به نام خدا

دانشگاه تهران

دانشکده مهندسی برق و کامپیوتر

ESL

گزارش پروژه دوم

نام و نام خانوادگی	امیرحسام جعفری راد
شماره دانشجویی	۸۱۰۱۰۰۲۴۷
تاریخ ارسال گزارش	

فهرست

بخش اول) استفاده از ابزار Fixed Point

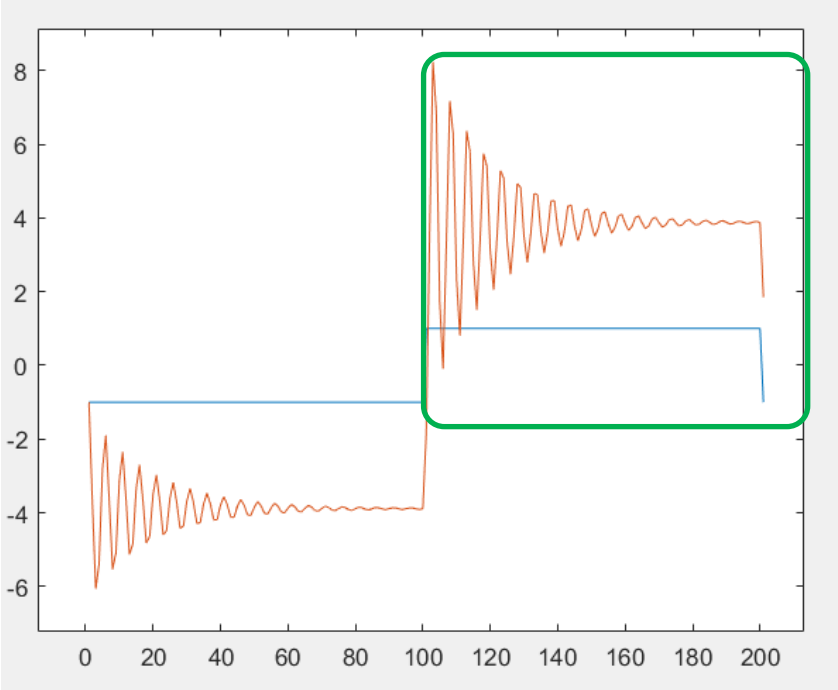
۱- دو مورد از مزایا و معایب تبدیل floating point به fixed point را بیان کنید:

در حالت fixed point سیستم دقت و کنترل پذیری بیشتری نسبت به حالت ممیز شناور دارد. همچنین در پیاده سازی سخت افزاری این سیستم به المان های کمتر و ساده تری نیاز است که باعث کاهش هزینه ساخت و افزایش کارایی و سرعت نیز می‌شود.

در سیستم ممیز ثابت امکان رخ دادن underflow یا overflow خیلی بیشتر از سیستم ممیز شناور است چرا که مدیریت این دو موضوع باید هنگام طراحی به درستی انجام شود. همچنین در تعداد بیت های یکسان برای هردو سیستم، سیستم ممیز ثابت دامنه و رنج محدود تری نسبت به سیستم ممیز شناور دارد.

۱-۱) بررسی فیلتر دیجیتال

۲- نتیجه شبیه‌سازی و نمودار output comparison:




```
Command Window
>> 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');
>> plot(OutputComparison.signals.values,'DisplayName','OutputComparison.signals.values')
fx >>
```

```
Workspace
Name ^   Value
OutputComp...   1x1 struct
sigsOut   1x1 Dataset
simIn   1x6 SimulationIn...
tout   201x1 double
```

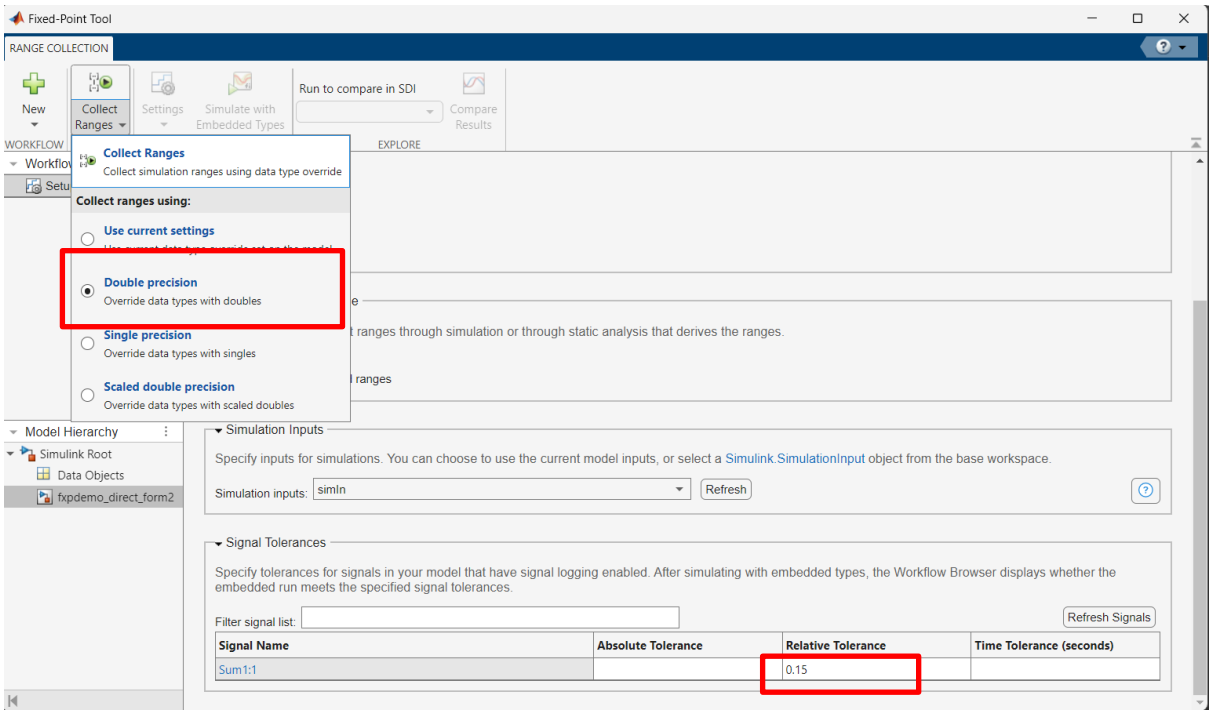
نمودار خروجی برحسب ورودی پس از اعمال کد متلب مربوطه به شرح بالا می‌باشد.

در این نمودار رابطه ورودی ها به فرم $x(t) = 0.5x(t-1) - 0.84x(t-2) - 0.09x(t-3) + u(t)$ و رابطه خروجی به فرم $y(t) = x(t) + 2.2x(t-1) + 1.85x(t-2) + 0.5x(t-3)$ می‌باشد.

خروجی این معادله تقریبا برابر با بخش سبز رنگ در تصویر می‌باشد که با نمودار اصلی برابر و بر آن منطبق است.

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

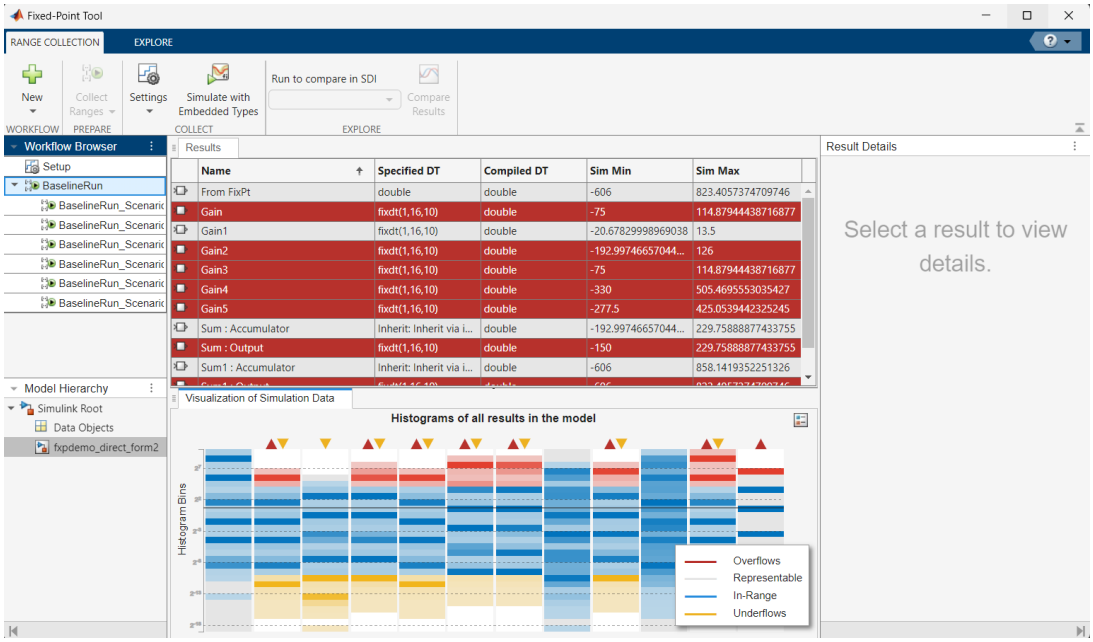
تنظیمات گفته شده مطابق تصویر اعمال شده است.



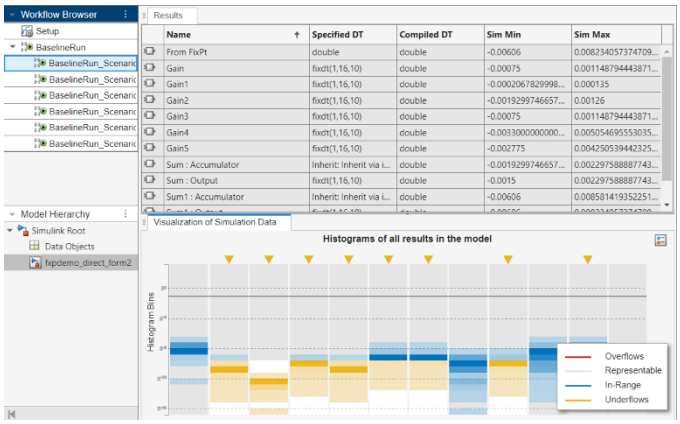
۳- وضعیت underflow و overflow به ازای هر ۶ ورودی و رابطه آن‌ها با سیگنال ورودی:

مشاهده می‌شود که در ابتدا میزان زیادی underflow رخ داده است که مطابق انتظار است. چرا که در ابتدا ورودی ها کوچک بوده و به ازای این ورودی ها سیستم دچار underflow شده است. انتظار میرود با افزایش دامنه ورودی میزان underflow کاهش پیدا کند که در نمودار نیز قابل مشاهده است و در مرحله ای که ورودی ها به میزان متوسطی رسیده اند دیگر اثری از underflow دیده نمیشود و مقدار آن برابر با صفر خواهد شد. همچنین در صورت افزایش بیش از حد ورودی انتظار میرود سیستم دچار overflow شود که در تصاویر نیز مشاهده می‌شود.

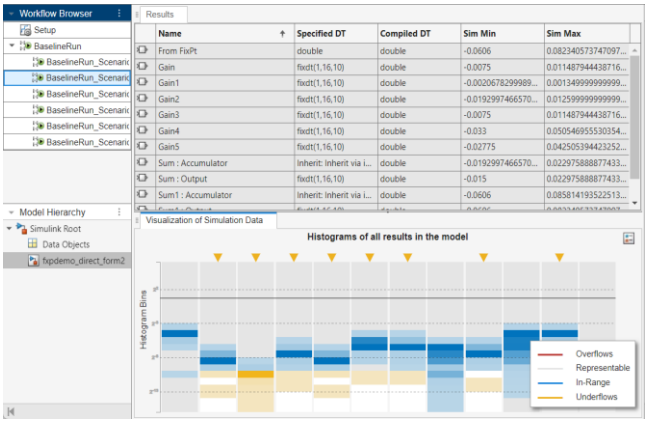
نمودار مربوطه:



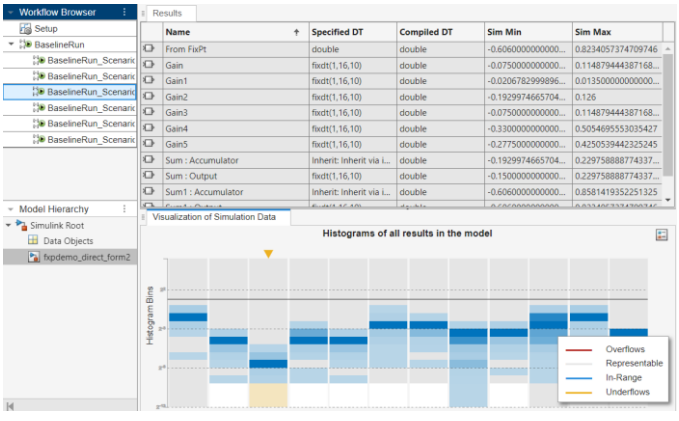
نمودار های سناریو های ۱ الی ۶ به تفکیک نیز به شرح ذیل می‌باشد:



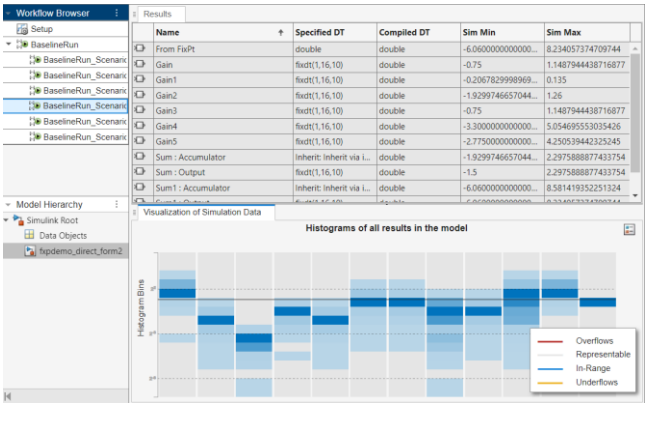
حالت اول



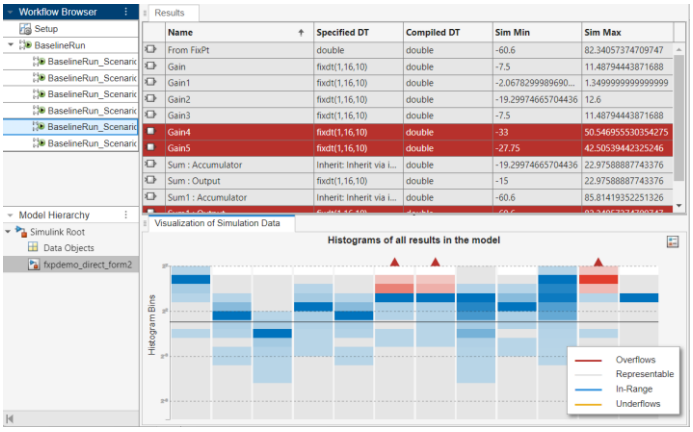
حالت دوم



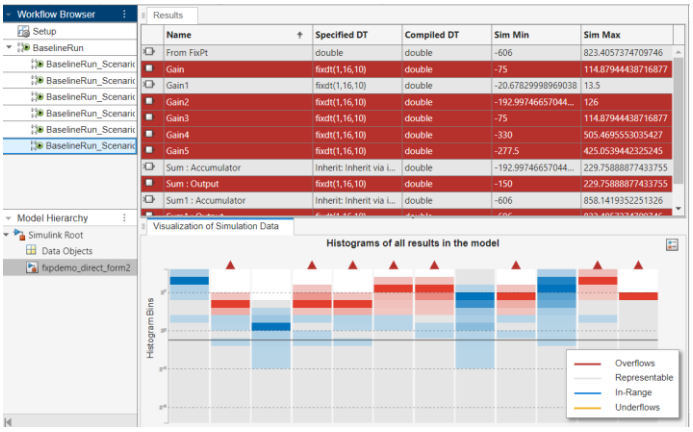
حالت سوم



حالت چهارم



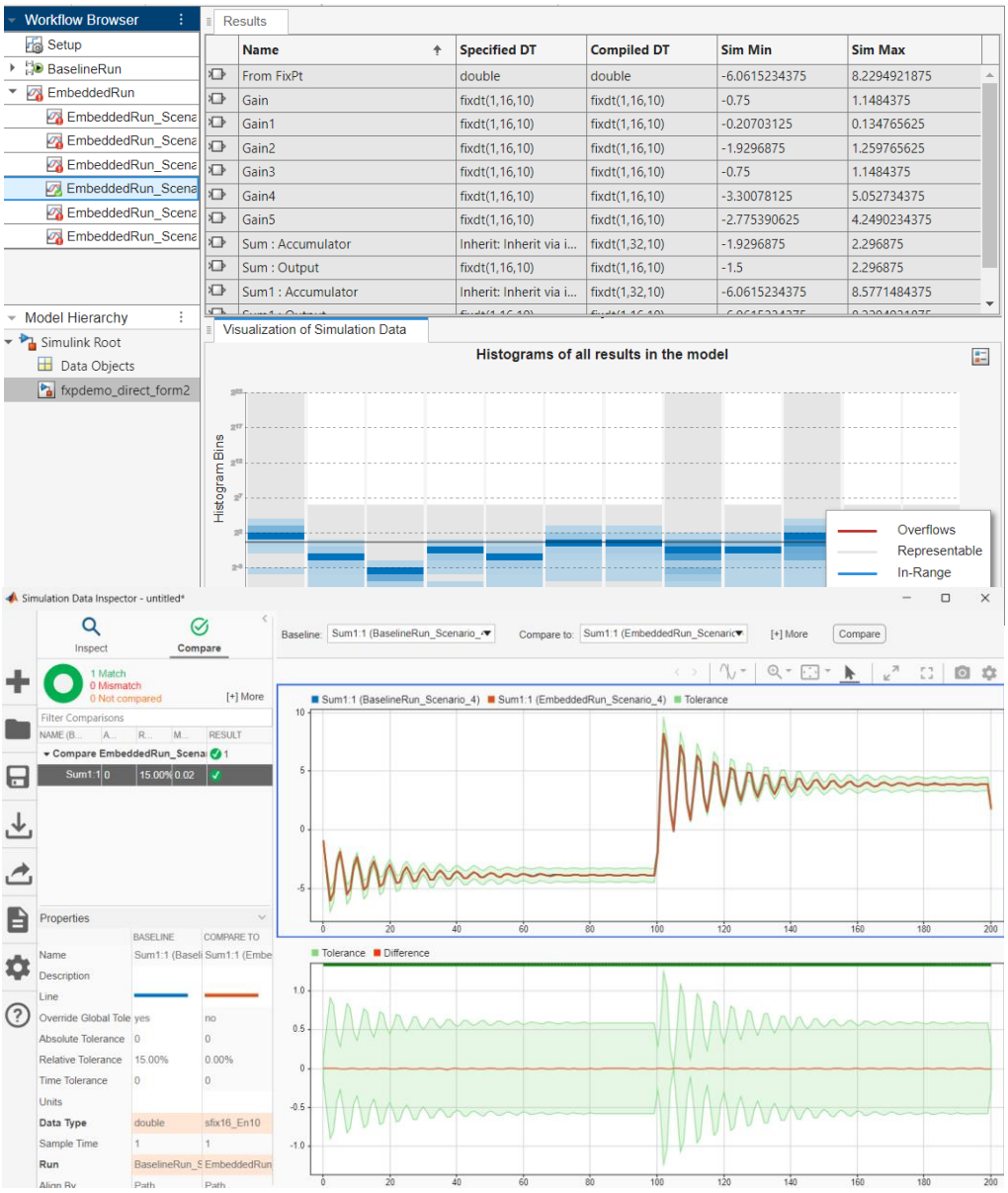
حالت پنجم



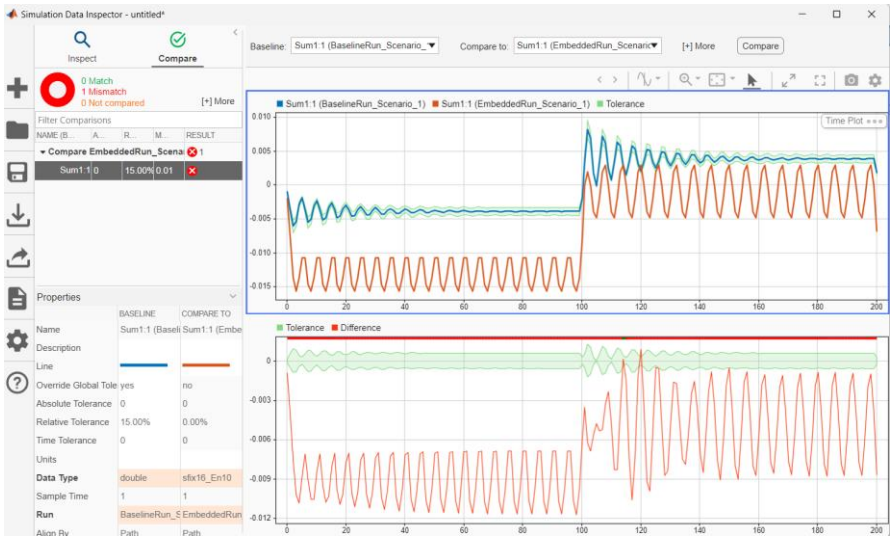
حالت ششم

۴- تصویر صفحه نمایش داده شده:

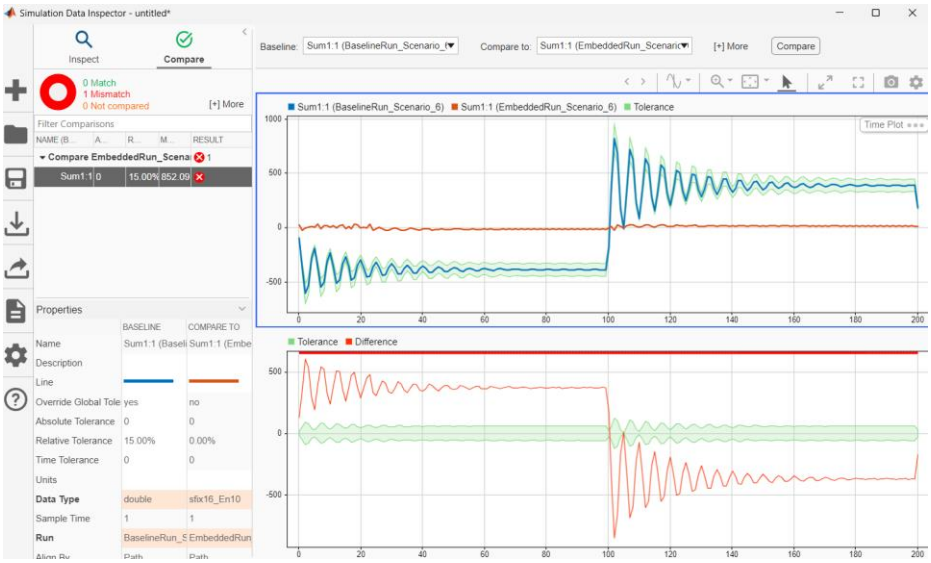
همانطور که مشاهده می‌شود تنها سناریو شماره ۴ پاسخ مطلوب را تولید کرده که به شرح ذیل می‌باشد:



۵- مرحله ۴ برای سناریو اول و ششم:



سناریو اول: در این حالت خروجی دچار underflow شده است و به همین دلیل مطابق تصویر خروجی در قسمت مطلوب و قابل نمایش قرار نگرفته است.

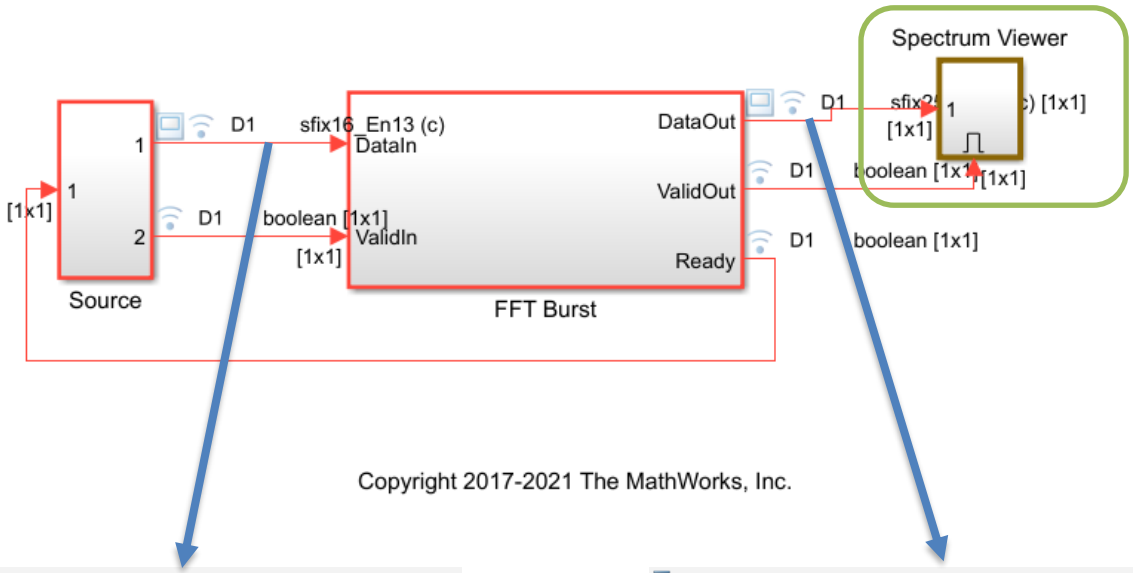


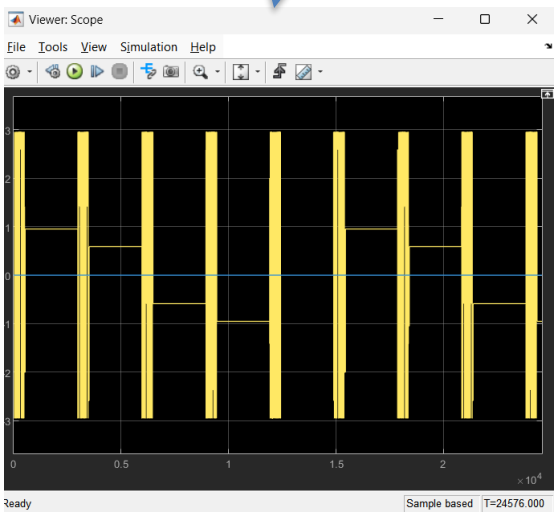
سناریو ششم: در این حالت سیستم دچار overflow شده است و به همین علت مطابق تصویر خروجی خروجی در بخش مطلوب و قابل نمایش قرار نگرفته است.

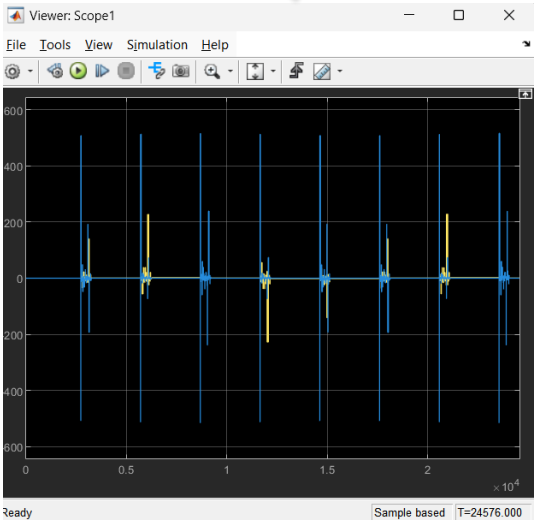
بخش دوم) استفاده از ابزار HDL coder

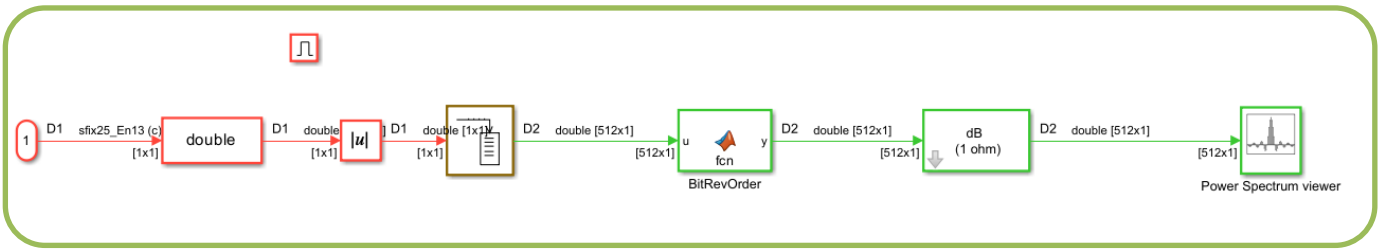
۱-۵) بررسی مدل FFT

۱-تصویر scope خروجی و ورودی و همچنین spectrum viewer





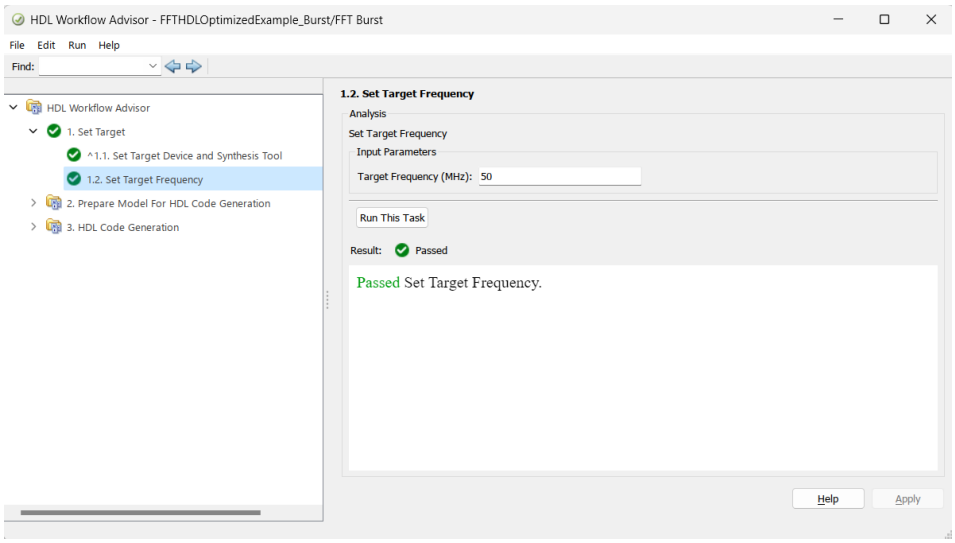




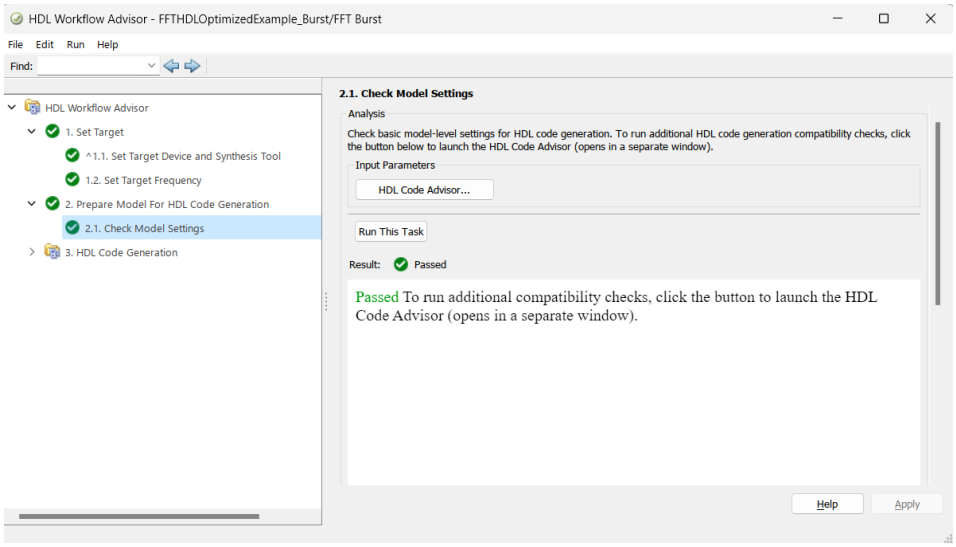
۲-۲) سنتز مدل و تولید کد Verilog مدل FFT

۲- نتیجه Run و موفقیت آمیز بودن مرحله:

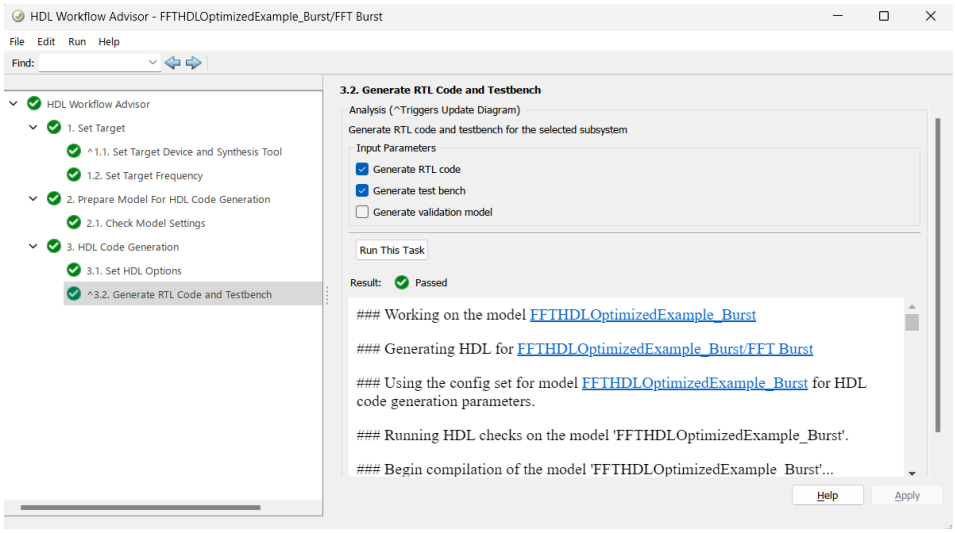
اول:



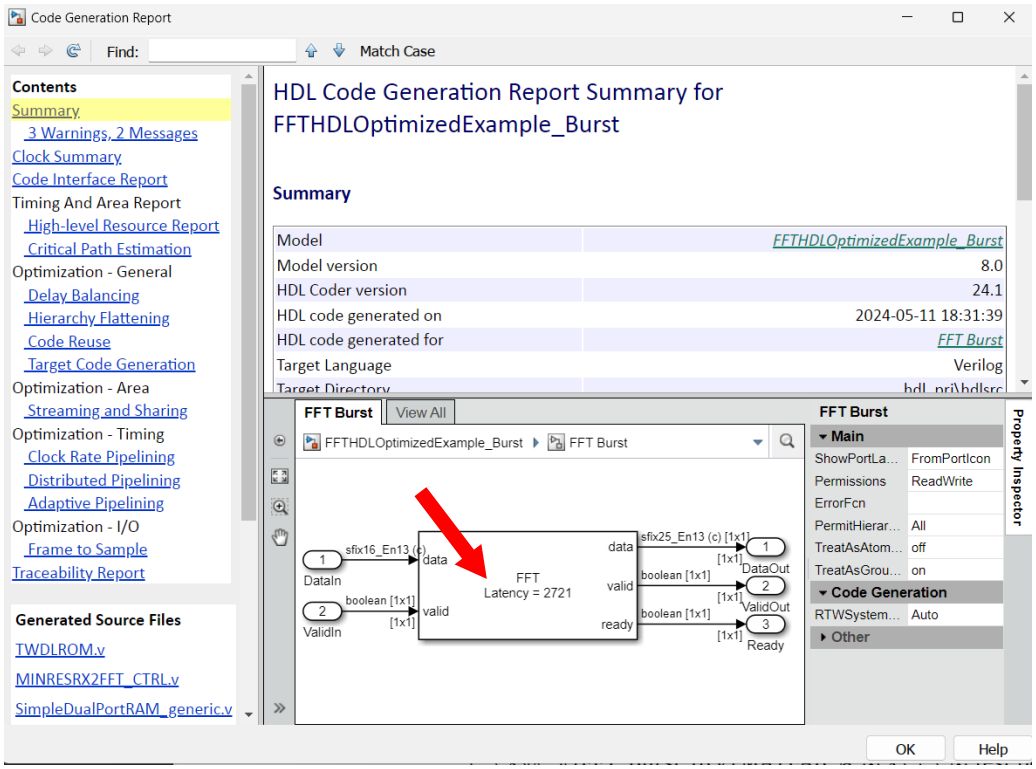
دوم:



سوم:



۳- تصویر Latency ماژول:



Latency به معنای مدت زمانی که این ماژول شروع به کار میکند تا لحظه ای که خروجی میدهد و کارش تمام می‌شود می‌باشد.

۴- تصویر critical path estimation و مفهوم آن:

Critical Path Details

Id	Propagation (ns)	Delay (ns)	Block Path
1	1.6950	1.6950	FFT

critical path estimation به معنای طولانی ترین و بیشتری تاخیری که ماژول در مدار خود دارد تا ورودی به خروجی برسد می‌باشد.

۵- تصویر تمام گزارش‌های مربوط به high-level resource report

Generic Resource Report for FFTHDLOptimizedExample_Burst

Summary

Multipliers	4
Adders/Subtractors	45
Registers	171
Total 1-Bit Registers	3074
RAMs	6
Multiplexers	86
I/O Bits	87
Static Shift operators	5
Dynamic Shift operators	0

Detailed Report

Report for Subsystem: *FFT Burst*

Multipliers (4)

```
25x25-bit Multiply : 4
```

Adders/Subtractors (45)

```
8x8-bit Adder : 14
51x51-bit Adder : 1
52x52-bit Adder : 2
9x9-bit Adder : 10
4x4-bit Adder : 1
2x2-bit Adder : 1
3x3-bit Adder : 1
20x20-bit Subtractor : 3
11x11-bit Subtractor : 2
8x8-bit Subtractor : 1
26-bit Subtractor : 6
51x51-bit Subtractor : 1
52x52-bit Subtractor : 2
```

Registers (171)

```
1-bit Register : 52
8-bit Register : 19
3-bit Register : 3
9-bit Register : 3
6-bit Register : 1
25-bit Register : 68
50-bit Register : 12
51-bit Register : 2
52-bit Register : 8
4-bit Register : 2
2-bit Register : 1
```

RAMs (6)

```
256x25-bit RAM : 6
```

Multiplexers (86)

```
3-bit 6-to-1 Multiplexer : 2
1-bit 6-to-1 Multiplexer : 2
20-bit 2-to-1 Multiplexer : 6
6-bit 6-to-1 Multiplexer : 1
11-bit 2-to-1 Multiplexer : 4
8-bit 3-to-1 Multiplexer : 6
8-bit 5-to-1 Multiplexer : 1
25-bit 10-to-1 Multiplexer : 2
26-bit 2-to-1 Multiplexer : 12
8-bit 2-to-1 Multiplexer : 5
25-bit 3-to-1 Multiplexer : 8
8-bit 10-to-1 Multiplexer : 2
8-bit 4-to-1 Multiplexer : 3
1-bit 3-to-1 Multiplexer : 3
25-bit 2-to-1 Multiplexer : 4
4-bit 25-to-1 Multiplexer : 1
1-bit 7-to-1 Multiplexer : 1
4-bit 7-to-1 Multiplexer : 1
9-bit 13-to-1 Multiplexer : 1
1-bit 22-to-1 Multiplexer : 1
1-bit 19-to-1 Multiplexer : 2
1-bit 20-to-1 Multiplexer : 1
1-bit 18-to-1 Multiplexer : 2
25-bit 14-to-1 Multiplexer : 2
25-bit 11-to-1 Multiplexer : 2
25-bit 5-to-1 Multiplexer : 2
1-bit 11-to-1 Multiplexer : 2
9-bit 6-to-1 Multiplexer : 1
2-bit 4-to-1 Multiplexer : 1
1-bit 10-to-1 Multiplexer : 1
1-bit 5-to-1 Multiplexer : 1
1-bit 8-to-1 Multiplexer : 1
1-bit 2-to-1 Multiplexer : 2
```

Number of Shift operators (5)

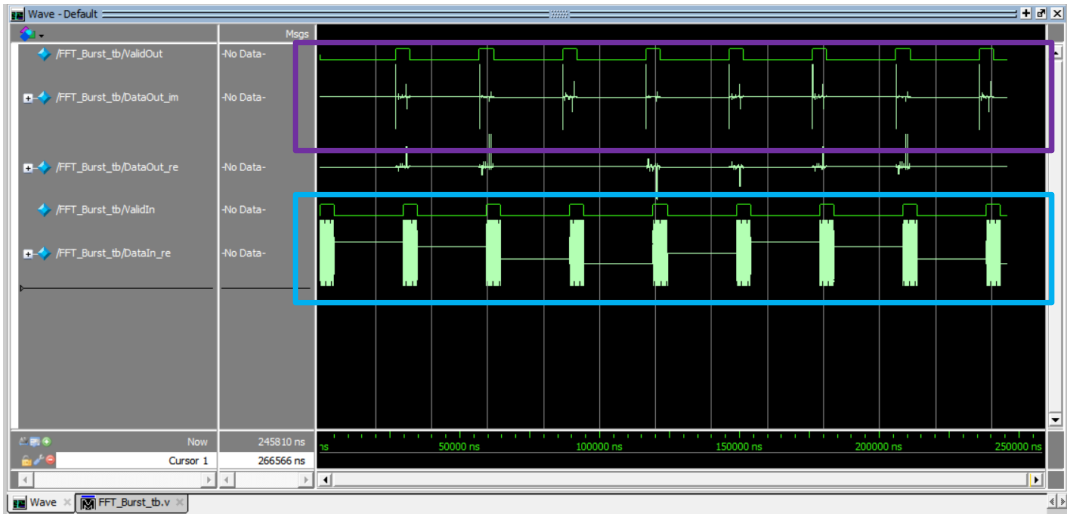
```
Static Shift Right : 5
```

Number of I/O Bits (87)

```
[+] Number of Input Bits: 35
[+] Number of Output Bits: 52
```

۲-۳) شبیه‌سازی کدهای تولید شده توسط HDL coder

۶- تصویر سیگنال‌های validOut, dataOut_im, dataOut_re, validIn, dataIn_re در شبیه‌سازی:

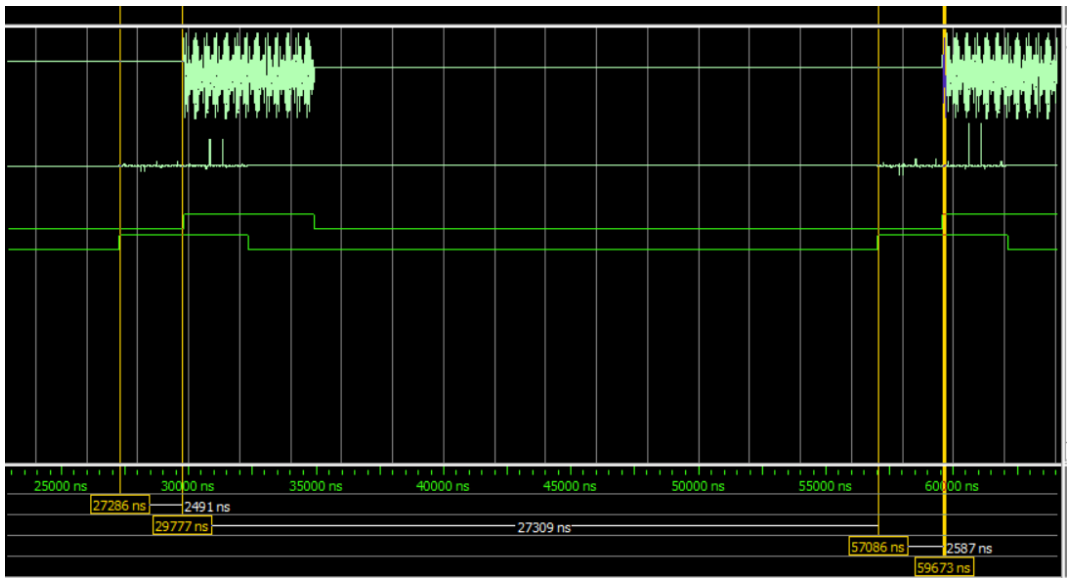


سیگنال DataIn_re بخش حقیقی سیگنال ورودی، سیگنال DataOut_re بخش حقیقی سیگنال خروجی و سیگنال DataOut_im بخش موهومی سیگنال خروجی را نمایش می‌دهد.

۷- مقایسه نتیجه شبیه‌سازی modelsim با سیگنال‌های خروجی Simulink:

همانطور که در تصویر بالا نشان داده شده است دقیقا همان سیگنال ورودی که در سیمولینک بود به رنگ آبی و همان سیگنال خروجی با رنگ بنفش مشخص شده اند. در واقع نتیجه سیمولینک و شبیه سازی مطابق انتظار یکسان است.

۸- استخراج مقدار Latency:

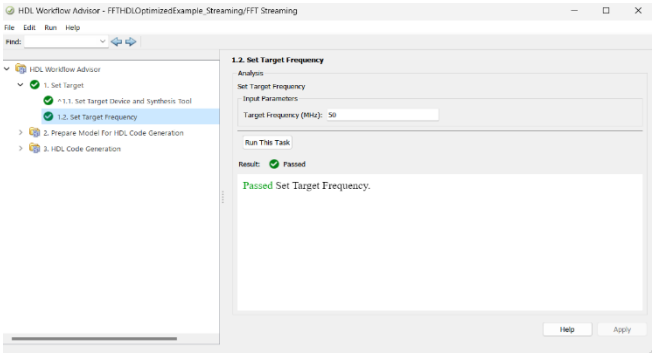


همانطور که در تصویر مشخص است میزان تاخیر یعنی فاصله زمانی میان فعال شدن سیگنال های validIn و validOut که در دوحالت این مقدار استخراج شده است. علت تفاوت جزئی موجود در این حالت نیز مربوط به تست بنچ مورد استفاده در شبیه سازی است.

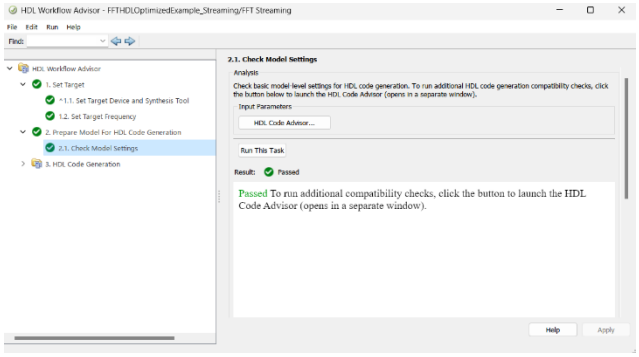
۹- نحوه کار واحد Butterfly Unit:

این واحد یکی از موثرترین واحد ها در پیاده سازی ماژول FFT میباشد و به این صورت کار میکند که ابتدا دو ورودی با تایپ اعداد گنگ دریافت کرده و بر اساس اپلیکیشن و نیاز عملیاتی را بر روی آن ها انجام می‌دهد. درواقع تفاوت اصلی این واحد با بقیه در نوع ورودی های قابل قبول آن است که اعداد گنگ هستند.

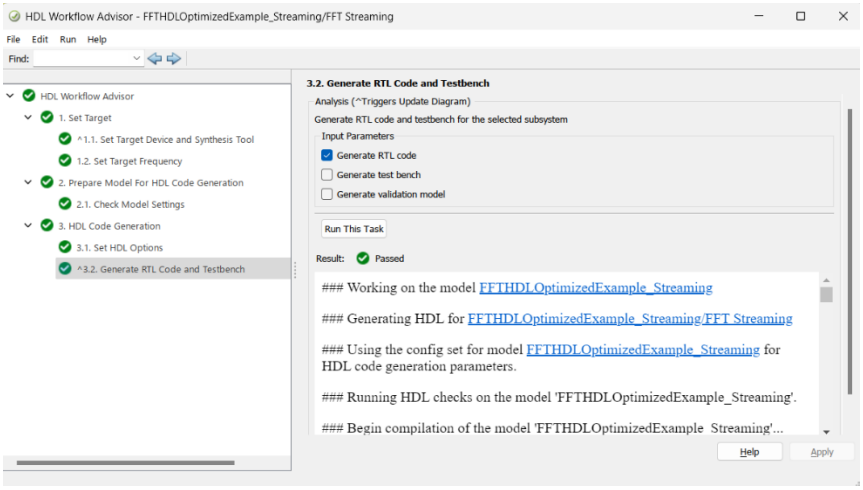
۱۰- مراحل ۲ تا ۵ برای فایل FFTHDLOptimizedExample_Streaming.slx:



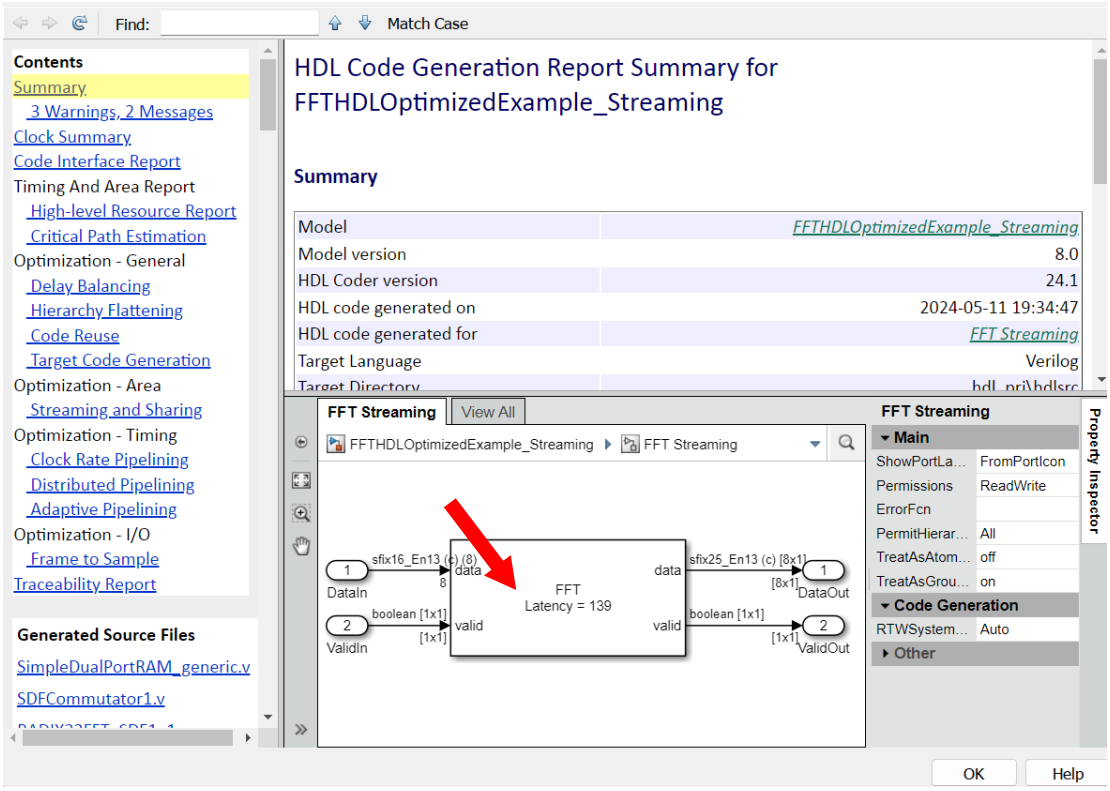
مرحله اول



مرحله دوم



مرحله سوم



Critical Path Details

Id	Propagation (ns)	Delay (ns)	Block Path
1	3.5100	3.5100	FFT

Generic Resource Report for FFTHDLOptimizedExample_Streaming

Summary

Multipliers	108
Adders/Subtractors	993
Registers	3637
Total 1-Bit Registers	52683
RAMs	160
Multiplexers	2512
I/O Bits	660
Static Shift operators	54
Dynamic Shift operators	0

Detailed Report

Report for Subsystem: *FFT Streaming*

Multipliers (108)

```
19x16-bit Multiply : 32
21x16-bit Multiply : 32
23x16-bit Multiply : 28
25x16-bit Multiply : 16
```

Adders/Subtractors (993)

```
5x5-bit Adder : 44
18x18-bit Adder : 16
3x3-bit Adder : 100
4x4-bit Adder : 44
19x19-bit Adder : 32
9x9-bit Adder : 27
2x2-bit Adder : 59
7x7-bit Adder : 16
36x36-bit Adder : 8
20x20-bit Adder : 16
21x21-bit Adder : 32
38x38-bit Adder : 8
22x22-bit Adder : 16
23x23-bit Adder : 32
40x40-bit Adder : 7
24x24-bit Adder : 8
25x25-bit Adder : 16
42x42-bit Adder : 4
26x26-bit Adder : 8
18x18-bit Subtractor : 16
19x19-bit Subtractor : 32
20x20-bit Subtractor : 97
11x11-bit Subtractor : 54
17-bit Subtractor : 162
36x36-bit Subtractor : 8
21x21-bit Subtractor : 32
38x38-bit Subtractor : 8
22x22-bit Subtractor : 16
23x23-bit Subtractor : 32
40x40-bit Subtractor : 7
24x24-bit Subtractor : 8
25x25-bit Subtractor : 16
42x42-bit Subtractor : 4
26x26-bit Subtractor : 8
```

Registers (3637)

```
16-bit Register : 210
1-bit Register : 1029
5-bit Register : 18
2-bit Register : 71
17-bit Register : 144
18-bit Register : 224
3-bit Register : 256
4-bit Register : 18
19-bit Register : 192
7-bit Register : 27
9-bit Register : 27
6-bit Register : 27
35-bit Register : 96
36-bit Register : 16
20-bit Register : 224
21-bit Register : 240
37-bit Register : 96
38-bit Register : 16
22-bit Register : 272
23-bit Register : 120
39-bit Register : 84
40-bit Register : 14
24-bit Register : 16
25-bit Register : 128
41-bit Register : 48
42-bit Register : 8
26-bit Register : 16
```

RAMs (160)

```
32x17-bit RAM : 16
8x17-bit RAM : 16
16x18-bit RAM : 16
8x18-bit RAM : 16
8x19-bit RAM : 32
4x20-bit RAM : 16
8x20-bit RAM : 16
8x21-bit RAM : 16
8x22-bit RAM : 16
```

Multiplexers (2512)

```
2-bit 8-to-1 Multiplexer : 18
1-bit 7-to-1 Multiplexer : 42
5-bit 5-to-1 Multiplexer : 1
1-bit 4-to-1 Multiplexer : 45
5-bit 6-to-1 Multiplexer : 1
2-bit 9-to-1 Multiplexer : 4
17-bit 8-to-1 Multiplexer : 16
3-bit 8-to-1 Multiplexer : 48
3-bit 7-to-1 Multiplexer : 40
1-bit 12-to-1 Multiplexer : 32
3-bit 15-to-1 Multiplexer : 16
5-bit 7-to-1 Multiplexer : 8
2-bit 7-to-1 Multiplexer : 19
17-bit 4-to-1 Multiplexer : 16
5-bit 10-to-1 Multiplexer : 8
17-bit 9-to-1 Multiplexer : 16
3-bit 14-to-1 Multiplexer : 16
17-bit 3-to-1 Multiplexer : 16
1-bit 3-to-1 Multiplexer : 49
4-bit 5-to-1 Multiplexer : 1
4-bit 6-to-1 Multiplexer : 1
19-bit 2-to-1 Multiplexer : 160
18-bit 8-to-1 Multiplexer : 16
4-bit 7-to-1 Multiplexer : 8
18-bit 4-to-1 Multiplexer : 16
4-bit 10-to-1 Multiplexer : 8
18-bit 9-to-1 Multiplexer : 16
18-bit 3-to-1 Multiplexer : 16
3-bit 6-to-1 Multiplexer : 68
1-bit 6-to-1 Multiplexer : 36
20-bit 2-to-1 Multiplexer : 162
6-bit 6-to-1 Multiplexer : 27
11-bit 2-to-1 Multiplexer : 108
9-bit 2-to-1 Multiplexer : 54
9-bit 4-to-1 Multiplexer : 27
2-bit 2-to-1 Multiplexer : 55
7-bit 2-to-1 Multiplexer : 16
16-bit 10-to-1 Multiplexer : 54
17-bit 2-to-1 Multiplexer : 324
3-bit 5-to-1 Multiplexer : 1
19-bit 6-to-1 Multiplexer : 16
1-bit 9-to-1 Multiplexer : 48
1-bit 5-to-1 Multiplexer : 43
3-bit 9-to-1 Multiplexer : 24
19-bit 4-to-1 Multiplexer : 16
19-bit 3-to-1 Multiplexer : 16
2-bit 5-to-1 Multiplexer : 2
2-bit 6-to-1 Multiplexer : 1
21-bit 2-to-1 Multiplexer : 208
20-bit 6-to-1 Multiplexer : 16
20-bit 4-to-1 Multiplexer : 16
20-bit 3-to-1 Multiplexer : 16
1-bit 8-to-1 Multiplexer : 9
21-bit 6-to-1 Multiplexer : 16
21-bit 4-to-1 Multiplexer : 16
21-bit 3-to-1 Multiplexer : 16
22-bit 2-to-1 Multiplexer : 48
23-bit 2-to-1 Multiplexer : 160
22-bit 4-to-1 Multiplexer : 32
3-bit 3-to-1 Multiplexer : 16
22-bit 3-to-1 Multiplexer : 16
24-bit 2-to-1 Multiplexer : 48
25-bit 2-to-1 Multiplexer : 64
25-bit 3-to-1 Multiplexer : 16
26-bit 2-to-1 Multiplexer : 48
```

Number of Shift operators (54)

```
Static Shift Left : 54
```

Number of I/O Bits (660)

```
[+] Number of Input Bits: 259
[+] Number of Output Bits: 401
```

۱۱- مقایسه و تحلیل حالت Burst و Streaming:

Latency: مطابق نتایج بدست آمده حالت Streaming دارای Latency بسیار کمتری نسبت به حالت Burst دارد و سریع تر انجام می‌شود و لذا نیازمند منابع بیشتری می‌باشد. درنتیجه برای زمان هایی که تاخیر latency برای ما اهمیت دارد و محدودیتی از نظر هزینه و منابع نداریم گزینه مناسب تری است.

Critical Path: مطابق نتایج بدست آمده حالت Streaming دارای مسیر بحرانی و تاخیر بیشتری نسبت به حالت Burst دارد.

۱۲- تفاوت میان ارتباط Burst و Streaming:

در حالت Streaming سیستم در هر لحظه متناسب با داده های ورودی عملیات مرتبط را انجام میدهد و به محض تغییری در ورودی، در همان حین عملیات جدید بر اساس ورودی جدید انجام میشود درحالیکه در حالت Burst ابتدا داده ها بصورت پکیج های کاملی بسته بندی شده و پس از آماده شدن تمامی ورودی ها سیستم بصورت همزمان شروع به کار می‌کند.