



## درس طراحی در سطح سیستم

### تکلیف کامپیوتری ۲: طراحی و سنتز سطح بالا در MATLAB

دانشکده فنی دانشگاه تهران

دانشکده مهندسی برق و کامپیوتر

دکتر بیژن علیزاده

نیم سال دوم سال تحصیلی ۱۴۰۳-۱۴۰۴

نگارش: محمدمعین جنیدی جعفری (@moeinjo, rouzbehjoneidi@gmail.com)

#### مقدمه:

در این تمرین کامپیوتری با ابزارهای مختلف MATLAB برای سنتز در سطح بالا آشنا می‌شویم که ادامه توضیحات مربوط به هر بخش آمده است:

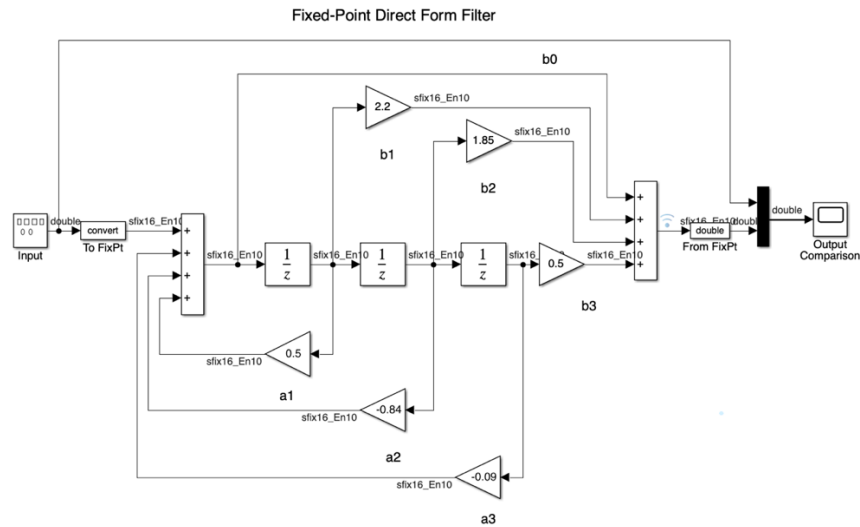
- **Fixed-Point Tool:** این ابزار در محیط MATLAB به شما کمک می‌کند تا مدل‌های خود را برای پردازش با دقت‌های ثابت برنامه‌ریزی کنید. با استفاده از این ابزار، می‌توانید مدل‌های خود را از ممیز شناور به ممیز ثابت تبدیل کرده و تنظیمات مختلف را برای دقت اعمال کنید.

- **HDL Coder:** این ابزار به شما این امکان را می‌دهد که الگوریتم‌ها و مدل‌های MATLAB و Simulink خود را به زبان‌های سخت‌افزاری مانند Verilog و VHDL تبدیل کنید. با استفاده از HDL Coder، می‌توانید سیستم‌های سخت‌افزاری خود را مدل‌سازی کرده و سپس آن‌ها را سنتز و شبیه‌سازی کنید.

توجه: مواردی که باید در گزارش قید شوند با شماره و رنگ **قرمز** در صورت پروژه نوشته شده‌اند.

#### ۱- استفاده از ابزار Fixed Point

در این بخش نحوه استفاده از ابزار fixed point را برای مقایسه دقت میان انواع داده‌های floating point و fixed point در مدل شکل ۱ فرا می‌گیریم. شکل ۱ نشان‌دهنده یک فیلتر دیجیتال هست که از المان‌هایی مانند sum, gain, delay تشکیل شده است.



شکل ۱- فیلتر دیجیتال

دو مورد از مزایا و دو مورد از معایب تبدیل floating point به fixed point را بیان کنید.

### ۱-۱- بررسی فیلتر دیجیتال

در این بخش به بررسی عملکرد مداری که قرار است در آن تبدیل floating point به fixed point را انجام دهیم، می‌پردازیم. برای این کار گام‌های زیر را انجام دهید:

- فایل "fxpdemo\_direct\_form2.slx" که در اختیار شما قرار گرفته را باز کنید.

شبیه‌سازی را با زدن گزینه Run انجام دهید و تصویر output comparison را در گزارش بیاورید. شکل موج خروجی را با توجه به شکل موج ورودی و مدار داده شده توجیه کنید.

- کد زیر را که تولیدکننده ۶ ورودی با دامنه‌های مختلف برای فیلتر دیجیتال است را در command window اجرا کنید:

```
simIn(1:6) = Simulink.SimulationInput('fxpdemo_direct_form2');  
  
simIn(1) = simIn(1).setBlockParameter('fxpdemo_direct_form2/Input',...  
    'Amplitude','0.001');  
simIn(2) = simIn(2).setBlockParameter('fxpdemo_direct_form2/Input',...  
    'Amplitude','0.01');  
simIn(3) = simIn(3).setBlockParameter('fxpdemo_direct_form2/Input',...
```

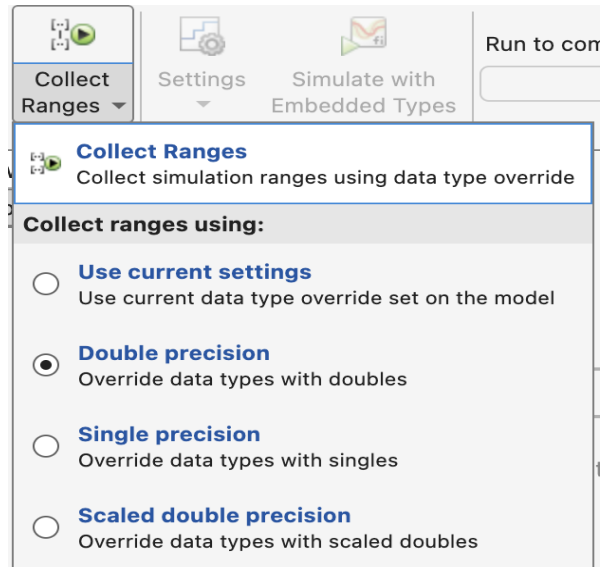


```
'Amplitude','0.1');  
simIn(4) = simIn(4).setBlockParameter('fxpdemo_direct_form2/Input',...  
'Amplitude','1');  
simIn(5) = simIn(5).setBlockParameter('fxpdemo_direct_form2/Input',...  
'Amplitude','10');  
simIn(6) = simIn(6).setBlockParameter('fxpdemo_direct_form2/Input',...  
'Amplitude','100');  
Simulink.sdi.markSignalForStreaming('fxpdemo_direct_form2/Sum1',1,'on');
```

## ۲-۱- تغییر مبنای فیلتر دیجیتال به Fixed Point و مقایسه با مبنای Floating Point

حال با استفاده از fixed point tool مبنای فیلتر دیجیتال را به fixed point تغییر می‌دهیم و مقایسه‌ای میان این دو مبنای fixed point و floating point انجام می‌دهیم. برای این کار مراحل زیر را طی کنید:

- بر روی صفحه کلیک راست کرده و گزینه fixed point tool را انتخاب کنید.
- از میان workflowهای موجود Range Collection را انتخاب کنید.
- در بخش simulation input گزینه simIn را انتخاب کنید تا ورودی‌ها با دامنه‌های مختلف که در بخش ۱-۱ به مدار دادیم به‌عنوان ورودی‌های شبیه‌سازی قرار گیرند.
- در بخش Signal Tolerances مقدار ۰.۱۵ را در فیلد Relative Tolerance وارد نمایید. این مقدار نشان‌دهنده حداکثر خطای نسبی یک سیگنال است تا قابل قبول باشد.
- همانند شکل ۲ تنظیمات collect ranges را روی double precision قرار دهید. زمانی که شما این کار را انجام می‌دهید، fixed point tool نوع داده‌های درون مدل شما را به double تغییر داده، مدل را شبیه‌سازی کرده و بیشینه و کمینه مقادیر هر یک از المان‌های مدل را محاسبه می‌کند.

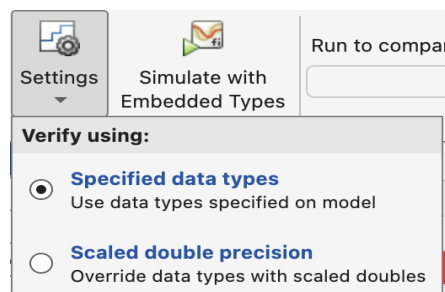


شکل ۲- تنظیمات collect ranges

- سپس collect ranges را اجرا کنید. با این کار شبیه‌سازی با روش floating point انجام می‌شود.
- در workflow browser وضعیت underflow و overflow به‌ازای ۶ ورودی مختلف نمایش داده شده است.

وضعیت underflow و overflow را به‌ازای هر ۶ ورودی گزارش و تحلیل کنید و بگویید چه رابطه‌ای میان رخ دادن underflow و overflow با دامنه سیگنال ورودی وجود دارد؟

در بخش setting گزینه specified data types را انتخاب کنید (مطابق شکل ۳).



شکل ۳- تنظیمات collect ranges

- بر روی گزینه simulate with embedded types کلیک کنید. با این کار شبیه‌سازی با روش fixed point انجام می‌شود و شما می‌توانید نتایج بدست آمده از دو روش را مقایسه کنید.



- همان طور که دیده می شود embedded run تنها برای یکی از ورودی ها به ازای tolerance relative قرار دادیم توانست خروجی مطلوب تولید کند.
- بر روی خروجی مطلوب کلیک راست کنید و open SDI را انتخاب کنید

تصویر صفحه نمایش داده شده را در گزارش بیاورید و نمودارهای نشان داده شده را تحلیل کنید.

مرحله ۴ را برای EmbeddeRun\_senarion\_1 و EmbeddeRun\_senarion\_6 تکرار کنید.

## ۲- استفاده از ابزار HDL coder

در این قسمت یاد خواهید گرفت که یک فیلتر FIR را در بخش سیمولینک متلب طراحی و کد Verilog یا VHDL آن را تولید کنید. توجه شود که در درس FPGA یاد خواهید گرفت که این کد را بر روی FPGA پیاده سازی کرده و از فیلتر طراحی شده استفاده کنید.

ابتدا شبیه سازی را اجرا کنید و از تصاویر خروجی در گزارش بیاورید و بگویید که هر یک از سیگنال ها چه هستند؟

در درس با فیلترهای FIR آشنا شده اید و می دانید که این فیلترها از ضرب متوالی یک سری ضرایب در سیگنال ورودی، خروجی را تولید می کنند.

با توجه به workspace در نرم افزار، بگویید که این فیلتر طراحی شده چند ضریب دارد؟

در بخش شبیه سازی همان طور که دیده می شود، از یک واحد تأخیر استفاده شده است. با توجه به ساختار فیلترهای FIR، بگویید چرا این تأخیر وجود دارد؟ (راهنمایی: می توانید تأخیر را حذف کنید و شبیه سازی را اجرا کرده و با حالت قبل مقایسه کنید).

در Simulink از تب های بالا، تب modeling را انتخاب کرده، سپس از گزینه های model setting، گزینه model properties را انتخاب کنید. وارد بخش callbacks شوید و سپس بر روی گزینه InitFcn کلیک کنید.

این بخش، پارامترهای فیلتر شما را نشان می دهد که قرار است سنتز شود. شما می توانید مقادیر آن را تغییر داده و فیلتر مورد نظر خود را طراحی کنید.



ذکر کنید اگر فیلتر بخواهیم که بتواند محدوده صدای انسان را عبور دهد (فرکانس ۸۰ تا ۸۰۰۰)، باید این پارامترها به چه مقادیر تنظیم شوند؟ (فرکانس نمونه‌برداری ۴۴۱۰۰)

بدون تغییر مقادیر، model properties را ببندید.

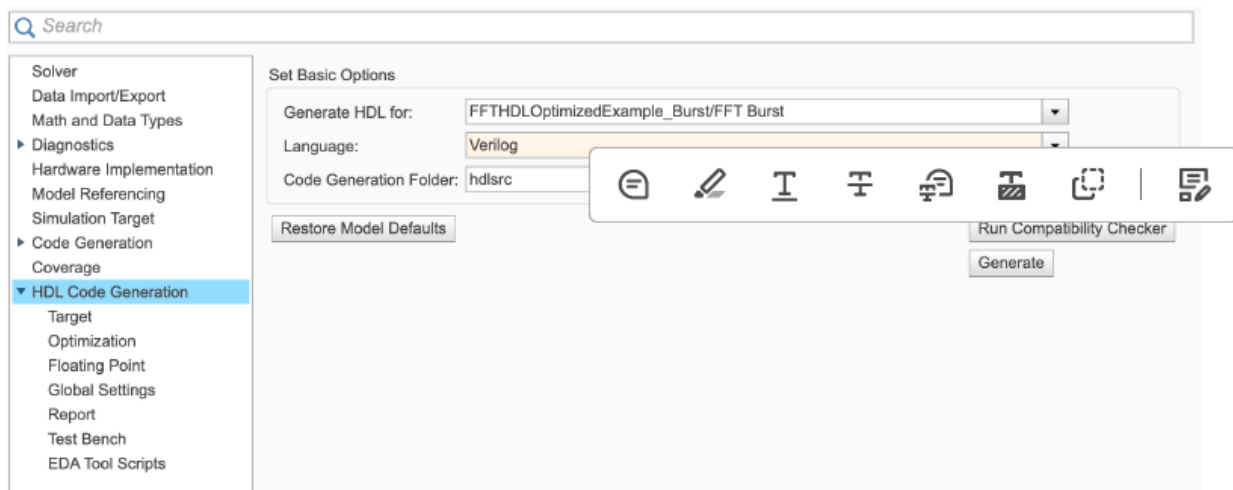
حال می‌خواهیم کد Verilog فیلتر مورد نظر را تولید کنیم.

به‌منظور تولید کد HDL مراحل زیر را طی کنید:

۱- وارد model setting بشوید.

۲- در بخش solver، type را fixed point و solver را بر روی discrete قرار دهید.

۳- بر روی بخش HDL Code Generation کلیک کنید و زبان سخت افزار را Verilog انتخاب کنید. (مطابق شکل ۴)



شکل ۴- نمایشی از محیط model properties بخش HDL Code Generation

۴- در بخش HDL code generation گزینه target را بزنید و فرکانس را 50MHz انتخاب کنید.

۵- در بخش optimization قسمت‌هایی برای pipeline و همچنین resource sharing وجود دارد که فعلاً با آن‌ها کاری نداریم.

۶- در بخش global setting نیز تنظیماتی از قبیل سنکرون بودن یا نبودن reset و یک‌سری تنظیمات دیگر که ممکن است برایتان آشنا باشد وجود دارد که با آن‌ها نیز کاری نداریم.

۷- در بخش report تمامی رپورت‌ها را فعال کنید تا در انتها به شما گزارش سنتز داده شود و از آن‌ها برای تحلیل استفاده کنید.

۸- در بخش testbench گزینه HDL testbench را فعال کنید تا تست بنچ نیز برای شما تولید شود. همچنین مقادیری نظیر setup time و hold time را نیز می‌توانید تنظیم کنید که با این مقادیر در درس FPGA آشنا خواهید شد. در این تمرین مقادیر را تغییر ندهید.

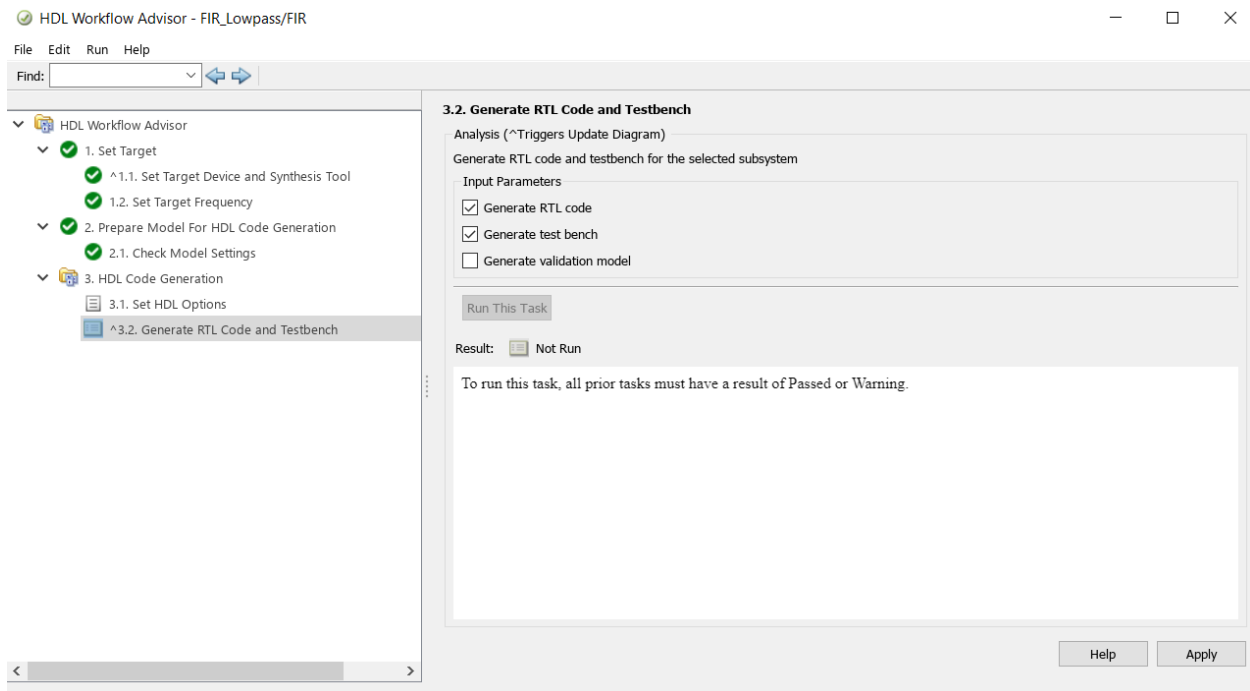


۹- تنظیمات را ذخیره و سپس از پنجره خارج شوید.

۱۰- بر روی ماژول Programmable FIR via Registers کلیک راست کنید و در بخش hdl code ، گزینه‌ی HDL Workflow Advisor را بزنید.

۱۱- در بخش Set Target Device and Synthesis Tool، Target را generic Asic/FPGA بگذارید.

۱۲- سپس مراحل را به نوبت run کرده. در مرحله سه حواستان باشد که گزینه generate testbench نیز فعال باشد. (تمام مراحل در صورت موفقیت تیک سبز خواهند داشت.) (مطابق شکل ۵)



شکل ۵- نمایی از HDL Workflow Advisor و انتخاب گزینه Generate test bench

۱۳- در این مرحله باید رپورت تولید شده را در گزارش بیاورید و همچنین تصویری از موفق بودن تمام مراحل در گزارش قرار دهید.

همچنین تصاویر رپورت تولید شده در متلب را نیز در گزارش خود قرار دهید و بر اساس آن Critical-Path، مقدار منابع استفاده شده و اطلاعات پورت‌های ورودی و خروجی را گزارش کنید.

۱۴- کدهای ورایلاگ تولید شده را در modelsim کامپایل کرده، سپس از تست بنچ داده شده استفاده کرده و نتیجه را گزارش کنید.



### بارمبندی سؤالات

- بخش ۱ : ۳۵ نمره

- بخش ۲ : ۶۵ نمره

- نام فایل تحویلی به صورت CA2\_Name\_#StudentNum.zip باشد.
- فایل آپلود شده باید شامل گزارش و فایل‌های تولید شده پس از انجام مراحل بالا باشد.
- فایل‌های هر بخش را در یک پوشه جداگانه قرار دهید.
- تمیزی گزارش و مرتب بودن فایل‌های آپلود شده نمره امتیازی دارد.

موفق و سلامت باشید.