۲. هدف این بخش تمرین، استفاده از ابزار Fixed-Point Designer برای جایگزینی نوع‌های ممیز شناور با نوع‌های ممیز ثابت مناسب در مدل روبات توسعه داده شده تمرین ۵ و مقایسه زمان اجرای کدهای تولید شده از آن‌ها است.

أ. کد کنترل کننده مدل فوق را در قالب یک تابع برای برد Arduino Mega2560 تولید کنید. تابع تولید شده را با استفاده از یک پروژه PlatformIO کامپایل کرده و با اجرای تابع step سطح بالا برای چندین نوبت و میانگین‌گیری، زمان اجرای تابع را در شبیه‌ساز Proteus برای سطوح مختلف بهینه‌سازی کامپایلر (O0-، O1-، O2-، O3- و Os-)جداگانه اندازه‌گیری کنید. از پروژه خود تا به اینجا یک نسخه پشتیبان تهیه کنید.

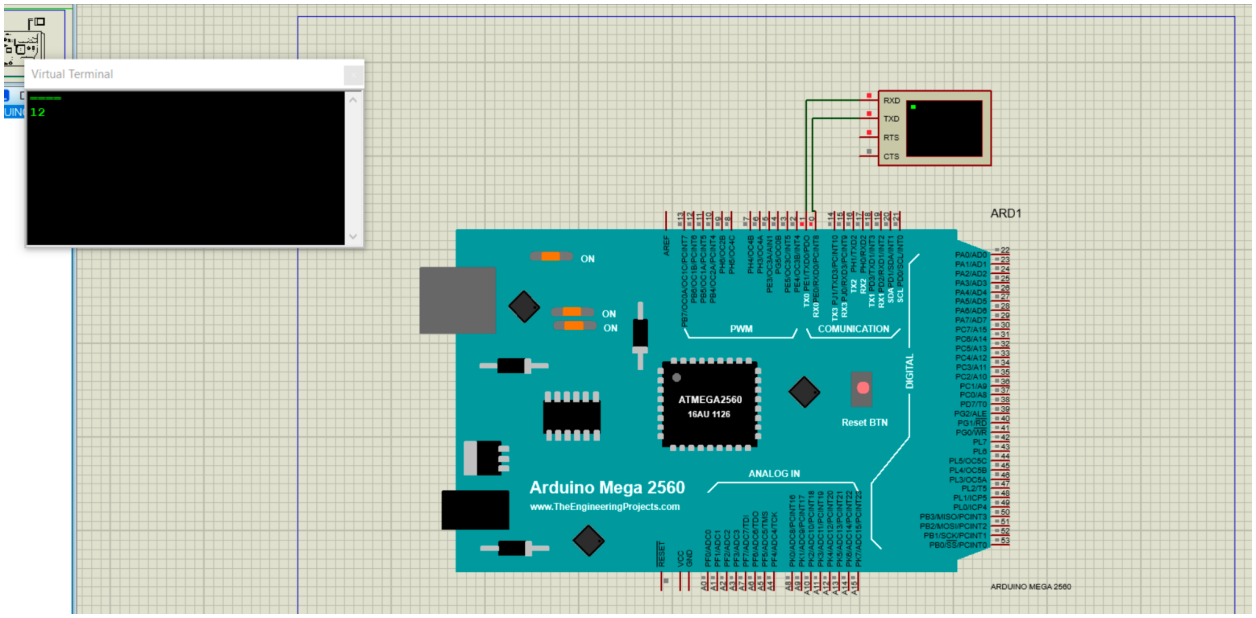
در ابتدا، یک پروژه platformio میسازیم و کد جنریت شده را در main قرار میدهیم. (توجه شود که باید تابع setup را خودمان اضافه بکنیم)

تابع setup را به صورت دستی به کد اضافه میکنیم. لازم از تا تابع پله را ۱۰۰ بار اجرا کنیم البته همانطور که در صورت سوال نیز مطرح شده است باید تا با استفاده از تابع Micros زمان اجرا را اندازه گیری بکنیم. البته این اندازه گیری کافی نیست و هدف از اندازه گیری به دست آوردن میانگین است. که این مقدار را حساب کرده و در خروجی نمایش میدهیم.

در ادامه تابع پیاده سازی شده آمده است:

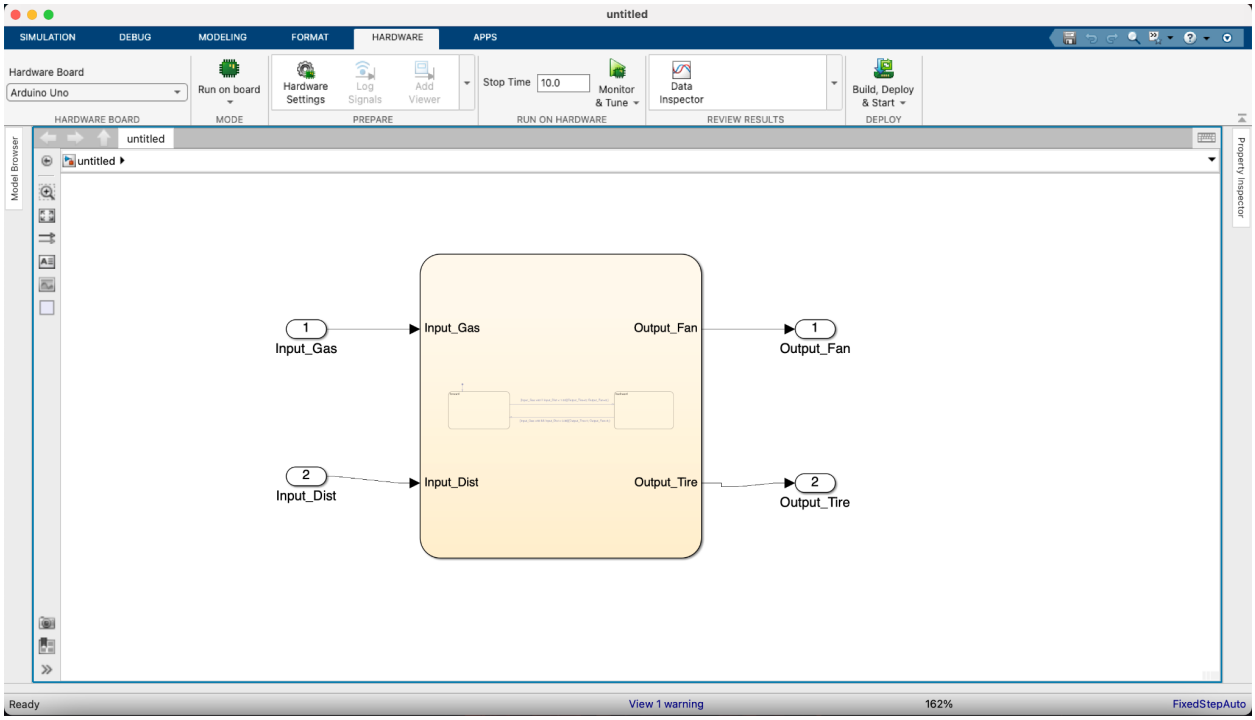
```
void setup() {
        Serial.begin(9600);
        untitled_initialize();
        unsigned long sum = 0;
        int count = 100;
        for (int i = 1; i <= count; ++i){
          unsigned long start = micros();
          untitled_step();
          unsigned long end = micros();
          unsigned long delta = end - start;
          sum += delta;
        }
        Serial.println((sum/count));
}
```

به ۶ روش خواسته شده کامپایل میکنیم و در تمام حالت ها خروجی زیر تولید شده است. (تفاوتی نداریم)

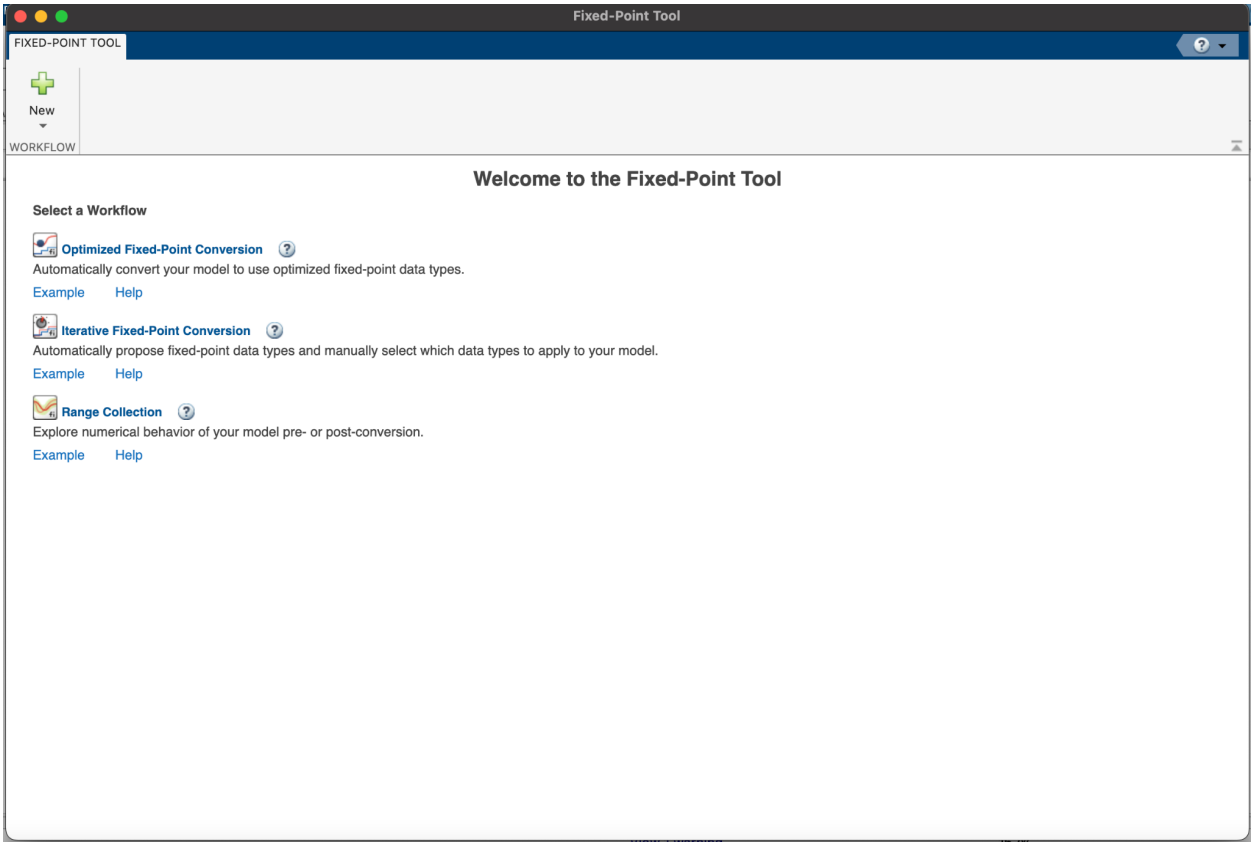


ب. با استفاده از ابزار Fixed-Point Designer[12]، عملیات زیر را برای کنترل‌کننده انجام دهید. ممکن است نیاز باشد برخی بلوک‌های خروجی را موقتا از مدل خارج کنید.

- آماده‌سازی مدل برای آنالیز (درج خودکار ورودی/خروجی‌ها برای تبدیل نوع)؛
- انجام شبیه‌سازی برای جمع‌آوری رنج و دقت سیگنال‌ها؛
- پیشنهاد داده‌های نوع ممیز ثابت به‌جای نوع‌های ممیز شناور؛ و
- اعمال تغییرات در مدل و جایگزینی نوع‌های جدید در مدل.

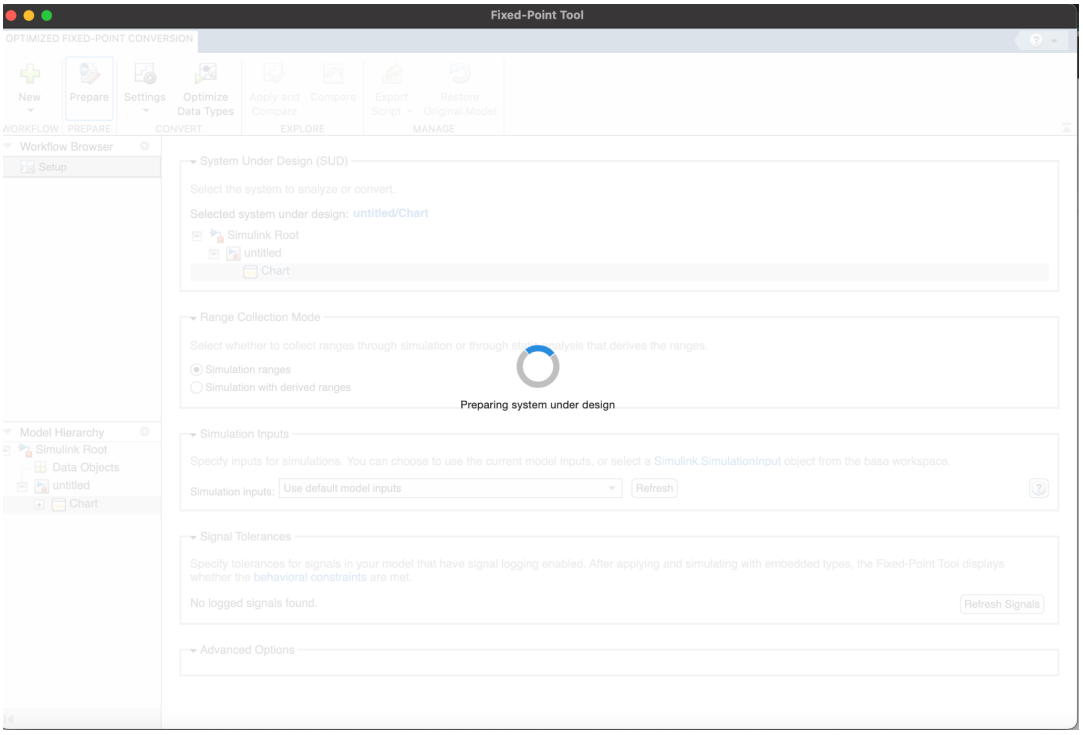


در این بخش سوال نیاز است تا از ابزاری به نام Fixed Point Designer استفاده کنیم. در این مرحله به صورت تصویری تمام مراحل مورد نیاز را نمایش داده ایم و دست آخر با استفاده از ابزاری به نام Ccoder کد جدید ایجاد شده را Build کرده ایم.

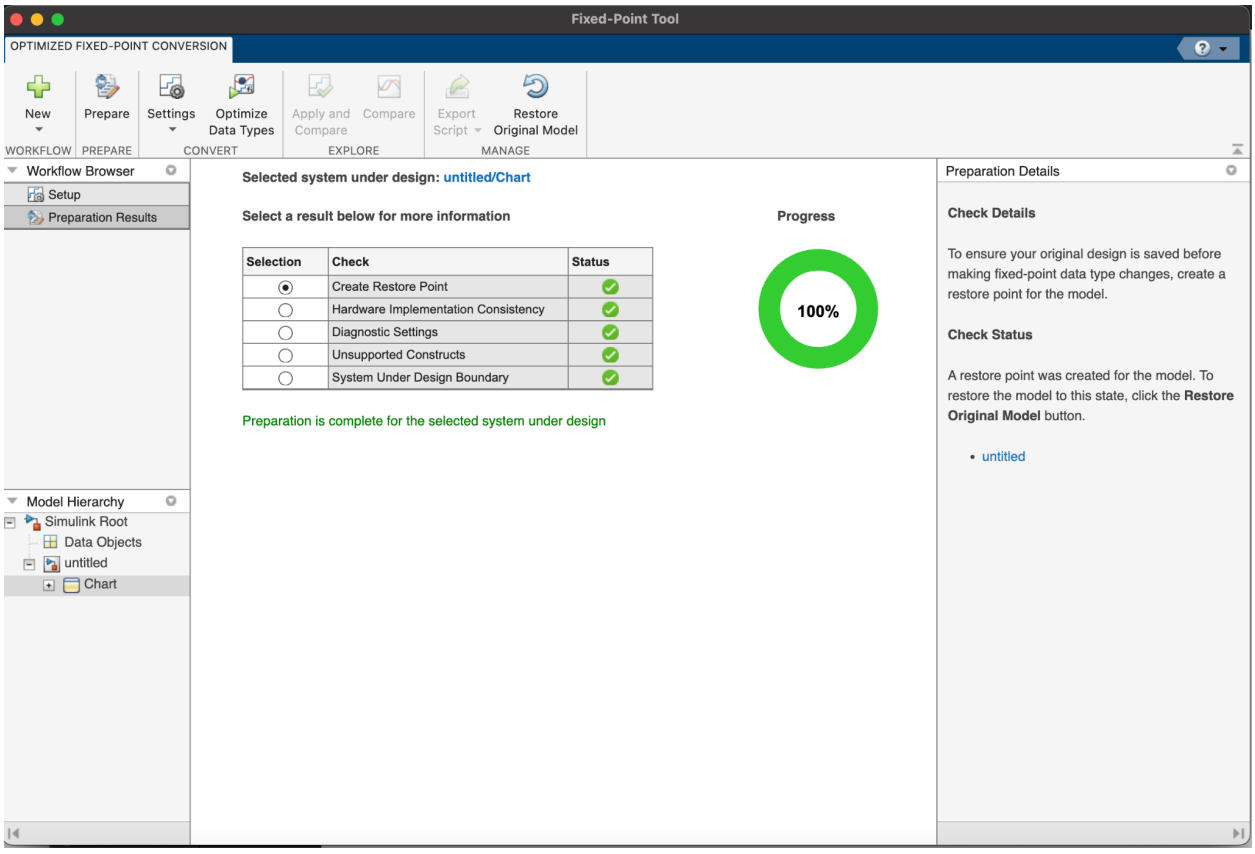


در مرحله اول بر روی Subsystem ای که قصد داریم بر روی آن ابزار Fixed Point Designer را اجرا کنیم انتخاب کرده و سپس با کلیک راست و انتخاب ابزار مورد نظر به صفحه فوق میرویم.

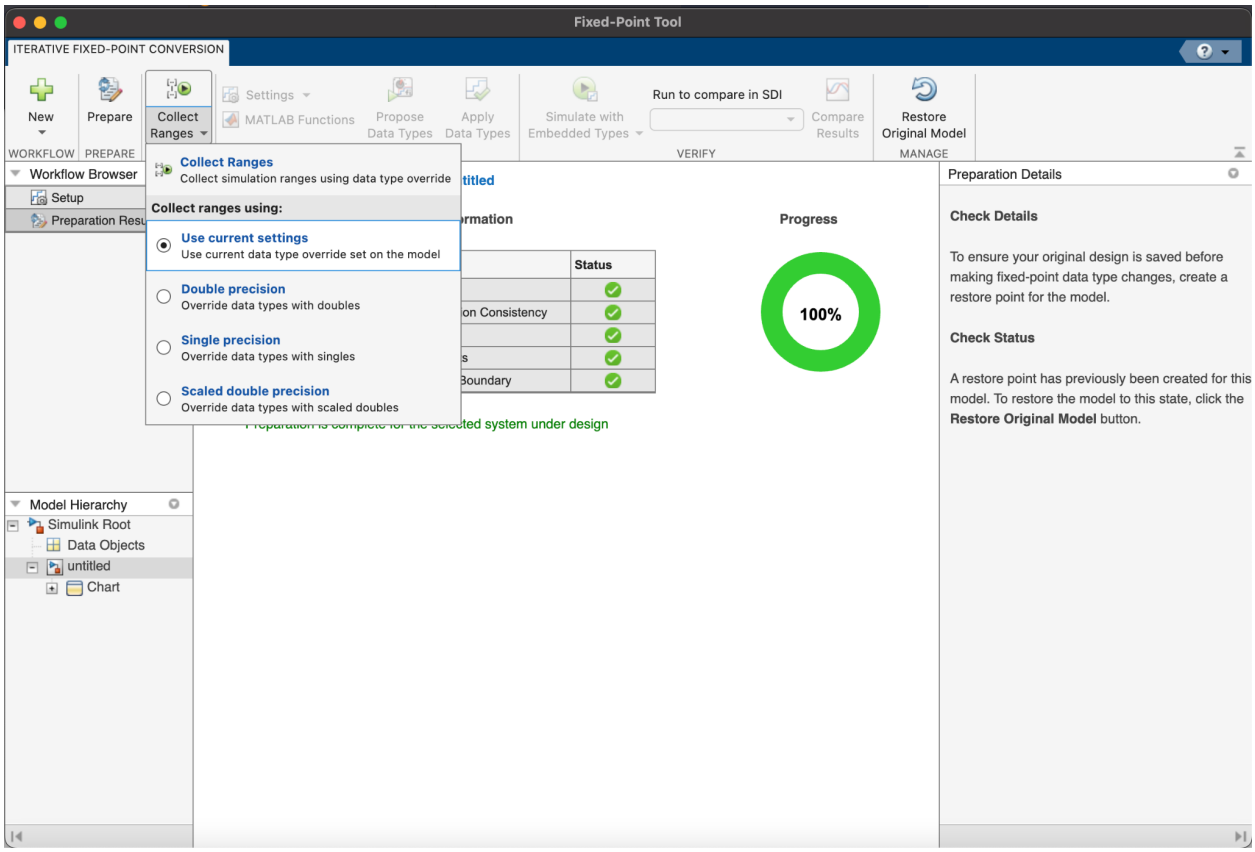
طبق تصویر بالا روند مورد نظر خود را انتخاب کرده و ادامه می دهیم. در مرحله بعدی باید مدل مورد نظر خود را آماده سازی کنیم (که با زدن دکمه Prepare) این اتفاق می افتاد.

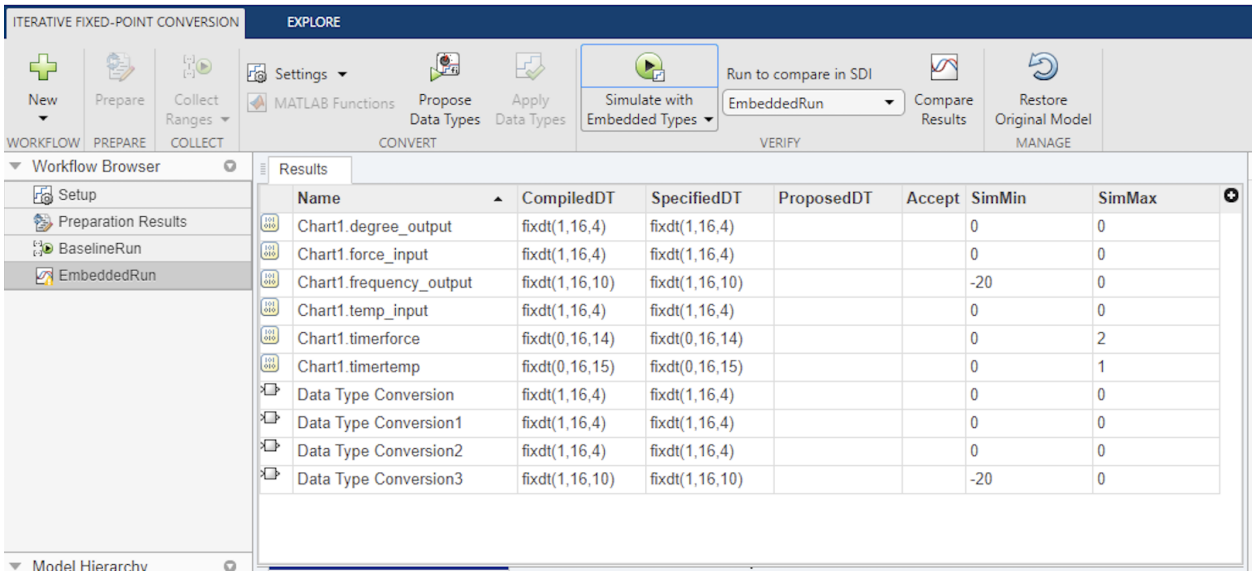


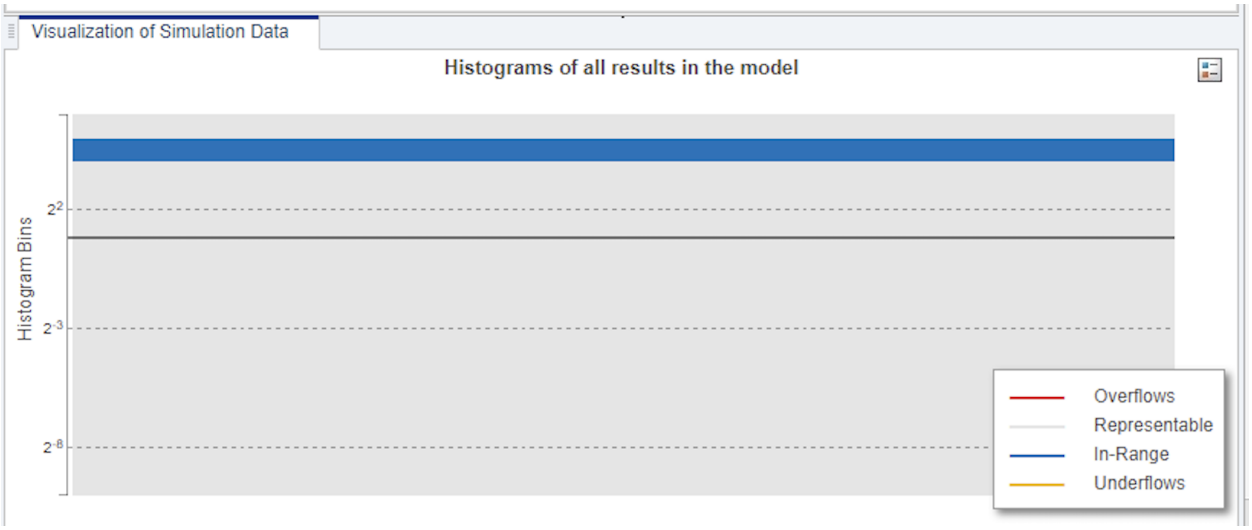
مدتی برای آماده سازی باید صبر کنیم تا در نهایت به مرحله زیر برسیم:



همانطور که مشخص است مدل کاملا آماده است. تا به اینجای کار مرحله اول ذکر شده در صورت سوال انجام شده است.

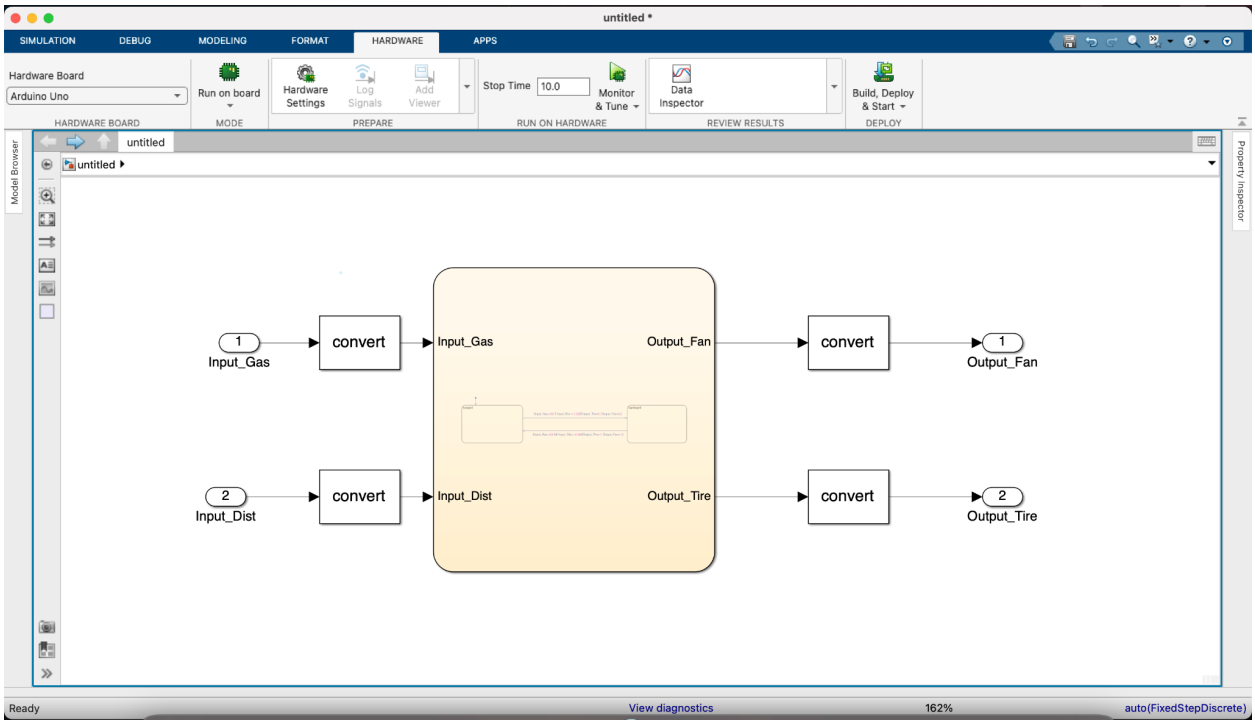






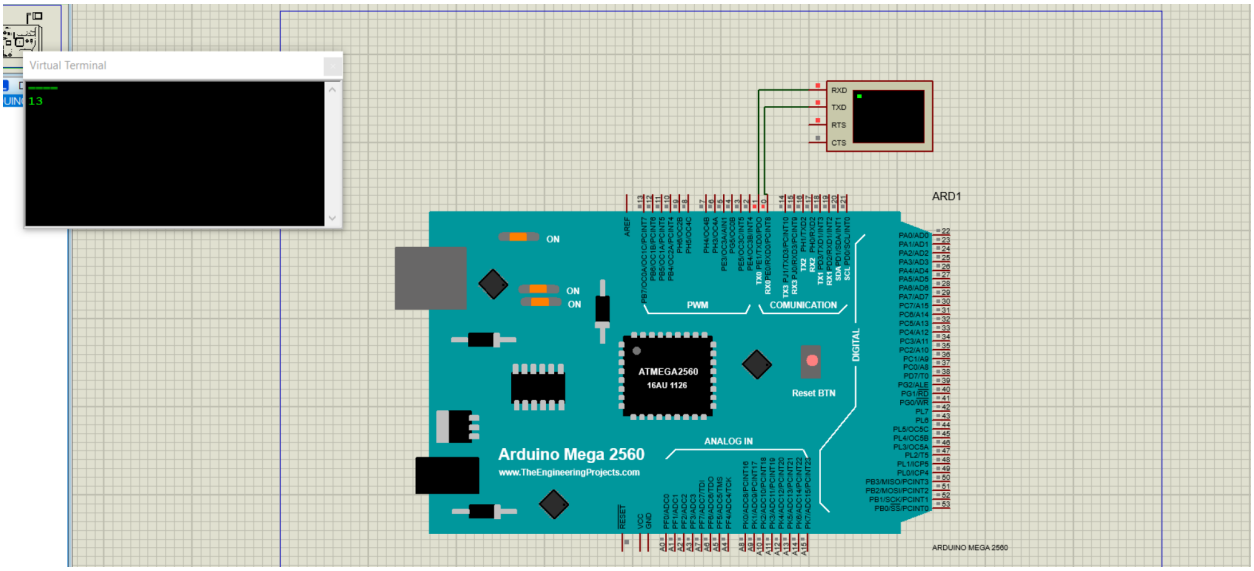
تمام مراحل خواسته شده در موارد بالا انجام شده است و در انتها نوع های جدید در مدل جاگذاری شده است.

تصویر زیر نشان میدهد که از Convert ها استفاده شده است در انتها.



ج. تولید کد را مجددا برای مدل ممیز ثابت انجام داده و مطابق بخش أ آن را برای برد کامپایل و شبیه‌سازی کنید. زمان اجرا را در این حالت نیز برای سطوح مختلف بهینه‌سازی کامپایلر جداگانه اندازه‌گیری کنید. زمان‌های اجرای اندازه‌گیری شده را در قالب جداولی در گزارش خود تنظیم کنید.

برای این قسمت لازم است تا صرفا تمام مراحل قسمت الف را دوباره تکرار بکنیم و مجددا برای خلاصه بودن گزارش در تمام ۶ حالت کامپایل شده خروجی همانند زیر داریم.



نتیجه گیری به این صورت است که بعد از شبیه سازی برای بار دوم در بهینه سازی های کامپایلر تاثیری نداشته است. و صرفا زمان اجرا ۱ واحد افزایش داشته است (همانطور که در شکل بالا مشخص است)