

Vivado Design Suite Tutorial

Design Analysis and Closure Techniques

UG938 (v2019.2) February 5, 2020





Revision History

The following table shows the revision history for this document.

Section	Revision Summary
02/05/2020 \	/ersion 2019.2
General Updates	 Updated design files removing redundant parts of the design not triggering suggestions. Added details on project integration of RQS files. Added section on creating ML strategies runs.
08/12/2019 \	/ersion 2019.1
General Updates	Validated for 2019.1.





Table of Contents

Revision History	2
Tutorial Overview	
Introduction	5
Tutorial Description	5
Software Requirements	
Locating Tutorial Design Files	
Lab 1: Setting Waivers with the Vivado IDE	6
Step 1: Starting the Vivado IDE	
Step 2: Generating the CDC Report	
Step 3: Waiving a Single CDC Violation	
Step 4: Generating a Report for Waived Violations	
Step 5: Generating a Text Report with Details for Waived Violations	13
Step 6: Waiving Multiple CDC Violations	
Step 7: Exporting Waivers	17
Step 8: Using the create_waiver Command	
Step 9: Waiving Multiple CDC Violations	19
Step 10: Waiving Multiple DRC Violations	21
Step 11: Generating a Summary Report for Waived Violations	26
Step 12: Using Waiver Commands	
Summary	
Lab 2: Using Report QoR Suggestions	
Introduction	
Step 1: Understanding the Design	
Step 2: Running Report QoR Suggestions	
Step 3: Understanding the Report	
Step 4: Run with Suggestions	
Step 5: Running ML Strategies	
Summary	



Appendix A: Additional Resources and Legal Notices	47
Please Read: Important Legal Notices	47





Tutorial Overview

Introduction

This tutorial uses the Vivado[®] design rules checker (report_drc), clock domain crossing checker (report_cdc), and quality of results enhancer (report_qor_suggestions) to analyze example designs for issues, and shows you how to take corrective actions.

Tutorial Description

Lab 1 walks you through creating waivers for CDC, methodology, and DRC violations.

Lab 2 is a guide to using the report_qor_suggestions (RQS) command.

Note: The designs used in this tutorial are intended to exhibit issues for demonstration purposes, and should not be used as a reference for designs outside this tutorial.

Software Requirements

This tutorial requires that the 2019.1 Vivado[®] Design Suite software release or later is installed.

For a complete list of system and software requirements, see the Vivado Design Suite User Guide: Release Notes, Installation, and Licensing (UG973).

Locating Tutorial Design Files

- 1. Download the reference design files from the Xilinx[®] website.
- 2. Extract the ZIP file contents into any write-accessible location.

This tutorial refers to the location of the extracted ZIP file contents as <Extract_Dir>.



Lab 1

Setting Waivers with the Vivado IDE

Introduction

In the Vivado[®] Design Suite, you can use the waiver mechanism to waive clock domain crossing (CDC), design rule check (DRC), or methodology check violations. After a violation is waived, it is no longer reported by the <code>report_cdc</code>, <code>report_drc</code>, or <code>report_methodology</code> commands. Waived checks are also filtered out from the mandatory DRCs run at the start of the implementation commands, such as <code>opt_design</code>, <code>place_design</code>, and <code>route_design</code>. For more information, see this link in the Vivado Design Suite User Guide: Design Analysis and Closure Techniques (UG906).

 \diamondsuit

 \bigcirc

IMPORTANT! The content of the waiver is built with the objects that exist when the waiver is created. However, if an instance referenced inside a waiver is replicated by Vivado[®], the replicated instance is automatically added to the waiver and saved in subsequent checkpoints and XDC.

This lab shows how to set waivers with the Vivado integrated design environment (IDE) using both menu commands and the Tcl Console. The lab focuses on CDC waivers, but the methods for waiving DRC and methodology violations are similar.

Step 1: Starting the Vivado IDE

This lab uses a Vivado design checkpoint (.dep file), which is a snapshot of a design. When you launch the Vivado IDE using a design checkpoint, a subset of the Vivado IDE functionality is available.

TIP: To launch the Vivado Tcl Shell on Windows, select Start \rightarrow All Programs \rightarrow Xilinx Design Tools \rightarrow Vivado <version > \rightarrow Vivado <version > Tcl Shell.

1. From the command line or the Vivado Tcl Shell, change to the directory where the lab materials are stored:

```
cd <Extract_Dir>/src/lab1
```

2. To start the Vivado IDE with the design checkpoint loaded, enter the following:

```
vivado my_ip_example_design_placed.dcp
```



 \bigcirc

TIP: You can disregard the critical warnings about the unbounded GT locations.

Step 2: Generating the CDC Report

In this step, you generate the CDC report to view the associated CDC violations.

- 1. Select **Reports** \rightarrow **Timing** \rightarrow **Report CDC**.
- 2. In the Report CDC dialog box, leave the default settings as-is, and click OK.

🝌 Report CDC 🛛 🗙
Report clock domain crossing (CDC) paths between clocks, even if set_false_path or set_clock_groups constraints have been applied.
Results name: cdc_1
Clocks
<u>F</u> rom:
<u>Ι</u> ο:
Report
Report from cells:
Waivers
Apply waivers
Report only waived paths
O Ignore all waivers
File Output
Export to file:
Overwrite Append
Options
Suspend message limits during command execution
Ignore command errors (quiet mode)
Command: report_cdc -name cdc_1
✓ Open in a new tab
Op <u>e</u> n in Timing Analysis layout
OK Cancel

The Summary (by clock pair) section of the CDC Report appears as follows.



2 I ≑ C	📢 🔍 🛛 Summa	ry (by clock pair)								
General Information	Severity ^1	Source Clock	Destination Clock	CDC Type	Exceptions	Endpoints	Safe	Unsafe	Unknown	No ASYNC_REG
Summary (by clock pair)	Oritical	my_ip_glblclk	my_ip_axi_aclk	No Common Primary Clock	False Path	2	1	1	0	C
Summary (by type)	Oritical	my_ip_axi_aclk	my_ip_drpclk	No Common Primary Clock	False Path	2	0	2	0	C
Summary (by waived endpoints)	Critical	my_ip_axi_aclk	my_ip_glblclk	No Common Primary Clock	False Path	942	12	351	579	185
CDC Details (928)	1 Info	my_ip_drpclk	my_ip_axi_aclk	No Common Primary Clock	False Path	1	1	0	0	C
my_ip_drpclk to my_ip_axi_aclk (1)	1 Info	input port clock	my_ip_drpclk	No Common Primary Clock	False Path	2	2	0	0	C
my_ip_glblclk to my_ip_axi_aclk (2)	Info	my_ip_glblclk	my_ip_drpclk	No Common Primary Clock	False Path	6	6	0	0	C
input port clock to my_ip_drpclk (2)	1 Info	my_ip_drpclk	my_ip_glblclk	No Common Primary Clock	False Path	2	2	0	0	C

The Summary (by CDC type) section appears as follows.

Q ≚ ≑ C		Q Summa	ry (by type)		
General Information	-	Severity ^1	ID	Count	Description
Summary (by clock pair)		\rm Oritical	CDC-1	536	1-bit unknown CDC circuitry
Summary (by type)		\rm Critical	CDC-4	4	Multi-bit unknown CDC circuitry
Summary (by waived endpoints)		\rm Critical	CDC-10	187	Combinational logic detected before a synchronizer
CDC Details (928)		\rm Critical	CDC-11	2	Fan-out from launch flop to destination clock
my_ip_drpclk to my_ip_axi_aclk (1)		Oritical	CDC-13	170	1-bit CDC path on a non-FD primitive
my_ip_glblclk to my_ip_axi_aclk (2)		\rm 6 Critical	CDC-14	5	Multi-bit CDC path on a non-FD primitive
input port clock to my_ip_drpclk (2)		👴 Warning	CDC-15	10	Clock enable controlled CDC structure detected
my_ip_axi_aclk to my_ip_drpclk (2)		1 Info	CDC-3	9	1-bit synchronized with ASYNC_REG property
my_ip_glblclk to my_ip_drpclk (6) my_ip_axi_aclk to my_ip_glblclk (913)		1 Info	CDC-9	5	Asynchronous reset synchronized with ASYNC_REG property

Step 3: Waiving a Single CDC Violation

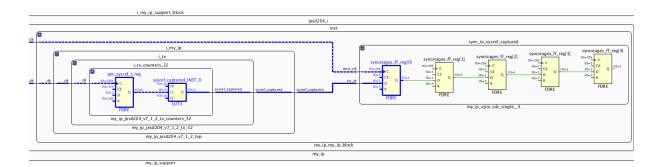
The $my_{ip_glblclk}$ to $my_{ip_axi_aclk}$ clock pair includes one Critical CDC-10 violation due to combinational logic on the CDC path. This step covers how to waive the CDC-10 violation.

Tcl Console Messages Timing ×								? _ 🗆
Q 素 ≑ C	Q, 131 («	my_ip	_glbIclk to my_ip_axi_aclk			🖌 🌗 Critical warning (1)	🖌 🌖 Warning (0) 🛛 🖌 🚺 Info	(1) Hide A
General Information	Severity 🔨 1	ID	Description	Depth	Exception	Source (From)	Destination (To)	Category
Summary (by clock pair)	Critical	CDC-10	Combinatorial logic detected before a synchronizer	5	False Path	i_my_ip_supporysref_r_reg/C	i_my_ip_suppors_ff_reg[0]/E	Unsafe
Summary (by type)	Info	CDC-3	1-bit synchronized with ASYNC_REG property	5	False Path	i_my_ip_supporot_sync_reg/C	i_my_ip_suppos_ff_reg[0]/D	Safe
CDC Details (928) my_ip_drpclk to my_ip_axi_aclk (1) my_ip_glblclk to my_ip_axi_aclk (2) input port clock to my_ip_drpclk (2)								

1. To view a schematic of the violation, select the CDC-10 row in the CDC Report, and click the Schematic toolbar button 3.

Note: Alternatively, you can press **F4** to generate the schematic. However, using the toolbar button provides a more detailed schematic that includes all the levels of the downstream synchronizer.





- 2. To waive the violation, select the **CDC-10** row in the CDC Report, right-click, and select **Create Waiver**.
- 3. In the Create Waiver dialog box, enter a description, and click **OK**.

🝌 Create Wa	aiver	×
Create waiver f	for 1 cdc path	4
User:	Xilinx	\otimes
Description:	This is a safe CDC per review with the team	\otimes
Tags:		
Tcl Command	d Preview	
Q,		
create_waive	r -type CDC -id CDC-10 -from [get_pins i_my_ip_support_block/jesd204_i/inst/i_my_ip/i_tx/i_tx_cou	nters
?	OK Cano	> :el



IMPORTANT! A waiver tracks the date the waiver was added, the user that added the waiver, and a description of why the violation was waived. The date is automatically added by the system. The Tags field is an optional description or list of keywords that can be used for documentation purposes.

4. After the waiver is created, check the CDC Report.

To indicate that a waiver was created, the CDC-10 row is gray and disabled.

Note: Rows are only disabled in the Report CDC result window from which the waivers were created.



Report is out of date because path waivers w	vere cha	nged. Rerun							
ຊ ≚ ≑ ຕ	1	Q H 4	log_ my_ip_	glblclk to my_ip_axi_aclk		V	🕙 🕒 Critical warning (1) 🛛 🕑 V	Varning (0) 🕑 🚯 Info (1)	Hide All
General Information	^	Severity ^1	ID	Description	Depth	Exception	Source (From)	Destination (To)	Category
Summary (by clock pair)	- 1	Critical		Combinational logic detected before a synchronizer	5	False Path	i_my_ip_supportsysref_r_reg/C	i_my_ip_supportes_ff_reg[0]/D	Unsafe
Summary (by type)	- 1	Info	CDC-3	1-bit synchronized with ASYNC_REG property	5	False Path	i_my_ip_supporot_sync_reg/C	i_my_ip_suppores_ff_reg[0]/D	Safe
Summary (by waived endpoints)	- 1								
 CDC Details (928) 	- 1								
my_ip_drpclk to my_ip_axi_aclk (1)									
my_ip_glblclk to my_ip_axi_aclk (2)									
input port clock to my ip drpclk (2)									

5. To see the impact of the CDC-10 waiver, select **Reports** → **Timing** → **Report CDC** to rerun Report CDC.

Note: When a waiver is created or deleted, you must rerun Report CDC, Report DRC, or Report Methodology to see the updated results.

6. See the CDC Report to view the updated information.

The differences from the previous Summary by clock pair and Summary by type sections are highlighted in red in the following figures.

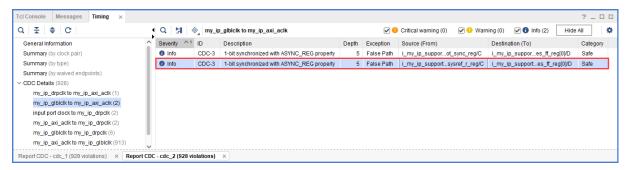
Q ≚ ♦ C	Summa	ry (by clock pair)									
General Information	Severity ^1	Source Clock	Destination C	Clock CDC	Стуре	Exceptio	ns Endpoints	Safe	Unsafe	Unknown	No ASYNC_REG
Summary (by clock pair)	Critical	my_ip_axi_aclk	my_ip_drpcll	k No C	Common Prim	ary Clock False P	ath 2	2 0	2	0	C
Summary (by type)	Oritical	my_ip_axi_aclk	my_ip_glblcl	lk No C	Common Prim	ary Clock False P	ath 942	2 12	351	579	185
Summary (by waived endpoints)	Info	my_ip_drpclk	my_ip_axi_a	iclk No C	Common Prim	ary Clock False P	ath -			0	
V CDC Details (928)	 Info 	my_ip_glblclk	my_ip_axi_a		Common Prim					-	
my_ip_drpclk to my_ip_axi_aclk (1)	 Info 	input port clock	my_ip_drpcl		Common Prim	-					
my_ip_glblclk to my_ip_axi_aclk (2) input port clock to my_ip_drpclk (2)	1 Info	my_ip_glblclk	my_ip_drpcl		Common Prim	-					
my_ip_axi_aclk to my_ip_drpclk (2)	 Info 	my_ip_drpclk	my_ip_glblcl	lk No C	Common Prim	ary Clock False P	ath 2	2 2	0	0	(
Tcl Console Messages Tim	ning ×										
	8	۰Q.	Summary	(by type)							
						Description					
Q X ♦ C General Information		Sever	ity ^1 IC	D	Count	Description	n CDC circu	itor			
Q X I ← C General Information Summary (by clock pair)		Sever O Cr	ity ^1 IC itical C	D CDC-1	Count 536	1-bit unknow		1			
Q X ♦ C General Information		Sever	ity ^1 IC itical C	D	Count			1	,		
Q X A C A C A C A C A C A C A C A C A C A		Sever Cr Cr	ity ^1 II itical C itical C	D CDC-1	Count 536	1-bit unknow	own CDC ci	rcuitry		synchror	izer
Q X ♦ C General Information Summary (by clock pair) Summary (by type)		Sever Cr Cr	ity ^1 II itical C itical C itical C	D CDC-1 CDC-4	Count 536 4	1-bit unknow Multi-bit unkr	own CDC ci al logic dete	rcuitry cted b	efore a		izer
Q. X. ♦ C General Information Summary (by clock pair) Summary (by type) Summary (by waived endpoints)	;)	Sever Cr Cr Cr Cr Cr Cr Cr	ity ^1 IC itical C itical C itical C itical C	D CDC-1 CDC-4 CDC-10	Count 536 4 186	1-bit unknow Multi-bit unkr Combination	own CDC ci al logic dete launch flop	rcuitry cted b to des	efore a tination		izer
Q X ♦ C General Information Summary (by clock pair) Summary (by type) Summary (by waived endpoints ✓ CDC Details (928) my_ip_drpclk to my_ip_axi_ my_ip_glblclk to my_ip_axi	aclk (1) _aclk (2)	Sever Cr Cr Cr Cr Cr Cr Cr	ity ^1 IC itical C itical C itical C itical C itical C	D CDC-1 CDC-4 CDC-10 CDC-11	Count 536 4 186 2	1-bit unknow Multi-bit unkr Combination Fan-out from	own CDC ci al logic dete launch flop th on a non-	rcuitry cted b to des FD pri	efore a tination mitive	clock	iizer
Q X ♦ C General Information Summary (by clock pair) Summary (by type) Summary (by waived endpoints ~ CDC Details (928) my_ip_drpclk to my_ip_axi_ my_ip_glblclk to my_ip_axi input port clock to my_ip_dr	;) _aclk (1) _aclk (2) rpclk (2)	Sever Cl Cl Cl Cl Cl Cl Cl Cl Cl Cl Cl Cl Cl	ity ^1 II itical C itical C itical C itical C itical C itical C	D CDC-1 CDC-4 CDC-10 CDC-11 CDC-13	Count 536 4 186 2 170	1-bit unknow Multi-bit unkn Combination Fan-out from 1-bit CDC pa	own CDC ci al logic dete launch flop th on a non- path on a n	rcuitry cted b to des FD pri on-FD	efore a tination mitive primitiv	clock /e	
Q X ♦ C General Information Summary (by clock pair) Summary (by type) Summary (by waived endpoints ✓ CDC Details (928) my_ip_drpclk to my_ip_axi_ input port clock to my_ip_ari input port clock to my_ip_dr my_ip_axi_aclk to my_ip_dr	aclk (1) aclk (2) pclk (2) rpclk (2)	Sever Cl Cl Cl Cl Cl Cl Cl Cl Cl Cl Cl Cl Cl	ity ^1 II itical C itical C itical C itical C itical C itical C itical C anning C	D CDC-1 CDC-4 CDC-10 CDC-11 CDC-13 CDC-14	Count 536 4 186 2 170 5	1-bit unknow Multi-bit unkn Combination Fan-out from 1-bit CDC pa Multi-bit CDC	own CDC ci al logic dete launch flop th on a non- path on a n controlled C	rcuitry cted b to des FD pri on-FD CDC s	efore a tination mitive primitiv tructure	clock /e detected	
Q X ♦ C General Information Summary (by clock pair) Summary (by type) Summary (by waived endpoints ✓ CDC Details (928) my_ip_drpclk to my_ip_axi_ my_ip_glblclk to my_ip_axi input port clock to my_ip_dr	aclk (1) _aclk (2) pclk (2) rpclk (2) rpclk (2) rclk (6)	 Sever Ci <l< td=""><td>ity ^1 II itical C itical C itical C itical C itical C itical C arning C fo</td><td>D CDC-1 CDC-4 CDC-10 CDC-11 CDC-13 CDC-14 CDC-15</td><td>Count 536 4 186 2 170 5 10</td><td>1-bit unknow Multi-bit unkr Combination Fan-out from 1-bit CDC pa Multi-bit CDC Clock enable 1-bit synchro</td><td>own CDC ci al logic dete launch flop th on a non- path on a n controlled (nized with A</td><td>rcuitry cted b to des FD pri on-FD CDC s SYNC</td><td>efore a tination mitive primitiv tructure _REG p</td><td>clock ve detected roperty</td><td></td></l<>	ity ^1 II itical C itical C itical C itical C itical C itical C arning C fo	D CDC-1 CDC-4 CDC-10 CDC-11 CDC-13 CDC-14 CDC-15	Count 536 4 186 2 170 5 10	1-bit unknow Multi-bit unkr Combination Fan-out from 1-bit CDC pa Multi-bit CDC Clock enable 1-bit synchro	own CDC ci al logic dete launch flop th on a non- path on a n controlled (nized with A	rcuitry cted b to des FD pri on-FD CDC s SYNC	efore a tination mitive primitiv tructure _REG p	clock ve detected roperty	

You can also view a summary with the list of waived endpoints.



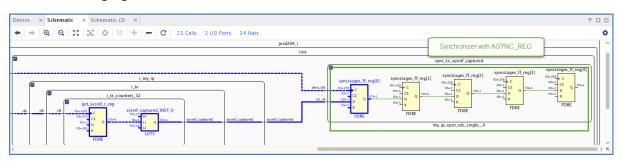
Tcl Console Messages Timing ×				
Q 素 ♠ C	4	Q Sun	nmary (by waived en	dpoints)
General Information	2	ID	Waived Endpoints	
Summary (by clock pair)		CDC-10	1	
Summary (by type)				
Summary (by waived endpoints)				
V CDC Details (928)				
my_ip_drpclk to my_ip_axi_aclk (1)				
my_ip_glblclk to my_ip_axi_aclk (2)				
input port clock to my_ip_drpclk (2)				
my_ip_axi_aclk to my_ip_drpclk (2)				
my_ip_glblclk to my_ip_drpclk (6)				
my_ip_axi_aclk to my_ip_glblclk (913)	~			
Report CDC - cdc_1 (928 violations) × Re	eport	CDC - cdc	_2 (928 violations)	×

The detailed section for the $my_{ip_glblclk}$ to $my_{ip_axi_aclk}$ CDC shows that the Critical CDC-10 was replaced with an Info CDC-3.



7. Select the new CDC-3 row, and click the Schematic toolbar button ¹. Double-click the **Q** pin of the output register to expand the schematic to match what is shown in the following figure.

The CDC path includes a 5-level synchronizer on the output of the selected destination register. This is the reason the CDC-10 was replaced with CDC-3 for this topology, as shown in the following figure.





IMPORTANT! By default, Report CDC only reports a single violation per endpoint and per clock pair. When multiple violations apply to the same endpoint, only the violation with the highest precedence is reported. Because CDC-10 has a higher precedence than CDC-3, only CDC-10 is reported when both CDC-10 and CDC-3 apply to the same endpoint. For more information on CDC rules precedence, see this link in the Vivado Design Suite User Guide: Design Analysis and Closure Techniques (UG906).

0

TIP: To report all of the CDC violations for each endpoint regardless of the precedence rules, use the command line option -all_checks_per_endpoint.

Step 4: Generating a Report for Waived Violations

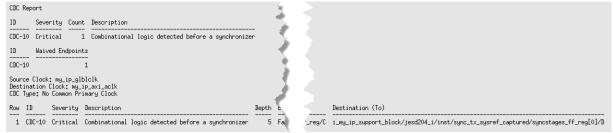
You can generate a report for the CDC, DRC, or methodology check violations that were waived. This step shows how to generate a report for waived CDC violations using the Tcl Console as well as the Vivado IDE menu commands.

Generating a Text Report for Waived Violations

1. In the Tcl Console, enter:

report_cdc -waived

In the CDC report, verify that a single CDC-10 violation is listed, because only one waiver was created.



Generating a Vivado IDE CDC Report for Waived Violations

- 1. Select Reports \rightarrow Timing \rightarrow Report CDC.
- 2. In the Report CDC dialog box, enable *Report only waived paths*, and click OK.
- 3. In the CDC Report, check the Summary (by clock pair) and CDC Details to verify that a single CDC-10 violation is listed.

Note: The icon next to the violation shows that the violation was waived *8*.



Q ≚ ≑ C	🖣 🔍 Summa	ry (by clock p	air)										
General Information	Severity ^ 1	Source Cloc	k Destination Clock	CDC Type	Exceptions	Endpoints	Safe	Unsafe	Unknown	No ASYN	C_REG		
Summary (by clock pair)	Critical	my_ip_glblc	lk my_ip_axi_aclk	No Common Primary Clock	False Path		1 0	1	0		0		
Summary (by type)													
Summary (by waived endpoints)													
CDC Details (1)													
my_ip_glblclk to my_ip_axi_aclk (1)													
	port CDC - cdc_2 (92	8 violations)	× Report CDC - cdc_	_3 (1 waived) ×									? _ 🗆
cl Console Messages Timing ×			X Report CDC - cdc_	· · · ·		Critic	al warning	g (1) 🕑	9 Warnin	ıg (0) 🕑) 🚯 Info (0)	Hide	
cl Console Messages Timing ×		⊗ _a my_ip_gl		<	Depth Exce	0 -	al warnin <u>c</u> rce (From)	_		ig (0) 🕑 Destination		Hide	
clConsole Messages Timing × Q X ♦ C	Q H	lD I	Ibicik to my_ip_axi_acik	<	Depth Exce 5 Falsi	ption Sou	rce (From)			Destination			All
Cl Console Messages Timing × Q X ♦ C General Information	Q B	lD I	Ibicik to my_ip_axi_acik	ς (ption Sou	rce (From)			Destination	n (To)		Category
Cl Console Messages Timing × Q X X I ⊕ C General Information Summary (by clock pair)	Q B	lD I	Ibicik to my_ip_axi_acik	ς (ption Sou	rce (From)			Destination	n (To)		All Category
Cl Console Messages Timing × Q X → Q C General Information Summary (by lock pair) Summary (by waived endpoints)	Q B	lD I	Ibicik to my_ip_axi_acik	ς (ption Sou	rce (From)			Destination	n (To)		All Category
Q ¥ ♥ C General Information Summary (by clock pair) Summary (by type)	Q B	lD I	Ibicik to my_ip_axi_acik	ς (ption Sou	rce (From)			Destination	n (To)		All Category

Step 5: Generating a Text Report with Details for Waived Violations

In this step, you generate text reports with additional details, including a list of all of the rules and all of the violations regardless of the waivers.

Generating a List of Rules with Waived Violations

1. In the Tcl Console, enter:

report_cdc -details -show_waiver

2. Verify that the my_ip_glblclk to my_ip_axi_aclk CDC-10 violation is waived and the two CDC-3 violations are not waived.

Note: In the text report, all of the rules are reported, whether they were waived or not. The Waived column indicates the status of the rule.

CDC Rep	ort									
ID	Severity	Count	Description				ς.			
CDC-11 CDC-13 CDC-14	Critical Info Critical Info Critical Critical Critical Critical Warning	10 4 5 186 2 170 5	1-bit unknown CDC circuitry 1-bit synkronzed with RSNR_REG property Nult-bit unknown CDC circuitry Reynkronous reset synkronized with RSNK_REG pr Combinatorial logic detected before a synkronize Lobit CDC path on a non-FD primitive Lobit CDC path on a non-FD primitive Clock emails controlled CDC structure detected	operty r						
ID CDC-10	Waived 1									
Destin	Clock: my_ ation Clock we: No Comm	: ny_ip	_axi_aclk				5			
2 C	 IC-3 Info IC-3 Info) 1) 1	escription -bit synchronized with ASYNC_REG property -bit synchronized with ASYNC_REG property ombinatorial logic detected before a synchronizer	 5 5	False Path False Path	Source (From) i_mg_ip_support i_mg_ip_support i_mg_ip_support		_32/got_sysref_r_reg/C	Destination (To) i.mg.ip.support.block/jest204_i/inst/sync_tx_sync/syncstages_ff_reg[0]/D i.mg.ip.support.block/jest204_i/inst/sync_tx_synef_captured/syncstages_ff_reg[0]/D i.mg.ip.support.block/jest204_i/inst/sync_tx_synef_captured/syncstages.ff_reg[0]/D	Waived N N Y



Generating a List of All Violations Regardless of the Waivers

1. In the Tcl Console, enter:

CDC Deserve

 \bigcirc

report_cdc -no_waiver

2. In the text report, verify that the table matches the original report from Report CDC before the CDC-10 waiver was created.

SeveritySource ClockDestination ClockCDC TypeExceptionsEndpointsSafeUnsafeUnknownNo ASYNC_REGCriticalmy_ip_glblclkmy_ip_axi_aclkNo Common Primary ClockFalse Path21100Criticalmy_ip_axi_aclkmy_ip_drpclkNo Common Primary ClockFalse Path21100Criticalmy_ip_axi_aclkmy_ip_axi_aclkNo Common Primary ClockFalse Path21100Criticalmy_ip_drpclkmy_ip_axi_aclkNo Common Primary ClockFalse Path22000Infomy_ip_drpclkmy_ip_atri_aclkNo Common Primary ClockFalse Path11000Infoinput port clockmy_ip_drpclkNo Common Primary ClockFalse Path22000Infomy_ip_glblclkNo Common Primary ClockFalse Path22000Infomy_ip_glblclkNo Common Primary ClockFalse Path22000Infomy_ip_glblclkNo Common Primary ClockFalse Path22000	LUL Kepor	t									
Critical my_ip_axi_aclk my_ip_drpclk No Common Primary Clock False Path 2 0 0 Critical my_ip_axi_aclk my_ip_glblclk No Common Primary Clock False Path 942 12 351 579 185 Info my_ip_drpclk my_ip_axi_aclk No Common Primary Clock False Path 942 12 351 579 185 Info my_ip_drpclk No Common Primary Clock False Path 1 0 0 0 Info input port clock my_ip_drpclk No Common Primary Clock False Path 2 2 0 0 Info my_ip_glblclk No Common Primary Clock False Path 6 6 0 0	Severity	Source Clock	Destination Clock	CDC Type	Exceptions	Endpoints	Safe	Unsafe	Unknown	No ASYNC_REG	
	Critical Critical Info Info Info	my_ip_axi_aclk my_ip_axi_aclk my_ip_drpclk input port clock my_ip_glblclk	my_ip_drpclk my_ip_glblclk my_ip_axi_aclk my_ip_drpclk my_ip_drpclk my_ip_drpclk	No Common Primary Clock No Common Primary Clock No Common Primary Clock No Common Primary Clock No Common Primary Clock	False Path False Path False Path False Path False Path	942 1 2	12 1 2 6	351 0 0	0 579 0 0 0 0	0 0 185 0 0 0 0	

TIP: You can also generate a list of all violations regardless of the waivers from the Vivado IDE. Select **Reports** \rightarrow **Timing** \rightarrow **Report CDC**. In the Report CDC dialog box, enable **Ignore all waivers**, and click **OK**.

Step 6: Waiving Multiple CDC Violations

The my_ip_axi_aclk to my_ip_drpclk CDC includes two Critical CDC-11 violations. This step covers how to waive both CDC-11 violations simultaneously.

Tcl Console Messages Timing ×								? _ 🗆 🖸
Q ≭ ≑ C	Q H	⊗ my_ip	_axi_aclk to my_ip_drpclk		\checkmark	🌖 Critical warning (2) 🛛 🖌 🕓 W	arning (0) 🕑 🚺 Info (0) 🛛 Hi	de All 🌼
General Information	Severity	N1 ID	Description	Depth	Exception	Source (From)	Destination (To)	Category
Summary (by clock pair)	Oritical	CDC-11	Fan-out from launch flop to destination clock	5	False Path	i_my_ip_supporetch_reg[10]/C	i_my_ip_supportff_reg[0]/CLR	Unsafe
Summary (by type)	Critical	CDC-11	Fan-out from launch flop to destination clock	5	False Path	i_my_ip_supporetch_reg[10]/C	i_my_ip_supportff_reg[0]/CLR	Unsafe
Summary (by waived endpoints)								
V CDC Details (928)								
my_ip_drpclk to my_ip_axi_aclk (1)								
my_ip_glblclk to my_ip_axi_aclk (2)								
input port clock to my_ip_drpclk (2)								
my_ip_axi_aclk to my_ip_drpclk (2)								
	¥							
Report CDC - cdc_1 (928 violations) × Report	rt CDC - cdc_2 (92	violations)	×					

1. To waive the violations, select the **CDC-11** rows in the CDC Report, right-click, and select **Create Waiver**.



Tcl Console Messages Timing ×								? _	0 6
Q ₹ ≑ C	Q 🔄 🔄 🗞 my_ip	_axi_aclk to my_ip_drpclk			🖌 🌗 Critical warning (2) 🛛 🖌	Warning (0)	🕑 🚯 Info (0) 🛛 Hi	de All	•
General Information	Severity ^1 ID	Description	Depth	Exc	ception Source (From)	Destination	(To)	Category	
Summary (by clock pair)	Critical CDC-11	Fan-out from launch flop to destination clock	ş	Fo	los Dath i mu in aunnar atab raal10		nnortff_reg[0]/CLR	Unsafe	
Summary (by type)	Oritical CDC-11	Fan-out from launch flop to destination clock	5		Path Properties	Ctrl+E	ortff_reg[0]/CLR	Unsafe	
Summary (by waived endpoints)			·		Elide Setting	,	•		
V CDC Details (928)				۰.	Highlight	,			
my_ip_drpclk to my_ip_axi_aclk (1)					Unhighlight				
my_ip_glblclk to my_ip_axi_aclk (2)				~					
input port clock to my_ip_drpclk (2)				\otimes	Mark	,			
my_ip_axi_aclk to my_ip_drpclk (2)					Unmark	Ctrl+Shift+N			
my_ip_glblclk to my_ip_drpclk (6)				н	Schematic				
my_ip_axi_aclk to my_ip_glblclk (913)					View Path Report				
my_ip_drpclk to my_ip_glblclk (2)					Report Timing on Source to Destination.				
					Set Maximum Delay				
					Set Bus Skew	1			
				2)	Create Waiver				
· · · · · · · · · · · · · · · · · · ·				-	Export to Spreadsheet				
Report CDC - cdc_1 (928 violations) × Report CDC	C - cdc_2 (928 violations)	×							

2. In the Create Waiver dialog box, enter a description, and click **OK**.

🝌 Create W	aiver ×
Create waivers	for 2 cdc paths
User:	Xilinx
Description:	Safe fanout. Circuitry has been reviewed
Tags:	
Tcl Command	1 Preview
Q	
	r -type CDC -id CDC-11 -from [get_pins {i_my_ip_support_block/jesd204_i/inst/i_my_ip_reset_block/stre r -type CDC -id CDC-11 -from [get_pins {i_my_ip_support_block/jesd204_i/inst/i_my_ip_reset_block/stre
<	>
?	OK Cancel

In the Timing Report, the two selected rows are disabled when the waivers are created.

Note: One waiver is created for each selected row. In this example, two waivers are created.

Severity \land 1	ID	Description	Depth	Exception	Source (From)	Destination (To)	Category
Oritical	CDC-11	Fan-out from launch flop to destination clock	5	False Path	i_my_ip_suppotch_reg[10]/C	i_my_ip_supporff_reg[0]/CLR	Unsafe
Critical	CDC-11	Fan-out from launch flop to destination clock	5	False Path	i_my_ip_suppotch_reg[10]/C	i_my_ip_supporff_reg[0]/CLR	Unsafe

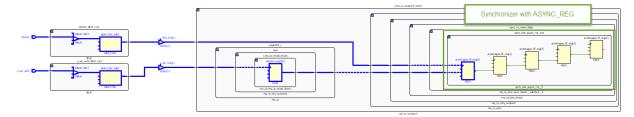
- 3. Select **Reports** → **Timing** → **Report CDC** to rerun Report CDC. In the Report CDC dialog box, make sure that *Report only waived paths* is unchecked, and click **OK**.
- 4. In the CDC Report, look at the my_ip_axi_aclk to my_ip_drpclk CDC.



The two Critical CDC-11 violations were replaced with two Info CDC-9 violations. Based on the CDC precedence rules, waiving CDC-11 unmasks CDC-9 for this circuit.

Q ¥ ♦ C	Q H 4	⊗_ my_ip	o_axi_aclk to my_ip_drpclk		V 🌖 🗸	critical warning (0) 🛛 🗹 🕒 Warnin	g (0) 🕑 🚯 Info (2) 🛛 Hide Al	I 🕹
General Information	Severity ^1	ID	Description	Depth	Exception	Source (From)	Destination (To)	Category
Summary (by clock pair)	Info	CDC-9	Asynchronous reset synchronized with ASYNC_REG property	5	False Path	i_my_ip_supporetch_reg[10]/C	i_my_ip_supportff_reg[0]/CLR	Safe
Summary (by type)	Info	CDC-9	Asynchronous reset synchronized with ASYNC_REG property	5	False Path	i_my_ip_supporetch_reg[10]/C	i_my_ip_supportff_reg[0]/CLR	Safe
<pre>CDC Details (928) my_lp_drpclk to my_lp_axi_aclk (1) my_ip_glblclk to my_ip_axi_aclk (2) </pre>								
input port clock to my_ip_drpclk (2) my_ip_axi_aclk to my_ip_drpclk (2)								
my_ip_glblclk to my_ip_glblclk (6) my_ip_axi_aclk to my_ip_glblclk (913)								

- To view a schematic of the violation, select the CDC-9 row in the CDC Report, and click the Schematic toolbar button 3.
- 6. Verify that there is a 5-level synchronizer on the destination clock domain.



7. Compare the new Summary (by type) information with the information from the previous CDC Report.

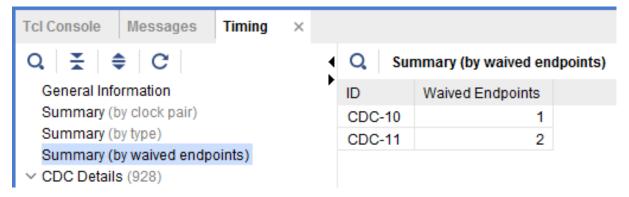
In the updated CDC Report, the two CDC-11 violations are no longer listed. Instead, there are two new CDC-9 violations.

Q 꽃 ♦ C	Q Summa	ry (by type)		
General Information	Severity ^1	ID	Count	Description
Summary (by clock pair)	Critical	CDC-1	536	1-bit unknown CDC circuitry
Summary (by type)	Critical	CDC-4	4	Multi-bit unknown CDC circuitry
Summary (by waived endpoints)	Critical	CDC-10	186	Combinational logic detected before a synchronizer
CDC Details (928)	Oritical	CDC-13	170	1-bit CDC path on a non-FD primitive
my_ip_drpclk to my_ip_axi_aclk (1)	Critical	CDC-14	5	Multi-bit CDC path on a non-FD primitive
my_ip_glblclk to my_ip_axi_aclk (2)	😐 Warning	CDC-15	10	Clock enable controlled CDC structure detected
input port clock to my_ip_drpclk (2)	Info	CDC-3	10	1-bit synchronized with ASYNC_REG property
my_ip_axi_aclk to my_ip_drpclk (2)	 Info 	CDC-9	7	Asynchronous reset synchronized with ASYNC_REG proper
my_ip_glblclk to my_ip_drpclk (6)				
my_ip_axi_aclk to my_ip_glblclk (913)				
my_ip_drpclk to my_ip_glblclk (2)				

8. Look at the Summary (by waived endpoints) information.



In the updated CDC Report, there are three waived endpoints. This number is different from the number of waived violations (2), because CDC-11 is a multi-bit violation.



9. Generate different text reports and compare the results with previous reports.

For example, you can run the following Tcl commands:

report_cdc -details
report_cdc -details -waived
report_cdc -details -show_waiver
report_cdc -details -no_waiver

The following report was generated using the report_cdc -details -waived Tcl command and shows that three violations were waived.

CDC Report	
ID Severity Count Description	
CDC-10 Critical 1 Combinatorial logic detected before a synchronizer CDC-11 Critical 2 Fan-out from launch flop to destination clock	
ID Waived	
CDC-10 1	
CDC-11 2	
Source Clock: ng_ip_glblclk Destination Clock: ng_ip_axi_aclk CDC Tupe: No Compon Primary Clock	
Row ID Severity Description Depth Exception S	Source (Fron)
	i_nw_ip_support /s_32/got_sysref_r_reg/C i_nw_ip_support_block/jesd204_i/inst/sync_tx_sysref_captured/syncstages_ff_reg[0]/D
Source Clock: mg_ip_axi_aclk Bestmation Clock: mg_ip_dPclk CC Type: No formor Frimary Clock	
Row ID Severity Description Depth Exception Source	e (From) Destination (To)
1 CDC-11 Critical Fam-out from launch flop to destination clock 5 False Path i_mg_ 2 CDC-11 Critical Fam-out from launch flop to destination clock 5 False Path i_mg_1	ip.support.block/ reg[10]/C i.mg.ip.support.block/i.jes204.phy/inst/jes204.phy.block.i/sync.rxtreset.data/spm.odc_async.rst.inst/arststages.ff.reg[0]/Cl sg[10]/C i.mg.ip.support.block/i.jes204.phy/inst/jes204.phy.block.i/sync.txtreset.data/spm.odc_async.rst.inst/arststages.ff.reg[0]/Cl

Step 7: Exporting Waivers

In this step, you export waivers with the write_waivers Tcl command.

Note: The XDC output file can be imported using the read_xdc or source Tcl commands.

1. To export the CDC waivers, enter: write_waivers -type cdc waivers.xdc.

TIP: Alternatively, because there are no DRC or methodology waivers, you can enter:

write_waivers waivers.xdc Of write_xdc -type waiver waivers.xdc.

2. Open the waivers.xdc file to view the three waivers.

 \bigcirc



Note: The following example is reformatted to better show the different command line options.

```
create_waiver -type CDC -id {CDC-10} -user "Xilinx" \
  -desc "This is a safe CDC per review with the team"
  -from [get_pins i_my_ip_support_block/jesd204_i/inst/i_my_ip/i_tx/
i_tx_counters_32/got_sysref_r_reg/C] \
  -to [get_pins {i_my_ip_support_block/jesd204_i/inst/
sync_tx_sysref_captured/syncstages_ff_reg[0]/D}] \
  -timestamp "<timestamp>" ;#1
create_waiver -type CDC -id {CDC-11} -user "Xilinx" \
  -desc "Safe fanout. Circuitry has been released"
  -from [get_pins {i_my_ip_support_block/jesd204_i/inst/
i_my_ip_reset_block/stretch_reg[10]/C}] \
  -to [get_pins {i_my_ip_support_block/i_jesd204_phy/inst/
jesd204_phy_block_i/sync_rx_reset_data/xpm_cdc_async_rst_inst/
arststages_ff_reg[0]/CLR}] \
  -timestamp "<timestamp>" ;#1
create_waiver -type CDC -id {CDC-11} -user "Xilinx" \
  -desc "Safe fanout. Circuitry has been released" \
  -from [get_pins {i_my_ip_support_block/jesd204_i/inst/
i_my_ip_reset_block/stretch_reg[10]/C}] \
  -to [get_pins {i_my_ip_support_block/i_jesd204_phy/inst/
jesd204_phy_block_i/sync_tx_reset_data/xpm_cdc_async_rst_inst/
arststages_ff_reg[0]/CLR}] \
  -timestamp "<timestamp>" ;#2
```

Step 8: Using the create_waiver Command

Waivers added from the Report CDC dialog box are created using the create_waiver command. You can view these commands as follows.

Note: You can use the create_waiver command line command for CDC, DRC, and methodology waivers. The options differ slightly depending on whether you are creating a CDC, DRC, or methodology waiver. For more information, including information on the different options, see the create_waiver command in the *Vivado Design Suite Tcl Command Reference Guide* (UG835).

- 1. Open the Vivado journal file (vivado.jou) to see the three distinct create_waiver commands issued by the Vivado IDE.
- 2. Scroll through the history of the Tcl Console to see the same three create_waiver commands.

TIP: The -from and -to options are used to specify the startpoints and endpoints. When a waiver is set from the Report CDC dialog box, both -from and -to are specified to match the exact violation. However, you can specify a CDC waiver using only the -from option or only the -to option, but more paths might be waived than expected.

 \bigcirc



Step 9: Waiving Multiple CDC Violations

In this step, you waive multiple CDC violations simultaneously.

1. In the CDC Report, view the my_ip_axi_aclk to my_ip_glblclk CDC under CDC Details.

This crossing has five CDC-14 violations, which are multi-bit violations. The five CDC-14 violations all start from the same two register clock pins:

i_my_ip_support_block/jesd204_i/inst/tx_cfg_test_modes_reg[2:1]/C

TIP: You can sort the table by the column ID to more easily see the five CDC-14 violations.

2 ¥ ≑ C	<u>।</u> २ 🔄 🕷		axi_aclk to my_ip_glblclk		🕑 🕛 Critical	warning (901) 🛛 🗑 🕛 Warning (1	0) 🕑 🚺 Info (2) 🛛 Hide	e All
General Information	^ Severity	ID ^ 1	Description	Depth	Exception	Source (From)	Destination (To)	Category
Summary (by clock pair)	Warning	CDC-15	Clock enable controlled CDC structure detected	0	False Path	i_my_ip_supportmodes_reg[1]/C	i_my_ip_suppod_nls_r_reg/D	Safe
Summary (by type)	Oritical	CDC-14	Multi-bit CDC path on a non-FD primitive	0	False Path	i_my_ip_supportodes_reg[2:1]/C	i_my_ip_supp/TXDATA[2:1]	Unknown
Summary (by waived endpoints)	O Critical	CDC-14	Multi-bit CDC path on a non-FD primitive	0	False Path	i my ip supportodes reg[2:1]/C	i my ip supp/TXDATA[2:1]	Unknown
CDC Details (928)	O Critical	CDC-14	Multi-bit CDC path on a non-FD primitive	0	False Path	i_my_ip_supportodes_reg[2:1]/C	i my ip supp/TXDATA[2:1]	Unknowr
my_ip_drpclk to my_ip_axi_aclk (1)	Critical	CDC-14	Multi-bit CDC path on a non-FD primitive	0	False Path	i my ip support odes reg[2:1]/C		Unknowr
my_ip_glblclk to my_ip_axi_aclk (2)	Oritical	CDC-14	Multi-bit CDC path on a non-FD primitive	0	False Path	i my ip support odes reg[2:1]/C	i my ip supp/TXDATA[2:1]	Unknow
input port clock to my_ip_drpclk (2)	O Critical	CDC-13	1-bit CDC path on a non-FD primitive	0	False Path	i_my_ip_supportmodes_reg[2]/C	i_my_ip_suppoT/TXCTRL2[0]	Unsafe
my_ip_axi_aclk to my_ip_drpclk (2)	Oritical	CDC-13	1-bit CDC path on a non-FD primitive	0	False Path	i_my_ip_supportmodes_reg[2]/C	i_my_ip_suppoT/TXCTRL2[1]	Unsafe
my_ip_glblclk to my_ip_drpclk (6)	Critical	CDC-13	1-bit CDC path on a non-FD primitive	0	False Path	i_my_ip_supportmodes_reg[2]/C	i_my_ip_suppoT/TXCTRL2[3]	Unsafe
my_ip_axi_aclk to my_ip_glblclk (913) my_ip_drpclk to my_ip_glblclk (2)	Critical	CDC-13	1-bit CDC path on a non-FD primitive	0	False Path	i my ip supportmodes reg[1]/C	i my ip suppST/TXDATA[0]	Unsafe

2. Because i_my_ip_support_block/jesd204_i/inst/ tx_cfg_test_modes_reg[*]/C matches five pins and you only need to target two of

those five pins, construct the list of startpoints as follows:

```
set startpoints [list \
   [get_pins i_my_ip_support_block/jesd204_i/inst/
tx_cfg_test_modes_reg[1]/C] \
   [get_pins i_my_ip_support_block/jesd204_i/inst/
tx_cfg_test_modes_reg[2]/C] \
   ]
```

3. To waive the five CDC-14 violations, use the create_waiver Tcl command with the -from option:

```
create_waiver -type {CDC} -id {CDC-14} -user {Xilinx} -desc {No more CDC 14!} -from $startpoints
```

- 4. From the Vivado IDE, select **Reports** \rightarrow **Timing** \rightarrow **Report CDC** to rerun Report CDC.
- 5. In the CDC Report, verify that the CDC-14 violations are no longer reported in the Summary section.



	1	Q Summar	ry (by type)		
General Information	1	Severity ^1	ID	Count	Description
Summary (by clock pair)		Critical	CDC-1	536	1-bit unknown CDC circuitry
Summary (by type)		Critical	CDC-4	4	Multi-bit unknown CDC circuitry
Summary (by waived endpoints)		Critical	CDC-10	186	Combinational logic detected before a synchronizer
CDC Details (923)		Critical	CDC-13	170	1-bit CDC path on a non-FD primitive
my_ip_drpclk to my_ip_axi_aclk (1)		👴 Warning	CDC-15	10	Clock enable controlled CDC structure detected
my_ip_glblclk to my_ip_axi_aclk (2)		 Info 	CDC-3	10	1-bit synchronized with ASYNC_REG property
input port clock to my_ip_drpclk (2)		1 Info	CDC-9	7	Asynchronous reset synchronized with ASYNC_REG property
my_ip_axi_aclk to my_ip_drpclk (2)	~				

6. To report only the waived violations, enter:

report_cdc -details -waived

The following figure shows the waived CDC violations in two different tables. The first table shows the 5 CDC-14 violations waived as multi-bit violations. The second table shows the 10 single-bit violations, calculated by multiplying the 5 multi-bit violations by 2 bits per multi-bit violation.

	Critical Critical	 1 2	Description Combinatorial logic detected b Ean-out from launch floe to de Multi-bit CDC path on a non-FI	estination_cloc										
ID 6 CDC-10 CDC-11 CDC-14	Waived 1 2 10													
Destinati CDC Type:	: No Commo	<: my_i non Pri	p_axi_aclk mary Clock											
Row ID	Sever	erity .	Description		1	Jepth E	Exception	Source (Fro	m)					
			Combinatorial logic detected bef	fore a synchron	izer	5 F	False Path	i_my_ip_sup	port_bloo	k/jesd204	_i/inst/i	_my_ip/i_t:	x/i_tx_c	ounters_32/gou
Source Cl Destinati	lock: my_i ion Clock: : No Commo	_ip_axi <: my_i mon Prim	_aclk	fore a synchron		5 F		i_my_ip_sup ce (From)	port_bloo	k/jesd204	_i/inst/i	_my_ip/i_t:	x/i_tx_c	ounters_32/gou
Source Cl Destinati CDC Type: Row ID 1 CDC-	Clock: my_i tion Clock: No Commo Sever 11 Criti	_ip_axi <: my_i mon Pri erity tical	_aclk p_drpclk mary Clock	 tination clock	Depth 5	Except False	tion Sour Path i_my.	ce (From) ip_support_			st/i_my_i	p_reset_bl		ounters_32/gou tch_reg[10]/C tch_reg[10]/L
Source Cl Destinati CDC Type: Row ID 1 CDC- 2 CDC- Source Cl Destinati	ilock: my_i ion Clock: No Commo -11 Criti -11 Criti ilock: my_i ion Clock:	_ip_axi <: my_i mon Pri erity tical tical tical _ip_axi. <: my_i	_aclk p_drpclk mary Clock Description Fan-out from launch flop to dest Fan-out from launch flop to dest _aclk	 tination clock	Depth 5	Except False	tion Sour Path i_my.	ce (From) ip_support_			st/i_my_i	p_reset_bl		 tch_reg[10]/P
Source Cl Destinati CDC Type: Row ID 1 CDC- 2 CDC- Source Cl Destinati	Clock: my_i ion Clock: : No Commo Sever 	_ip_axi. <: my_i mon Prin erity tical tical _ip_axi <: my_i mon Prin	_aclk p_drpclk mary [Jock Description Fan-out from launch flop to dest Fan-out from launch flop to dest _aclk _aclk	tination clock tination clock	Depth 5 5	Except False	tion Sour Path i_my, Path i_my,	ce (From) _ip_support_ _ip_support_			st/i_my_i	p_reset_bl		 tch_reg[10]/P

7. To export all the waivers inside a script and verify that a total of four waivers were added, enter:

write_waivers -type cdc waivers.xdc -force

Note: Because the waivers.xdc file already exists, the -force option must be specified to override the file.



 \bigcirc

TIP: Alternatively, because there are no DRC or methodology waivers, you can enter:

```
write_waivers waivers.xdc -force
or
```

write_xdc -type waiver waivers.xdc -force

The list of waivers inside waivers.xdc appears as follows.

```
WRITE CDC Waivers
cnd; write_waivers -type cdc -file waivers,xdc -force
current_instance -quiet
create_waiver -type CDC -id (CDC-10} -user "Xilinx" -desc "This is a safe CDC per review with the team" -from [get_pins i_mg_ip_support_block/jesd204_i/inst/i_mg_ip_it_x/i_tx_counters_32/got_
create_waiver -type CDC -id (CDC-11) -user "Xilinx" -desc "Safe fanout, Circuitry has been released" -from [get_pins i_mg_ip_support_block/jesd204_i/inst/i_mg_ip_rest_block/stretch_reg[1]/C)
create_waiver -type CDC -id (CDC-14) -user "Xilinx" -desc "No more CDC-14!" -from [list [get_pins {i_mg_ip_support_block/jesd204_i/inst/tx_cfg_test_modes_reg[1]/C] [get_pins {i_mg_ip_support_block/
create_waiver -type CDC -id (CDC-14) -user "Xilinx" -desc "No more CDC-14!" -from [list [get_pins {i_mg_ip_support_block/jesd204_i/inst/tx_cfg_test_modes_reg[1]/C] [get_pins {i_mg_ip_support_
current_instance -quiet
```

8. To import the waivers.xdc file, enter:

read_xdc waivers.xdc

The following warnings show that duplicate waivers were not added to the existing waivers. Only waivers that are exact duplicates of existing waivers are rejected.

```
WARNING: [Vivado_Tcl 4-935] Waiver ID 'CDC-10' is a duplicate and will
not be added again.
WARNING: [Vivado_Tcl 4-935] Waiver ID 'CDC-11' is a duplicate and will
not be added again.
WARNING: [Vivado_Tcl 4-935] Waiver ID 'CDC-11' is a duplicate and will
not be added again.
WARNING: [Vivado_Tcl 4-935] Waiver ID 'CDC-14' is a duplicate and will
not be added again.
```

Step 10: Waiving Multiple DRC Violations

In this step, you waive multiple DRC violations simultaneously.

- 1. Select **Reports** \rightarrow **Report DRC**.
- 2. In the Report DRC dialog box, leave all settings at their default, and click **OK**.



🝌 Report DRC			×
Check design against sele	ected rule decks and/or individual desi	ign rules.	4
Results name:	drc_1		
Waivers Apply waivers Display only v Ignore all waivers Rule Decks Vivado Rule Deck Vivado Rule Deck Vivado Rule Deck P @ default @ opt_check @ placer_ch @ router_che @ bitstream @ incr_eco_ @ eco_chec	is (7) is ecks ecks _checks checks	Q X ♦ ✓ All Rules (2135) > Netlist (737) > Pin Planning (110) > Clocking (4) > Memory (148) > Floorplan (10) > Implementation (239)	Î
 ✓ Open in a new tab ? 		Physical Configuration (842) OK	✓ Cancel

3. In the DRC Report, right-click **UCIO#1**, and select **Create Waiver** to create a waiver for the UCIO-1 violations.

Note: The UCIO#1 violation combines 125 individual violations into a single violation. Similarly, the NSTD#1 violation covers 113 ports.



	C × Timing			
੨ ੜ ≑ 📲 🔛	🖌 🔒 3 Critical Warnings 🛛 🖌 1 Warning 🛛 Hide All			
lame	Details			
V (1) NSTD-1 (1)				
• NSTD #1	113 out of 125 logical ports use I/O standard (IOSTANDARD) value 'DEFAULT', instead of a user as components to which it is connected. To correct this violation, specify all I/O standards. This design command: set_property SEVERITY {Warning} [get_drc_checks NSTD-1]. NOTE: When using the Vi s axi araddr[10], s axi araddr[9], s axi araddr[8], s axi araddr[7], s axi araddr[6], s axi araddr[7].	will f ivado	ail to generate a bitstream un Runs infrastructure (e.g. laun	less all logic ch_runs Tcl
UCIO-1 (1)				
	125 out of 125 logical ports have no user assigned specific location constraint (LOC). This may car	use I/0	Contention or incompatibility	with the bo
0 UCIO #1	correct this violation, specify all pin locations. This design will fail to generate a bitstream unless a [get_drc_checks UCIO-1]. NOTE: When using the Vivado Runs infrastructure (e.g. launch_runs Tc		Сору	
	correct this violation, specify all pin locations. This design will fail to generate a bitstream unless a		· · · · · · · · · · · · · · · · · · ·	
 UCIO #1 Implementation (1) 	correct this violation, specify all pin locations. This design will fail to generate a bitstream unless a [get_drc_checks UCIO-1]. NOTE: When using the Vivado Runs infrastructure (e.g. launch_runs Tc		Сору	c Ctrl+E
	correct this violation, specify all pin locations. This design will fail to generate a bitstream unless a [get_drc_checks UCIO-1]. NOTE: When using the Vivado Runs infrastructure (e.g. launch_runs Tc		Copy Violation Properties	
 Implementation (1) 	correct this violation, specify all pin locations. This design will fail to generate a bitstream unless a [get_drc_checks UCIO-1]. NOTE: When using the Vivado Runs infrastructure (e.g. launch_runs Tc		Copy Violation Properties Auto-Select Objects Auto-Mark Objects	
 Implementation (1) Routing (1) 	correct this violation, specify all pin locations. This design will fail to generate a bitstream unless a [get_drc_checks UCIO-1]. NOTE: When using the Vivado Runs infrastructure (e.g. launch_runs Tc		Copy Violation Properties Auto-Select Objects Auto-Mark Objects Wrap Lines	
 Implementation (1) Implementation (1) Routing (1) Chip Level (1) 	correct this violation, specify all pin locations. This design will fail to generate a bitstream unless a [get_drc_checks UCIO-1]. NOTE: When using the Vivado Runs infrastructure (e.g. launch_runs Tc		Copy Violation Properties Auto-Select Objects Auto-Mark Objects	

4. In the Create Waiver dialog box, look at the output in Tcl Command Preview, and click OK.

🝌 Create Wa	aiver	\times
Create waiver f	or 1 violation	λ
User:	Xilinx	\otimes
Description:	Waive UCIO DRC violations	\otimes
Tags:		
Tcl Command	1 Preview	
Q		
create_waive	r -of_objects [get_drc_violations -name drc_1 {UCIO-1#1}] -user Xilinx -description {Waive UCIO DRO	C viol
<		>
?	ОК Салс	el

5. To generate the drc_waivers.xdc file and verify that the waiver is waiving all 125 objects, enter:

write_waivers -type DRC drc_waivers.xdc

6. In the XDC file, look at the expanded port list, and notice that some of the strings from the violations message were converted to wildcards (*).



Strings are automatically converted to wildcards for UCIO-1, NSTD-1, TIMING-15, and TIMING-16 type violations. For UCIO-1, the numbers of objects in the violations are replaced with wildcards, because the numbers of elements are not meaningful.

*
* WRITE DRC WAIVERS
cmd: write_waivers -type DRC drc_waivers.xdc
current_instance -quiet
create_waiver -type DRC -id {UCIO-1} -user "Xilinx" -desc "Waive UCIO DRC violations" -objects [get_ports { refclkOp glblclkp refclkOn tx_start_of_frame[3] tx_start_of_multiframe[3] glblclkn
tx_reset drpclk tx_start_of_multiframe[2] txp[3] tx_start_of_multiframe[0] tx_start_of_frame[2] tx_start_of_frame[1] s_axi_rready tx_sync tx_sysref tx_aresetn s_axi_rdata[1] s_axi_rvalid
s_axi_rresp[0] s_axi_rresp[1] s_axi_rdata[0] s_axi_rdata[2] s_axi_rdata[3] s_axi_rdata[4] s_axi_rdata[9] s_axi_rdata[5] s_axi_rdata[10] s_axi_rdata[6] s_axi_rdata[7] s_axi_rdata[8]
s_axi_rdata[15] s_axi_rdata[11] s_axi_rdata[12] s_axi_rdata[13] s_axi_rdata[16] s_axi_rdata[17] s_axi_rdata[18] s_axi_rdata[14] s_axi_rdata[20] s_axi_rdata[22] s_axi_rdata[27]
s_axi_rdata[23] s_axi_rdata[19] s_axi_rdata[25] s_axi_rdata[26] s_axi_arready s_axi_rdata[28] s_axi_rdata[24] s_axi_rdata[30] s_axi_rdata[31] s_axi_araddr[2] s_axi_arvalid s_axi_rdata[29]
s_axi_araddr[7] s_axi_araddr[3] s_axi_araddr[4] s_axi_araddr[5] s_axi_araddr[6] s_axi_bvalid s_axi_araddr[9] s_axi_araddr[10] s_axi_bready s_axi_wstrb[0] s_axi_wstrb[1] s_axi_araddr[8]
s_axi_bresp[1] s_axi_wready s_axi_wvalid s_axi_wdata[0] s_axi_bresp[0] s_axi_wstrb[2] s_axi_araddr[11] s_axi_wdata[4] s_axi_wdata[1] s_axi_wstrb[3] s_axi_wdata[2] s_axi_wdata[3]
s_axi_wdata[9] s_axi_wdata[5] s_axi_wdata[6] s_axi_wdata[7] s_axi_wdata[8] s_axi_wdata[14] s_axi_wdata[10] s_axi_wdata[11] s_axi_wdata[12] s_axi_wdata[13] s_axi_wdata[19] s_axi_wdata[15]
s_axi_wdata[16] s_axi_wdata[17] s_axi_wdata[18] s_axi_wdata[24] s_axi_wdata[20] s_axi_wdata[28] s_axi_wdata[26] s_axi_wdata[27] s_axi_awready s_axi_awvalid txp[4]
tx_start_of_multiframe[1] txp[1] tx_start_of_frame[0] txp[2] txn[1] txn[3] txn[2] s_axi_awaddr[11] txn[0] txp[0] s_axi_awaddr[2] s_axi_awaddr[7] s_axi_awaddr[7] s_axi_awaddr[7] s_axi_awaddr[8]
s_axi_awaddr[9] s_axi_awaddr[2] txn[4] s_axi_awaddr[5] s_axi_awaddr[6] s_axi_awaddr[3] s_axi_awaddr[4] }] -strings { "*" } -strings { "*" } -timestamp "Wed Mar 14 22:57:14 GMT 2018" ;#1

7. To delete the DRC waiver and rewrite the waiver using wildcards to target a subset of the ports objects, enter:

```
delete_waivers [get_waivers -type drc]
create_waiver -type DRC -id {UCIO-1} -user "Xilinx" -desc "Waive
selected UCIO violations" -objects [get_ports { s_axi_rdata[*]
s_axi_wdata[*] s_axi_araddr[*] } ] -strings { "*" } -strings { "*" }
```

Note: This command only covers a subset of the original 125 objects.

- 8. Select **Reports** \rightarrow **Report DRC** to rerun Report DRC.
- 9. In the Report DRC dialog box, select **Display only waived violations** to report only waived violations, and click **OK**.



🝌 Report DRC			×
Check design against sele	ected rule decks and/or individua	I design rules.	4
<u>R</u> esults name: <u>Interactive report file:</u> <u>Export to file:</u>	drc_2		©
Waivers Apply waivers Apply waivers Display only v Ignore all waivers Rule Decks Vivado Rule Deck	3	Rules (2120 of 2136)	
default opt_check opt_check opt_check opt_check of opt_check opt_check	ecks ecks _checks _checks	 All Rules (2136) Netlist (737) Pin Planning (110) Clocking (4) Memory (148) Floorplan (11) Implementation (239) Physical Configuration (842) 	Ŷ
 ✓ Open in a new tab ? 		ОК	Cancel

In the DRC Report, verify that only 68 violations are waived out of 125.



IMPORTANT! You cannot waive READONLY or NODISABLE violations. For example, if you enter:

create_waiver -type DRC -id RTSTAT-1 -description "Waive RTSTAT-1"



The Vivado tools issue the following error:

ERROR: [Vivado_Tcl 4-934] Waiver ID 'RTSTAT-1' is READONLY or NODISABLE and cannot be waived. These Factory designations specify that a check is required and may not be overridden by user action.

Step 11: Generating a Summary Report for Waived Violations

This step covers how to use the report_waivers Tcl command to generate a summary report for CDC, DRC, and methodology waivers.



IMPORTANT! Before running the *report_waivers* command, you must rerun Report CDC, Report DRC, or Report Methodology to ensure that added or removed waivers are included in the statistics reported by *report_waivers*.

1. To rerun Report CDC, enter:

report_cdc

2. To rerun Report DRC, enter:

report_drc

Note: You do not need to rerun Report Methodology, because no methodology waivers were set.

3. To create a summary report, enter:

report_waivers

By default, report_waivers reports only waived violations. The following figure shows the UCIO-1, CDC-10, CDC-11, and CDC-14 rules, which have defined waivers.



Table Of		100000								
3. REPOR	rt deta Rt deta	ILS (DRC)	DOLOGY: no waive	rs)						
1. REPOR	RT SUMM	IARY								
Waiver 1	Туре Т	otal Vios	Remaining Vios		Used Waivers	Set Waivers				
DRC METHODOL			171 0	68 0	1 0	1 0				
CDC Note: Ti		157 ort is bas	944 ed on the most r	13 recent report_	4 drc/report_met	4 hodology/repor	t_odc runs.			
2. REPOR	RT DETA	ILS (DRC)								
Rule	Sever	ity	Description		Total Vios		Waived Vios			
Concernance of the			g Unconstrained				68	1	1	
4 REPOR	RT DETA	ILS (CDC)								
		ty Descri				Total Vios	Remaining Vios	Waived Vios	Used Waivers	Set Waivers
CDC-11	Critic	al Fan-ou	ational logic de t from launch fl bit CDC path on	lop to destina	tion clock	r 187 2 10	186 0 0	1 2 10	1 2 1	1 2 1
Note: Ar	ny 'Rul	e'which i	s flagged by ' * ' wber of wessages	is an aggreg		and its counts	are based on t	he number of	objects represe	nted,

Note the number of waived objects and total violations:

- The aggregating DRCs are reported as 1 violation per object inside the violation. Because there are 113 objects in NSTD-1, 125 objects in UCIO-1 plus 1 in RTDAT-13, a total number of 239 DRC violations are reported in the Summary table.
- The Report Summary table reports all of the violations.
- The Report Details tables only report the check IDs that have one or more waivers.
- 4. To generate detailed tables with all of the rules, including rules with no waivers, enter:

report_waivers -show_msgs_with_no_waivers

The following figure shows the report with all DRC and CDC rules reported in the Report Details.



Table Of (Contents												
1. REPORT 2. REPORT 3. REPORT	SUMMARY DETAILS (DRC)	DDOLOGY: no waive	rs)										
1. REPORT	SUMMARY												
Waiver Typ	pe Total Vios	Remaining Vios	Waived Vios	Us	ed Waivers	Set Wa	aivers						
DRC METHODOLOG CDC Note: This	957	171 0 944 sed on the most r	68 0 13 recent report.	1 0 4 .drc.	/report_met	1 0 4 hodolos	yy/report_c	dc runs.					
	DETAILS (DRC)												
Rule	Severity		1		Total Vios	Renat	ining Vios	Waived Vios	Used	d Waivers	Set	. Waivers	
NSTD-1*	Critical Warr	ning Unconstrain ning Unspecified ning Insufficier	I/O Standard			57 113 1		68 0 0	1 0 0		1 0 0		
4. REPORT	DETAILS (CDC)												
	everity Descri	iption					Total Vio	s Remaining	Vios	Waived V	lios	Used Waivers	Set Waiver
CDC-11 Cr CDC-14 Cr CDC-1 Cr CDC-3 In CDC-3 In CDC-4 Cr CDC-9 In CDC-13 Cr	ritical Fan-ou ritical Multi- ritical 1-bit nfo 1-bit ritical Multi- nfo Asynch ritical 1-bit	national logic de ut from launch fl bit CDC path on unknown CDC circ synchronized wit -bit unknown CDC roronus reset syr CDC path on a no enable controlle	op to destina a non-FD prim cuitry ch ASYNC_REG p circuitry chronized wit m-FD primitiv	ntio niti rop h R	n clock ve erty SYNC_REG pr		187 2 10 536 9 28 5 170 10	186 0 536 9 28 5 170 10		1 2 10 0 0 0 0 0 0 0		1 2 1 0 0 0 0 0 0 0 0	1 2 1 0 0 0 0 0 0 0 0
		is flagged by '*' mber of message:		ati	ng wessage	and it:	s counts are	e based on th	ne nu	wher of ob	ject	s represented,	

5. To run Report Methodology, enter:

report_methodology

6. To generate detailed tables with all of the rules, including rules with no waivers, enter:

report_waivers -show_msgs_with_no_waivers

The exact statistics are reported, as shown in the following figure.

Note: This figure does not include the Report Details (CDC) section.



Waiver Typ	e Total Vio	os Rem	aining Vios	Waived Vios	Used Wa	ivers	Set Waiver	s					
DRC METHODOLOG CDC Note: This	957	 171 157 944 pased o	n the most r	 68 0 13 ecent report_	1 0 4 drc/repo		1 0 4 hodology/re	- port_0	odo runs	•			
2. REPORT	DETAILS (DRO	:)											
Rule	Severity		Description	I.	Tota	l Vios	Remaining	Vios	Waived	Vios	Used Waive	rs S	et Waivers
UCIO-1* NSTD-1* RTSTAT-13	Critical Wa	arning		ed Logical Po II/O Standard t Routing			57 113 1		68 0 0		1 0 0	 0 0	
	DETAILS (ME		 GY)										
	Severity		ption		Total	Vios	Remaining	Vios	Waived '	∕ios	Used Waiver	s Se	t Waivers
LUTAR-1*			ives async r n CDC Logic	eset alert	40 1		40 1		0		0 0	0	

Step 12: Using Waiver Commands

In this step, you run additional commands related to the waivers.

1. To return a collection of CDC waiver objects, enter:

get_waivers -type cdc

The following CDC waivers are returned:

CDC-10#1 CDC-11#1 CDC-11#2 CDC-14#1

2. To filter the list of waivers to only return CDC-14 waivers, enter:

get_waivers -filter {ID == CDC-14}
CDC-14#1

3. To report all of the properties on a CDC waiver object, enter:

report_property [lindex [get_waivers -type cdc] end]

The following properties are returned:

Property	Туре	Read-only	Value
CLASS	string	true	cdc_waiver
DESCRIPTION	string	false	No more CDC-14!
ID	string	true	CDC-14
INDEX	string	true	1
IS_SCOPED	bool	true	0
NAME	string	true	CDC-14#1
OBJECT_COUNTS	string	true	pins:2
SCOPE	string	true	



TAGS	string	false	
TIME	string	true	<timestamp></timestamp>
TYPE	string	true	CDC
USED_CNT	string	true	10
USER	string	true	Xilinx

Note: You cannot retrieve the design objects attached to a waiver object.

4. To delete all of the previously created CDC-14 waivers, enter:

```
delete_waivers [get_waivers -filter {ID == CDC-14}]
```

Note: After a waiver object is deleted, the waiver no longer applies and the violations that it waived are reported again.

5. To delete all of the remaining CDC waivers, enter:

```
delete_waivers [get_waivers -type cdc]
```

Summary

In this lab, you accomplished the following:

- Waived CDC and DRC violations
- Generated reports for waived violations
- Exported waivers
- Used waiver commands



Lab 2

Using Report QoR Suggestions

Introduction

The report_qor_suggestions (RQS) command enables the Vivado[®] Design Suite tools to analyze a design and provide automated solutions for enhancing QoR. The command can be run on an open design after synthesis or after any stage in the implementation flow. RQS evaluates the design in five key areas and suggests fixes or improvements in these areas. The five areas are utilization, clocking, constraints, congestion, and timing. Recommendations from RQS can take the following forms:

- RQS objects. These can add:
 - 。 Switches to a given command
 - Properties to a given design object
 - Implementation strategies customized for the design using machine learning algorithms
- Text recommendations that require intervention by the user.

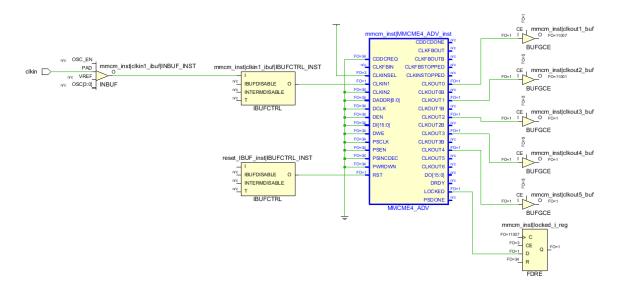
This tutorial will cover how to work with the RQS objects contained within RQS files in a project based environment. Non project flow steps are also covered but not explicitly run.

Step 1: Understanding the Design

This lab uses a pre-built design to demonstrate some of the features of RQS. Suggestions are triggered by the design of the RTL and the placement of blocks using floorplanning. The pre-built design contains the following modules:

• Clocking Module: The main clocking circuit for the design resides in clocking_module.vhd. For simplicity, RST is tied to GND. LOCKED is registered and tied to an output port. The structure of this block is shown in the following figure.

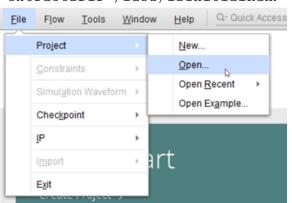




- **Reg CLKA to CLKB Module:** This module contains a synchronous CDC for a large bus. It registers input data using CLKA and then passes it to a register on the CLKB domain to be passed to the output. Registering large buses on different related clock domains can impact hold slack (WHS/THS) and setup slack (WNS/TNS).
- **Bit Expander and Bit Reducer Modules:** These modules enable the expansion and contraction of internal data widths so that the design does not run out of I/Os. The modules take an arbitrary data width and expand or contract it to or from a desired size. The expansion and contraction logic creates many logic levels and should be untimed. When untimed, they are ignored by report_qor_suggestions.

The following steps cover opening the project and examining the placement of the floorplanned modules.

In the Vivado Design Suite, go to File → Project → Open and select the project located in <extract_Dir>/lab2/1_InitialRun.



- 2. In the Flow Navigator, click **Open Synthesized Design**. When prompted, select **Synthesis Settings** to load the design with the Original Constraints.
- 3. In the Netlist view, look at the hierarchy.



N top
> 🚍 Nets (18)
> 🚍 Leaf Cells (11)
✓ ✓ clk300_to_clk600_ffs_i (reg_clka_to_clkb)
> 🗁 Nets (33008)
> 🗁 Leaf Cells (22004)
> 🚺 bit_expander_i (bit_expander)
> I bit_reducer_i (bit_reducer)
> 🚺 mmcm_inst (clocking_module)

4. In Device view, look at the pblock. This has been added to help trigger suggestions on the design without requiring a highly utilized design.

Device							? _ D @ X
← →	€. (9. S	[b] () HI	Po 🗉		\$
	ł						SLR1
	X0Y9	X1Y9	K2Y9	X3Y9	<u></u>	X5Y9	
	X0Y8	X1Y8	X2Y8	ХЗҮВ	X4Y8	X5Y8	
							:
	<u>X0Y7</u>	<u>X1Y7</u>	<u></u> <u>K2Y7</u>	<u>X3Y7</u>	<u> </u>	X5Y7	
	X0Y6	X1Y6	X2Y6	X3Y6	<u>X4Y6</u>	X5Y6	
	 x0Y5	V4) (F	-	V-31 (F	1	x5Y5	
	1	X1Y5	K2Y5	X3Y5	K4Y5	X315	SLRO
	1 X0Y4		-2 <mark>K</mark> 2Y4	ХЗҮ4	<u>×4Y4</u>	X5Y4	
	ł						
	X0Y3	<u>x1Y3</u>	<u>x2Y3</u>	<u>X3Y3</u>	<u></u>	<u>x5</u> Y3	
	X0Y2	X1Y2	<u>x2Y2</u>	ХЗҮ2	<u>x4Y2</u>	X5Y2	
	i xoyi	X1Y1	1 X2Y1		K4Y1		i i
	XOYO	X1Y0	K2Y0	X3Y0	X4Y0	X5Y0	





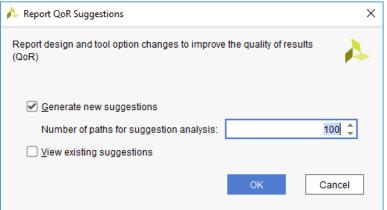
0

TIP: When a block or blocks are selected, you can investigate the design further by pressing F4 to open the schematic tools.

Step 2: Running Report QoR Suggestions

This step covers running the <code>report_qor_suggestions</code> command to generate a report. The command can be run on an open design at any stage of the implementation flow after synthesis. In project mode, this is typically after synthesis or implementation. In non-project mode, this can be after <code>synth_design,link_design,opt_design,place_design,phys_opt_design, or route_design.</code>

1. In the Vivado IDE, from the pull down menus, click **Reports** → **Report QoR Suggestions...** to bring up the dialog box shown in the following figure.



The equivalent Tcl command is:

report_qor_suggestions -quiet -name qor_suggestions_1

The command will:

- Examine the design and generate new suggestions
- Generate a report on the suggestions

The report opens automatically in the integrated design environment (IDE). Due to the interactive nature of the report, only one instance of the report can be open at any time.

Note: By default, the RQS command reports on the 100 worst failing paths per clock group. You can change the number of paths that RQS uses for the analysis of timing-critical paths by modifying the - max_paths switch. Increasing this number generates more suggestions, but on paths that are reducing in criticality. This may help in the later stages of design closure once all the key items are resolved.



Step 3: Understanding the Report

This step explains the different sections of the generated QoR Suggestions report. On the left of the report window, you can navigate to the different sections of the report; on the right, more information is provided.

1. In the generated report, under RQS Summary, select **GENERATED**. This brings up the report section shown in the following figure.

QoR Suggestions						×				
Q ≚ 0 + [≥]	: Q ≚ 0 0 GEN	ERAT	ΈD							
General	ID	\checkmark	GENERATED_AT	APPLICABLE_FOR	AUTOMATIC	DESCRIPTION				
 ~ RQS Summary GENERATED ~ XDC RQS_XDC-1-1 ~ Utilization ~ RQS_UTIL-3-1 ~ Timing ~ RQS_TIMING-33-1 ~ RQS_TIMING-54_1-1 ~ Clocking ~ RQS_CLOCK-15-1 	~ ⊖ xDC	•								
	RQS_XDC-1-1		none	synth_design	No	Tight Constraints for given paths. Review critical paths with difficult requiren				
		\checkmark								
	RQS_UTIL-3-1	\checkmark	none	opt_design	No	High utilization of certain types of cells in a PBlock. Try to reduce the utilizat				
		\checkmark								
	RQS_TIMING-33-1	\checkmark	none	synth_design	Yes	Retiming of Flops in Critical paths can improve the timing.				
	RQS_TIMING-54_1-1	\checkmark	none	opt_design	No	Found high fanout nets that are driven by flops and driving control signals.				
		\checkmark								
	RQS_CLOCK-15-1		none	opt_design	No	High THS due to synchronous CDCs. Try to reduce the number of timed path				
	K. Strategies are available only in default/explore at successfully routed design.									
				ort Suggestions		ions to Project				

The GENERATED section provides a list of all the suggestions that have just been generated at this stage of the current run. Each suggestion has a description that details the reason for the suggestion. Additionally, for each suggestion the following information is provided.

Item	Description	Comment					
GENERATED_AT	This shows what stage of the design the suggestion was generated at. Typical values <pre>place_design</pre> or <pre>route_design</pre>	As you progress through the design stages, the decisions that the tool makes are based on the information available at the time. Additionally, information accuracy increases after placement and again after routing. Some early suggestions may not be required once you have run through the flow.					
		Some early suggestions may be required to solve issues later in the flow.					
APPLICABLE_FOR	This shows what stage must be rerun in order for the suggestion to take effect.	Most suggestions are executed at either <code>synth_design</code> Or <code>opt_design</code>					
SOURCE	Details where the suggestion was generated	current run or if from a previous run, a file name that contained the suggestion.					
AUTOMATIC	Details whether the suggestion is executed automatically or the user must manually intervene	Automatic suggestions will either recommend a switch the tool or a property to be added to a cell or net					

Under the other sections of the report there are details about the individual suggestions that have been generated.

2. Click on the **RQS_XDC_1_1** hyperlink. This will take you to the details section for this suggestion.



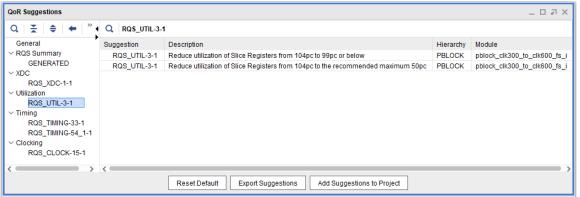
QoR Suggestions												
Q ¥ ♦ ← *	Q RQS_XDC-1-	1										
General < RQS Summary GENERATED XDC XDC VDC RQS_XDC-1-1 VUllization RQS_UTIL-3-1 Timing RQS_TIMING-33-1 Clocking RQS_CLOCK-15-1	Suggestion RQS_XDC-1-1	No of Paths 2	Logic Levels 7	Routes 8	Slack -0.908	Req. 1.667	Skew 0.145	Datapath Delay 2.414	Cell% 50.20	Route% 49.80	Source Clock clk_600_clk_wiz_0	Destination Cloudk_600_clk_wia
<>	<	Res	set Default	Export Su	ggestions	Ado	d Suggest	tions to Project				:

The suggestion description says that the timing constraint is too tight for the given path(s).

The path has a large negative slack which would stand out in a timing report. Timing paths use net delays that are optimal, this gives the tools the correct order to place and route them. Close analysis shows this is a 600 MHz path with seven logic levels. This is a path that will need to be fixed. It is not possible to fix every path automatically. For this tutorial, assume that a false path constraint has been missed. Right-click on the path and select **set false path** \rightarrow **startpoint to endpoint**. Alternatively, enter the following in the Tcl console to add this.

```
set_false_path -to [get_cells clk300_to_clk600_ffs_i/bit_reducer_i/
tmp_r_reg]
```

- 3. Click on the back arrow button * to go back to the GENERATED view.
- 4. In the GENERATED view, click on **RQS_UTIL-3-1**. These suggestions examine utilization of different primitive types within a pblock.



RQS examines the utilization of the full design, pblocks and SLRs. In addition, it also looks at control sets and compares design numbers against thresholds from a characterized model that may move thresholds depending on the design. These thresholds are not hard limits of the device but rather a guidance threshold that has been shown to affect timing performance.



For this case, there is high register usage in the clk300_to_clk600_ffs_i pblock. RQS provides a general text recommendation to reduce the utilization of this primitive type. Sometimes it will provide automatic suggestions too. It is also worth noting that opt_design has not been run yet. This could further reduce utilization but this is not certain until you have run the command. RQS models do not change between synth_design and opt_design. Another way to resolve this is to increase the size of the pblocks.

 Click Window → Physical Constraints. In the Physical Constraints window, select the clk300_to_clk600_ffs_i pblock and view the Statistics tab in the Pblock Properties window. This will display the utilization of the pblock. You can see that the CLB utilization is 104.19%.

Pblock Properties $? = \Box \not a \times$						
pblock_clk300_to_clk6	■ pblock_clk300_to_clk600_fs_i ← 😝 🔅					
Physical Resource Estimates						
Site Type	Available	Used	% Util			
CLB LUTs	10560	2982	28.24			
LUT as Logic	10560	2982	28.24			
LUT as Memory	5280	0	0.00			
CLB Registers	21120	22004	104.19			
Register as Flip Flop	21120	22004	104.19			
Register as Latch	21120	0	0.00			
CARRY8	1320	0	0.00			
F7 Muxes	5280	0	0.00			
F8 Muxes	2640	0	0.00			
F9 Muxes	1320	0	0.00			
General Properties	Statisti	cs Cel	ls C≀ ⊲	▶ ≡		

6. To resolve this issue, move the pblock to a larger clock region. Enter the following at the TCL console.

```
resize_pblock pblock_clk300_to_clk600_fs_i \
-add {SLICE_X0Y0:SLICE_X34Y59} -remove \
{CLOCKREGION_X1Y4:CLOCKREGION_X1Y4} \
-locs keep_all
```

The updated pblock utilization is now 79.04%. This is still above the threshold but as this is a constructed design, it holds performance better than a more typical design at this utilization.

- 7. Click on the back arrow button * to go back to the GENERATED view.
- 8. Examine the TIMING-54-1 suggestion. This suggestion has found some high fanout nets that may be better to move to global routing. At this stage it is informative to the user only. As <code>opt_design</code> does this automatically, RQS will make this recommendation only if it can improve congestion or timing. It will wait until after <code>place_design</code> before making this assessment.



Qolt Suggestions					_ D 7 X
Q	1 Q 2 0 0 GEN	RATED			
General	10	GENERATED,AT	APPLICABLE, FOR	AUTOMATIC	DISCRIPTION
 RQS Summary 	>DXDC	8			
GENERATED	9 RQS,XDC-1-1	synth_design	synth_design	No	Tight Constraints for given paths. Review critical paths with difficult requirements and either reduce logic de
~ XDC	~ D Utilization	8			
RQ5_X0C-1-1	9 RQS,UTL-3-1	synth_design	opt_design	No	High utilization of certain types of cells in a PBlock. Try to reduce the utilization of that particular type of cell
 Utilization 	∽ ⊇ Timing	2			
RQ5_UTIL-3-1	RQ5_TIMING-33-1	synth, design	synth_design	Yes	Retiming of Flops in Critical paths can improve the timing.
- Timing	RQS_TIMING-54_1-1	🕑 synth_design	opt_design	No	Found high fanout nets that are driven by flops and driving control signals. Insert buffers to use global routin
RQ5_TIMING-33-1	~ E Clocking	8			
RQS_TIMING-54_1-1	P RQS_CLOCK-15-1	Synth_design	opt_design	No	High THS due to synchronous CDCs. Try to reduce the number of timed paths, the uncertainty and the clock
 Clocking RQS.CLOCK-15-1 	(
Php_0006-15-1	ML Strategies are available only	in default/explore at s	uccessfully routed de	nsign.	
		Reset Defa	sult Export Sug	gestions /	Add Suggestions to Project

Clicking through to the details sections for this suggestion shows that the signal has a fanout of 11000 as shown in the following figure:

Q	Q RQS_TIMING-54_1	-1												
General	Suggestion	No of Paths	Logic Levels	Routes	Slack	Req.	Skew	Fanout	Datapath Delay	Cell%	Route%	Source Clock	Destination Clock	Startpoint
RQS Summary	RQS_TIMING-54_1-1	1	0	1	0.862	1.667	-0.172	11002	0.491	23.00	77.00	clk_600_clk_wiz_0	clk_300_clk_wiz_0	data_in_i_reg[0]/C
GENERATED	RQS_TIMING-54_1-1	1	0	1	2.513	3.333	-0.145	11000	0.530	21.30	78.70	clk_300_clk_wiz_0	clk_300_clk_wiz_0	clk300_to_clk600_
~ XDC	RQS_TIMING-54_1-1	1	0	1	0.852	1.667	-0.145	11000	0.530	21.30	78.70	clk_600_clk_wiz_0	clk_600_clk_wiz_0	clk300_to_clk600_
RQS_UTIL-3-1 ~ Timing RQS TIMING-33-1														

- 9. There are no AUTOMATIC suggestions to export at this stage for this design. Close the design and discard changes if you are prompted to save the design. In the Flow Navigator, click **Open Implemented Design**. If prompted regarding the design being out-of-date, ignore the message and continue to open the design. This is a design that has been run though with the updated timing and pblock constraints you just applied.
- 10. When the design is open, click **Reports** \rightarrow **Report QoR Suggestions...**.
- 11. Examine the suggestions. There are now two clocking suggestions. Clocking suggestions can lead to large leaps in QoR so should be examined carefully.

QoR Suggestions								
$Q = X = \varphi$ \Rightarrow $Q = X = \varphi$ (generated)								
General	ID	GENERATED	AT APPLICABLE_FOR	AUTOMATIC	DESCRIPTION			
~ RQS Summary		✓						
GENERATED	RQS_UTIL-3-2	route_design	opt_design	No	High utilization of certain types of cells in a PBlock. Try to reduce the utilization of that particular type of cel			
 Utilization 	✓ □ Clocking	 Image: A start of the start of						
RQS_UTIL-3-2	RQS_CLOCK-1-1	route_design	opt_design	Yes	Critical paths have high clock skew due to sub-optimal clock roots. Apply CLOCK_DELAY_GROUP prope			
~ Clocking	RQS_CLOCK-2-1	route_design	opt_design	No	Sub-optimal clock network topology that can increase clock skew. Use BUFGCE_DIVs instead of BUFGC			
RQS_CLOCK-1-1 RQS_CLOCK-2-1	✓							
Strategy	RQS_STRAT-64-1	route_design	none	No	ML Strategy suggestion to improve timing			
RQS_STRAT-64-1	RQS_STRAT-22-1	route_design	none	No	ML Strategy suggestion to improve timing			
RQS_STRAT-22-1	RQS_STRAT-44-1	route_design	none	No	ML Strategy suggestion to improve timing			
RQS_STRAT-44-1	<				>			
	Reset Default Export Suggestions Add Suggestions to Project							

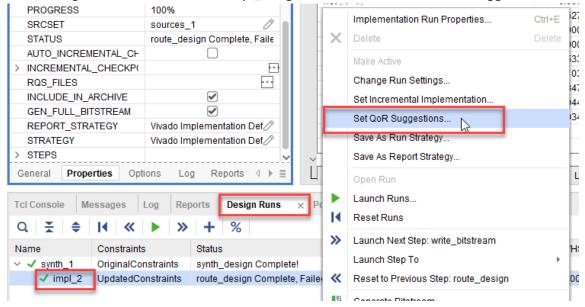
RQS_CLOCK-1-1 is recommending CLOCK_DELAY_GROUP and is an AUTOMATIC suggestion. In addition there are strategy suggestions that will be covered later in this lab. You can explore the timing paths by navigating to the path.



RQS_CLOCK-2-1 is a manual (AUTOMATIC=No) suggestion recommending that buffers are swapped from BUFGCE to BUFGCE_DIV. Double-click the path to open up the timing report. In the timing report click on the **Clock Uncertainty** hyperlink. You can see that there is a Phase Error (PE) element that is shown. Phase Error gets added when the source and destination clocks are from different MMCM output pins. The suggestion requests you to use the same MMCM output pin, connect them to BUFGCE_DIVs with different divisors. This modification requires a manual RTL modification.

Clock Uncertainty Equation $\qquad imes$					
((TSJ^2 + DJ^2)^1/2) / 2 + PE					
Total System Jitter (TSJ)	0.071ns				
Discrete Jitter (DJ)	0.110ns				
Phase Error (PE) 0.120ns					

- 12. Export the AUTOMATIC suggestion RQS_CLOCK-1-1. Firstly, ensure that RQS_CLOCK-1-1 is selected. Other suggestions can also be selected. Suggestions that are not AUTOMATIC are ignored and Strategy suggestions cannot be selected. Click **Add Suggestions to Project** and select a suitable directory.
- 13. In the Source window, expand **Utility Sources** and locate the rqs_report.rqs file that has been added to the project.
- 14. In the Design Runs window, select Impl_2, right-click and select Set QoR Suggestions.



15. Select the **rqs_report.rqs** file that was added to the project.



🝌 Set QoR Suggestions		×
Apply QoR Suggestions to the c suggestions file or disable the generated from report_qor_sug	suggestions. Suggestions are 🛛 👔	
Specify suggestion file	rqs_report.rqs 🗸 🗸	
Copy sources into pro	oject	
O Disable suggestions		
	OK Cancel	

In the next step you will run the implementation and see the suggestion being applied.

Step 4: Run with Suggestions

You will now examine what happens when a suggestion is applied and how it is reported. Then you will add further suggestions to the RQS file.

- 1. Close the open project and open the next project, 2_Updated_RTL_Files.xpr.
- 2. In the Flow Navigator, click Open Implemented Design.
- 3. While the run is opening, take the opportunity to examine the log file for the implementation run. In the Reports tab, select **impl_1_route_implementation_log_0** to open up the implementation log file.

In the log file, locate the following.

```
1. Read QOR Suggestions Summary
Read QOR Suggestions Summary
Suggestion Summary | Incr Friendly | Total |
1
      3 |
3 |
| Total Number of Suggestions
                                                   0 |
                                       ENABLED
                                                  0 |
   APPLICABLE_AT
                                                            synth_design
                                                   0 |
                                                         0 1
                                                   0 |
                                                         3 |
    opt_design
                                                         0 |
     That overlap with synthesis suggestions |
                                                   0 |
                                                          0 |
                                                   0 |
    place_design
    phys_opt_design
                                                   0 |
                                                          0 |
                                                   0 |
                                                          0 |
    route_design
   NOT ENABLED
                                                   0 |
                                                          0 |
      _ _ _ _ _ _ _ _ _ _ _ _
                                     - - - + - - -
INFO: [Vivado_Tcl 4-1103] Successfully read QoR suggestions file: <dir>/
1_InitialRun.srcs/utils_1/imports/impl_2/rqs_report.rqs.
```



This table reports a summary of what suggestions will be run at what stage of the flow and the message confirms that the file was read successfully.

Because the RQS file is binary, it cannot be read in a text editor. Therefore, to get more details a full report can be generated by running the following command at any time after the suggestion file has been read. (in the project mode this will require a Pre/Post TCL hook to be used):

report_qor_suggestions -of_objects [get_qor_suggestions]

The command will report all the suggestions that are in memory. As you are running this before any calls to <code>report_qor_suggestions</code> that generate new suggestions (RQS generates new suggestions when called without <code>-of_objects</code> switch), it will only show suggestions from the file. When reporting on existing suggestions, the result is almost instantaneous.

Finally, search for the following

```
INFO: [Vivado_Tcl 4-1067] Applying enabled auto TCL RQS suggestion for opt_design: RQS_CLOCK-1-1
```

It is at this point where the suggestion is executed and the CLOCK_DELAY_GROUP constraint is applied.

4. With the Implemented Design now open, click **Reports** → **Report QoR Suggestions...**. This time ensure the **View existing suggestions** box is checked as shown in the following figure.

A Report QoR Suggestions	×
Report design and tool option changes to improve (QoR)	the quality of results
✓ Generate new suggestions	
Number of paths for suggestion analysis:	100 🌲
✓ <u>V</u> iew existing suggestions	
	OK Cancel

5. In the report, under RQS Summary select **APPLIED**. This shows all the AUTOMATIC suggestions that have been applied. Here you can see that RQS_CLOCK-1-1 is applied.

QoR Suggestions							_ L 2 X
Q ≚ ≑ ←	» •	Q ₹ ⊕ APPLIE)				
General	2	ID	APPLICABLE_FOR	\checkmark	GENERATED_AT	AUTOMATIC	SOURCE
V RQS Summary		Clocking					
GENERATED		RQS_CLOCK-1-1	opt_design	1	route_design	Yes	C/Data/TUTORIAL/2019.2/UG938/work/2_Updated_RTL_Files/2_Updated
APPLIED							
< DOO 1111 9.4	ř	<					\$
, , ,		[Reset Default	Expo	t Suggestions	Add Suggestion	ns to Project

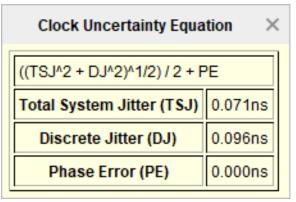
Note that the source of this suggestion is not the current_run.



6. Click **EXISTING**. This shows suggestions that were in the RQS file that have not been APPLIED. When you click in, you can see that this is the manual BUFGCE_DIV suggestion. As it can never be applied, it will remain in the EXISTING bucket. You cannot click through to the timing report as this is captured from the previous run and may no longer be valid. You can run timing analysis and look at the phase error of the net clocking scheme.

```
report_timing -to [get_clocks clk_300_clk_wiz_0] -to [get_clocks \
clk_600_clk_wiz_0] -name 1
```

You can see that phase error has been reduced to zero when you click into **Clock Uncertainty** link in the generated timing report.



7. Click **GENERATED**. Here you can see the strategy suggestions. These are a special type of suggestion that use machine learning to generate an implementation strategy that is customized for the design.

Step 5: Running ML Strategies

- 1. Keeping the project open from the previous step, at the TCL console, type cd <extract_dir>/Lab2. Remember this directory as in the next step you will write some files to it.
- 2. Type write_qor_suggestions -strategy_dir ./ and navigate to this directory. You should see the following files:

impl_1Project_MLStrategyCreateRun1.tcl	16/01/2020 15:26	TCL File	2 KB
impl_1Project_MLStrategyCreateRun2.tcl	16/01/2020 15:26	TCL File	2 KB
impl_1Project_MLStrategyCreateRun3.tcl	16/01/2020 15:26	TCL File	2 KB
impl_1SuggestionFile1.rqs	16/01/2020 15:26	RQS File	4 KB
impl_1SuggestionFile2.rqs	16/01/2020 15:26	RQS File	4 KB
impl_1SuggestionFile3.rqs	16/01/2020 15:26	RQS File	4 KB
NonProject_MLStrategyCreateRun1.tcl	16/01/2020 15:26	TCL File	1 KB
NonProject_MLStrategyCreateRun2.tcl	16/01/2020 15:26	TCL File	1 KB
NonProject_MLStrategyCreateRun3.tcl	16/01/2020 15:26	TCL File	1 KB



There are three files for each strategy:

- a. The *Project*.tcl file can be sourced within the project it was created in. This will create a new run and reference the Implementation Strategy and Suggestions that are defined in the RQS file.
- b. In the *NonProject*.tcl file is an example of how this can be setup for implementation flows that do not use the project flow. This file does not contain all the items required to run a flow and is for demonstration purposes only. For example, you must integrate this with open_checkpoint and any reporting commands you wish to run. It gives an example of referencing the RQS file and setting the directives.
- c. In *each* RQS file, there are the all the normal suggestion objects and one strategy suggestion object. This file is common for both project and non project flows.
- 3. Source the TCL project files. Enter the following at the TCL console.

```
source ./impl_1Project_MLStrategyCreateRun1.tcl
source ./impl_1Project_MLStrategyCreateRun2.tcl
source ./impl_1Project_MLStrategyCreateRun3.tcl
```

4. Now examine the **Design Runs** window. You will see that three new runs have been created.

Design Runs ? _ D 🔊 ×							
$\mathbf{Q}_{\mathbf{A}} \mid \mathbf{\Xi}_{\mathbf{A}} \mid \mathbf{\Phi}_{\mathbf{A}} \mid \mathbf{W} \mid \mathbf{V}_{\mathbf{A}} \mid \mathbf{V}_{\mathbf$							
Name	Constraints	Status	Elapsed	Incremental			
✓ ✓ synth_1	constrs_1	synth_design Complete!	00:11:17	Off			
impl_1 (active)	constrs_1	route_design Complete, Failed Timing!	00:30:06	Off			
impl_1_ML_Strategy_1	constrs_1	Not started		Off			
impl_1_ML_Strategy_2	constrs_1	Not started		Off			
impl_1_ML_Strategy_3	constrs_1	Not started		Off			

5. Select one of the runs. Examine the **Implementation Run Properties**. Each directive has been set to RQS.



Implementation Run Propertie	s			? _ 🗆	l 🛛 🗙	<
impl_1_ML_Strategy_1			+	⇒ 6	0	ł
Incremental Implementation:	Not	set				î
<u>S</u> trategy:	10	Vivado In	nplementati	on 👻		
Description:	Def	ault setti	ngs for Imple	ementation.		l
✓Design Initialization (init_de	esign)					
tcl.pre					•••	
tcl.post					•••	
✓Opt Design (opt_design)						
is_enabled			✓			
tcl.pre					•••	
tcl.post					•••	
-verbose			\cap			
-directive*		RQS			~	
More Options					-	
YPower Opt Design (power_	opt_de	esign)				~
General Properties Opt	tions	Log	Reports	Messages	3	

6. Click on **Properties** and examine the RQS File property. Note that RQS file has been set up automatically for you.





Implementation Run Properties	? _ D @ X
impl_1_ML_Strategy_1	← → ☆
Q X ♦ H + =	A ↓ Z ↓
	IIIIpi_1_ML_Oullegy_1
NEEDS_REFRESH	
PARENT	synth_1
PART	xcvu5p-flva2104-2LV-e 🖉
PR_CONFIGURATION	Ø
PROGRESS	0%
SRCSET	sources_1
STATUS	Not started
AUTO_INCREMENTAL_CHECKF	
> INCREMENTAL CHECKPOINT	E1
RQS_FILES	C:/TMP/UG938/impl_1Suggestio
INCLUDE_IN_ARCHIVE	
GEN_FULL_BITSTREAM	
REPORT_STRATEGY	Vivado Implementation Default R 🖉
STRATEGY	Vivado Implementation Defaults 🧷
> STEPS	~
General Properties Options	Log Reports Messages

- 7. You can either launch these runs or open the project located in the directory
 <extract_dir>/lab2/project 3_ML_Strategies.(On Linux, the project path is
 <extract_dir>/Lab2/3_ML_Strategies/3_ML_Strategies.xpr).
- 8. Finally you can examine the results of the run with ML Strategies. Of the three runs, one has closed timing and two have improved overall timing.

Design Runs					? _ D @ X		×
Q 素 ♦ I4 ≪ ▶	- » +	%					
Name	Constraints	Status	Elapsed	Incremental	WNS	TNS	W
✓ ✓ synth_1	constrs_1	synth_design Complete!	00:11:17	Off			
✓ impl_1 (active)	constrs_1	route_design Complete, Failed Timing!	00:30:06	Off	-0.141	-80.53	0.
impl_1_ML_Strategy_1	constrs_1	route_design Complete, Failed Timing!	00:41:34	Off	-0.147	-47.83	0
impl_1_ML_Strategy_2	constrs_1	route_design Complete!	00:15:42	Off	0.221	0.000	0
impl_1_ML_Strategy_3	constrs_1	route_design Complete, Failed Timing!	00:34:35	Off	-0.094	-42.50	0
< ⊂							>



Summary

In this lab, you used RQS to conduct a complex analysis of a demonstration design. You firstly examined the reports that showed RQS provided recommendations to solve implementation problems, then generated an RQS file and added it to a project implementation run. The Vivado implementation tools executed these suggestions automatically for you. You subsequently performed further analysis and generated ML Strategy Suggestions, and after running more runs, ultimately achieving design closure.





Appendix A

Additional Resources and Legal Notices

Please Read: Important Legal Notices

The information disclosed to you hereunder (the "Materials") is provided solely for the selection and use of Xilinx products. To the maximum extent permitted by applicable law: (1) Materials are made available "AS IS" and with all faults, Xilinx hereby DISCLAIMS ALL WARRANTIES AND CONDITIONS, EXPRESS, IMPLIED, OR STATUTORY, INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, OR FITNESS FOR ANY PARTICULAR PURPOSE; and (2) Xilinx shall not be liable (whether in contract or tort, including negligence, or under any other theory of liability) for any loss or damage of any kind or nature related to, arising under, or in connection with, the Materials (including your use of the Materials), including for any direct, indirect, special, incidental, or consequential loss or damage (including loss of data, profits, goodwill, or any type of loss or damage suffered as a result of any action brought by a third party) even if such damage or loss was reasonably foreseeable or Xilinx had been advised of the possibility of the same. Xilinx assumes no obligation to correct any errors contained in the Materials or to notify you of updates to the Materials or to product specifications. You may not reproduce, modify, distribute, or publicly display the Materials without prior written consent. Certain products are subject to the terms and conditions of Xilinx's limited warranty, please refer to Xilinx's Terms of Sale which can be viewed at https:// www.xilinx.com/legal.htm#tos; IP cores may be subject to warranty and support terms contained in a license issued to you by Xilinx. Xilinx products are not designed or intended to be fail-safe or for use in any application requiring fail-safe performance; you assume sole risk and liability for use of Xilinx products in such critical applications, please refer to Xilinx's Terms of Sale which can be viewed at https://www.xilinx.com/legal.htm#tos.

AUTOMOTIVE APPLICATIONS DISCLAIMER

AUTOMOTIVE PRODUCTS (IDENTIFIED AS "XA" IN THE PART NUMBER) ARE NOT WARRANTED FOR USE IN THE DEPLOYMENT OF AIRBAGS OR FOR USE IN APPLICATIONS THAT AFFECT CONTROL OF A VEHICLE ("SAFETY APPLICATION") UNLESS THERE IS A SAFETY CONCEPT OR REDUNDANCY FEATURE CONSISTENT WITH THE ISO 26262 AUTOMOTIVE SAFETY STANDARD ("SAFETY DESIGN"). CUSTOMER SHALL, PRIOR TO USING



OR DISTRIBUTING ANY SYSTEMS THAT INCORPORATE PRODUCTS, THOROUGHLY TEST SUCH SYSTEMS FOR SAFETY PURPOSES. USE OF PRODUCTS IN A SAFETY APPLICATION WITHOUT A SAFETY DESIGN IS FULLY AT THE RISK OF CUSTOMER, SUBJECT ONLY TO APPLICABLE LAWS AND REGULATIONS GOVERNING LIMITATIONS ON PRODUCT LIABILITY.

Copyright

© Copyright 2012-2020 Xilinx, Inc. Xilinx, the Xilinx logo, Alveo, Artix, Kintex, Spartan, Versal, Virtex, Vivado, Zynq, and other designated brands included herein are trademarks of Xilinx in the United States and other countries. All other trademarks are the property of their respective owners.

