به وضعیت بعدی برود. برای این منظور از دستورات HAL_GPIO_ReadPin و HAL_GPIO_WritePin استفاده کنید. (10 نمره)

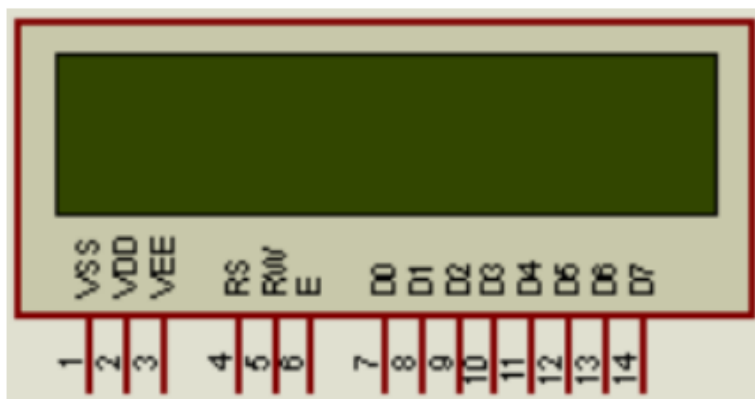
برنامه های میکروکنترلر (main.c , *.hex , *.ioc) را به همراه برنامه پروتئوس (pdsprj). در پوشه Q1 ذخیره کنید.

سوال 2

در این سوال قصد داریم که یک تایمر را به صورت mm:ss راه اندازی کنیم که دارای دو کلید شروع و پایان است. برای راه اندازی آن مراحل زیر را اجرا کنید:

- 1- ابتدا به دو پایه ی میکرو دو ورودی شروع و پایان را میدهیم. برای این مدار از دو Push Button که خروجی آن Pull Down میشود استفاده میکنیم. زمانی که کلید اول فشرده میشود تایمر باید شروع و با فشردن کلید دوم باید توقف یابد. دقت شود که برای پیاده سازی آن باید از Interrupt استفاده شود. در مورد نحوه استفاده از آن تحقیق و با توجه به آن این بخش را پیاده سازی کنید.

- 2- برای شمارش تایمر نیاز است که Timer داخلی میکرو را فعال کنیم. با توجه فرکانس کلاک، Timer را بر روی 1 ثانیه فعال کنید. همچنین دقت کنید برای شمارش نیاز است که Interrupt داخلی Timer فعال شود تا با هر دوره اتمام Timer زمان یک ثانیه افزایش یابد. (از تاخیر حاصل از تابع HAL_Delay صرف نظر شود)
- 3- حال برای نمایش زمان نیاز است که میکرو به یک LCD متصل شود. با اتصال LCD Alphanumeric 2*16 به میکرو زمان را در خط اول آن نمایش و در خط دوم شماره دانشجویی خود را نمایش بدهید. (60 نمره)



برنامه های میکروکنترلر (main.c , *.hex , *.ioc) را به همراه برنامه پروتئوس (pdsprj) در پوشه Q2 ذخیره کنید.

سوال 3

در این سوال میخواهیم با استفاده از ADC میکروکنترلر از یک سیگنال سینوسی نمونه برداری کنیم:

الف) در مورد واحد ADC تحقیق کنید و چگونگی اضافه کردن آن به stm32 را بررسی کنید. یک پایه برای ADC برای گرفتن ورودی آنالوگ (سیگنال سینوسی) تعیین کنید.

ب) برنامه ای بنویسید که با استفاده از ADC سیگنال سینوسی را نمونه برداری کند و مقادیر نمونه برداری شده را با دو رقم اعشار در سطر اول یک LCD Alphanumeric 2x16 بر حسب ولتاژ نمایش و در سطر دوم شماره دانشجویی خود را نمایش دهید.

د) مدار مورد نظر شامل stm32 و LCD را در پروتئوس بسته و با اعمال یک سیگنال سینوسی به عنوان ورودی ADC (به صورتی که دامنه بیش از 5 ولت نداشته باشد و شامل مقادیر منفی ولتاژ نشود)، ولتاژهای خروجی نمونه برداری شده را روی LCD نمایش دهید. (30 نمره)

برنامه های میکروکنترلر (main.c , *.hex , *.ioc) را به همراه برنامه پروتئوس (pdsprj) در پوشه Q3 ذخیره کنید.

سوال 4) در این سوال میخواهیم نحوه کار با پریفرال TIM و تولید سیگنال PWM با استفاده از آن را بررسی کنیم:

الف) در مورد واحد TIM تحقیق کنید و چگونگی اضافه کردن آن به stm32 را بررسی کنید. سپس با استفاده از آن یک سیگنال PWM با فرکانس 100 هرتز (با تغییر پارامترهای تایمر مثل ARR و PSC) و دیوتی سیکل 40٪ تولید کرده و در پروتئوس نمایش دهید.

ب) در این قسمت میخواهیم با استفاده از ADC که در سوال قبل کار با آن را آموختیم، دیوتی سیکل سیگنال PWM قسمت الف را تغییر دهیم. برای این کار برای ورودی ADC متغیر، از یک پتانسیومتر در پروتئوس استفاده کنید و سپس با تغییر مقدار پتانسیومتر، دیوتی سیکل را از 10٪ تا 90٪ تغییر داده و نمایش دهید. (دقت کنید ماکسیمم ولتاژ ورودی stm32 در پروتئوس 5 ولت است.)

برنامه های میکروکنترلر (main.c , *.hex , *.ioc) را به همراه برنامه پروتئوس (pdsprj) در پوشه Q4 ذخیره کنید.

لطفا در ارسال تمرینات به موارد زیر توجه بفرمایید ، در صورت عدم رعایت هر یک از موارد زیر تمرین شما تصحیح نخواهد شد :

- تشابه در حل سوالات به صورت جدی بررسی خواهد شد. در صورت تشخیص تمرین مشابه نمره تقسیم خواهد شد.
 - در صورت دست نویس بودن تمرین ، نوشته ها خوانا باشند و کیفیت اسکن آن ها مناسب باشد.
 - پاسخ ها در قالب یک فایل pdf جمع و ارسال شوند. همچنین تمامی کد ها با ذکر اینکه مربوط به کدام سوال هستند در پوشه ای با نام Codes ذخیره شده سپس تمامی فایل ها در قالب یک فایل zip جمع و با نام student_number.zip ارسال شوند .
- در صورت هر گونه ابهام در سوالات برای هر یک از سوالات به دستیار مربوطه تنها از طریق ایمیل دانشگاهی با موضوع Series#-Q# (که در آن # شماره سری تمرین و سوال مورد نظر است) ایمیل شود.
- سوال 1و2) آقای بیان – bayan.mahdiyar@ut.ac.ir
- سوال 3و4) خانم صالحی – [sasabasalehi@ut.ac.ir](mailto:sabasalehi@ut.ac.ir)