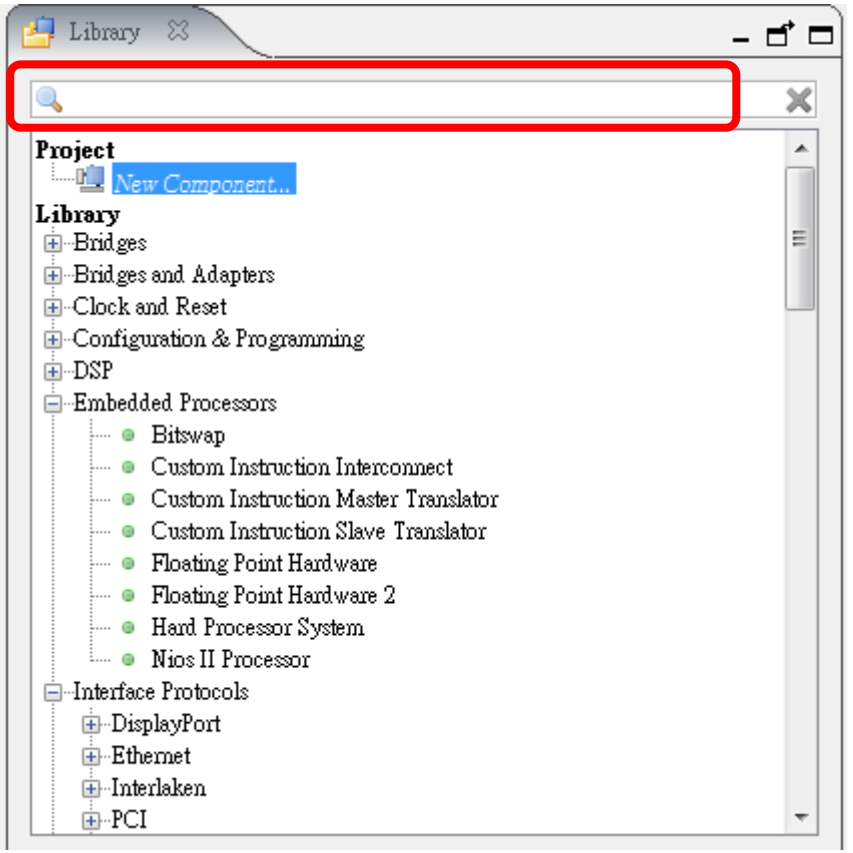
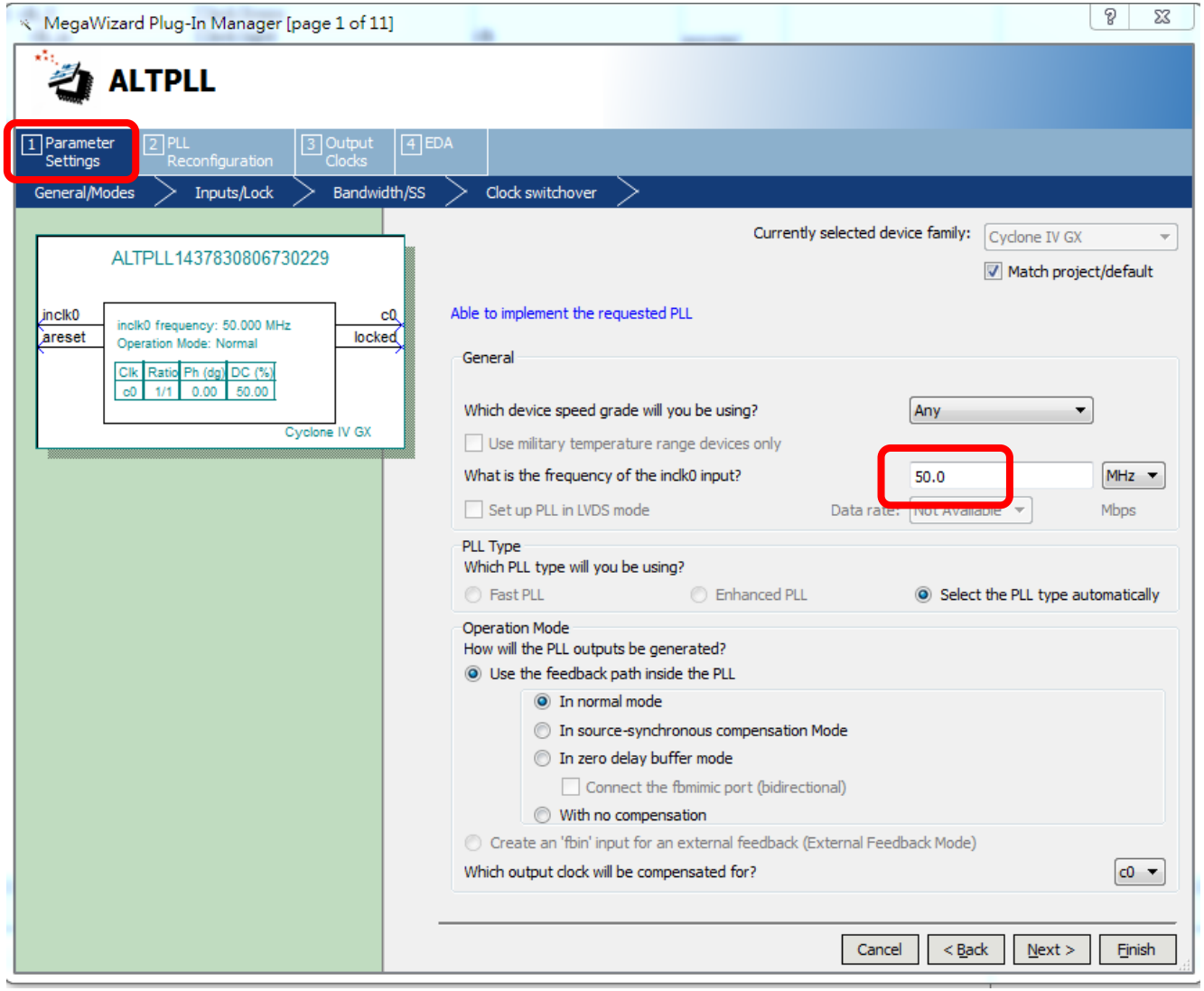
Lab 2 Qsys Tutorial

IPs in this lab

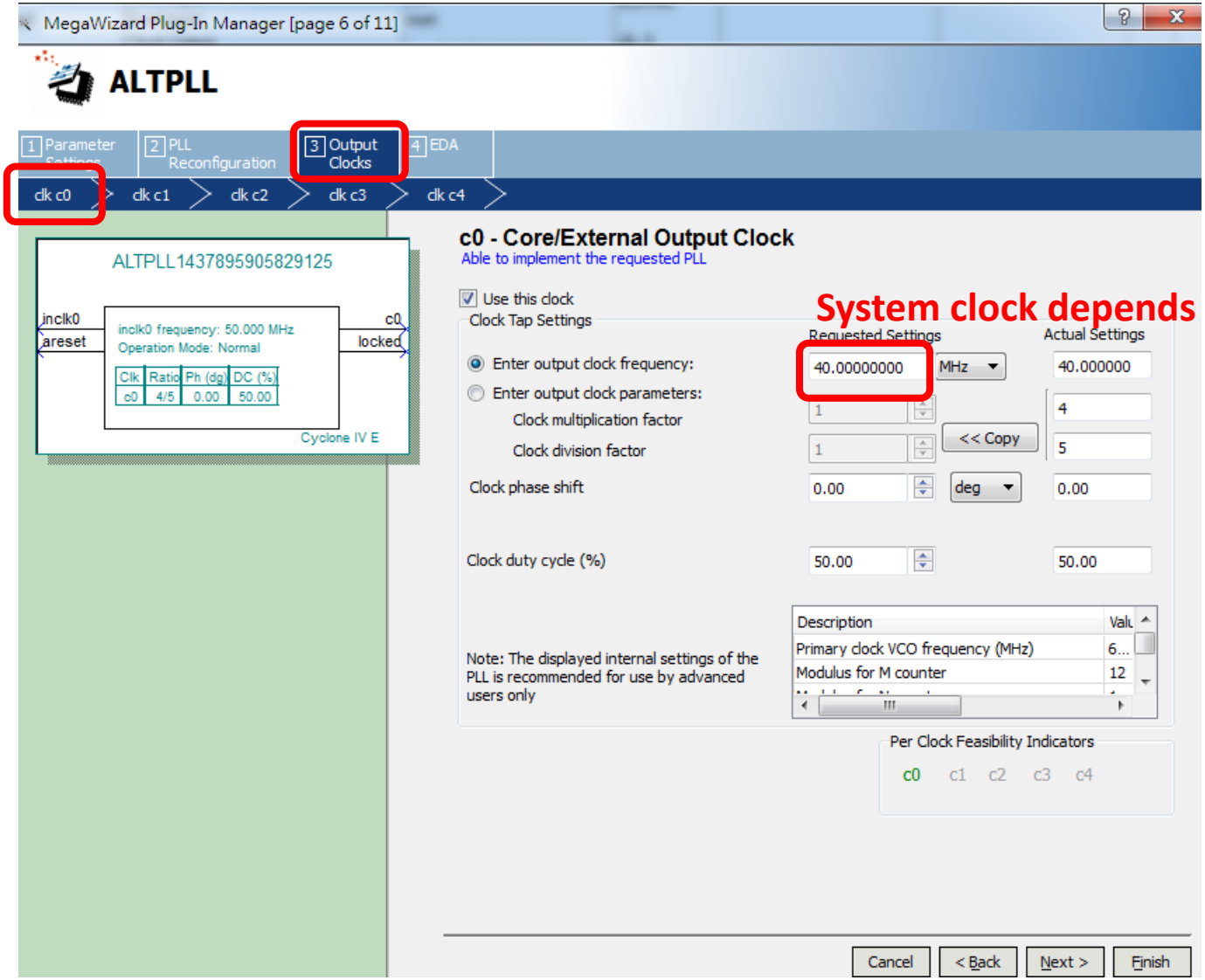


- Altera IPs
  - Avalon ALTPLL
  - UART (RS-232 Serial Port)
- Custom module
  - RSA Avalon-MM master interface module

IP configuration – Avalon ALTPLL

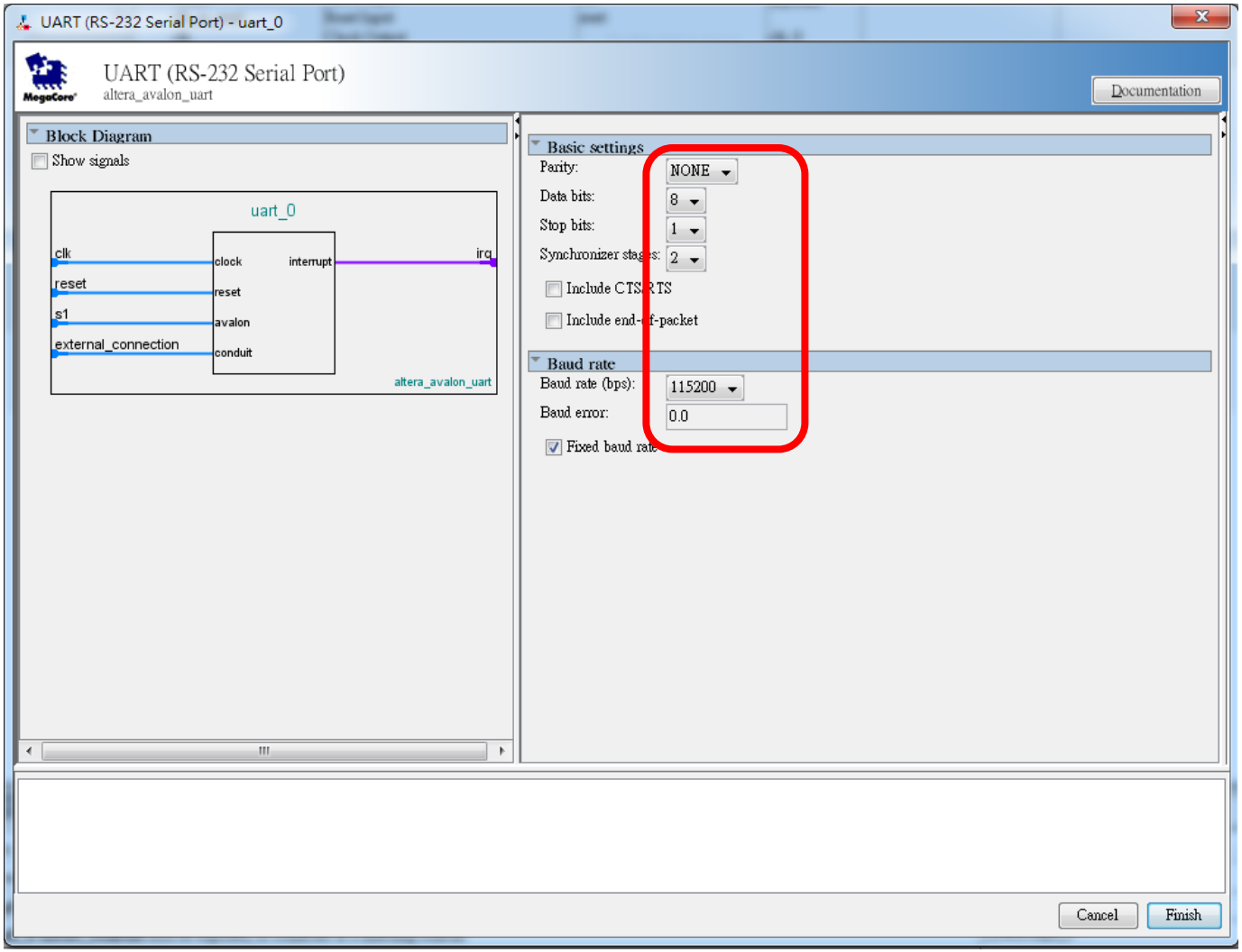


IP configuration – Avalon ALTPLL

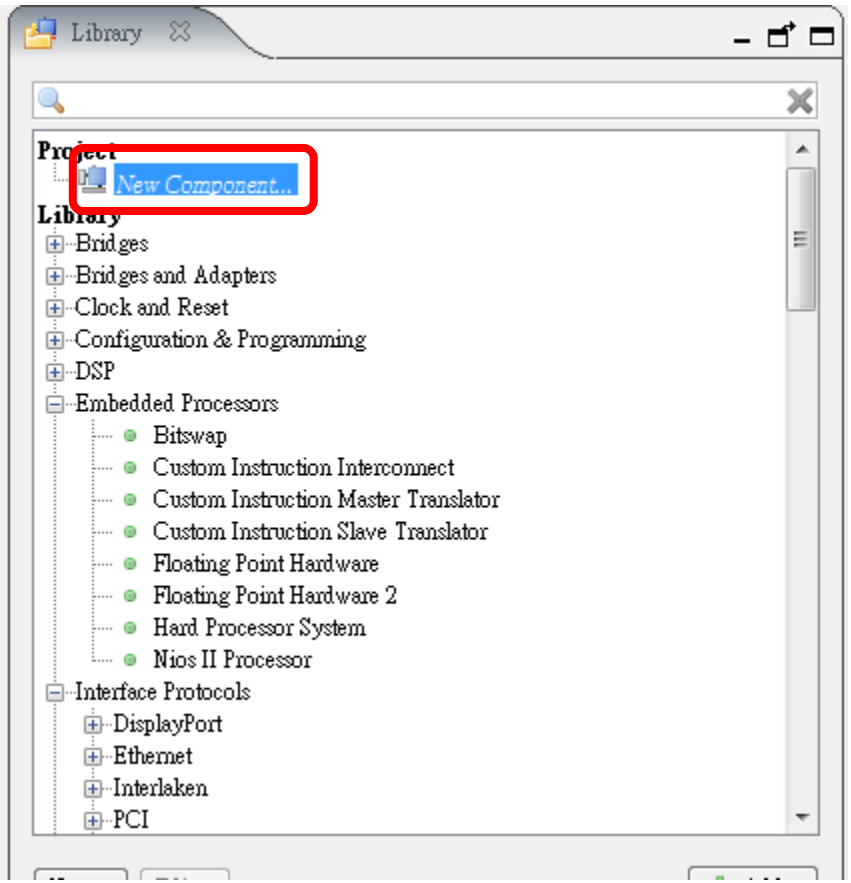


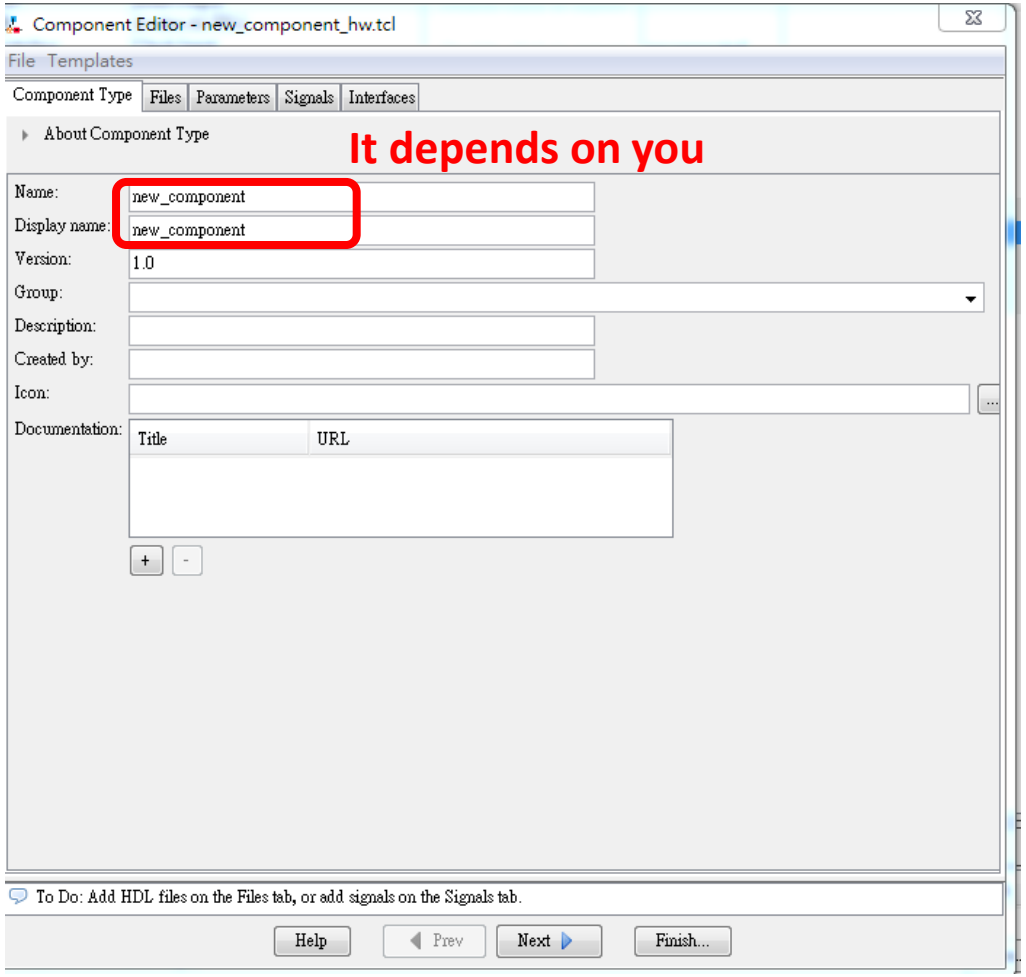
System clock depends on your design

IP configuration – UART (RS232)

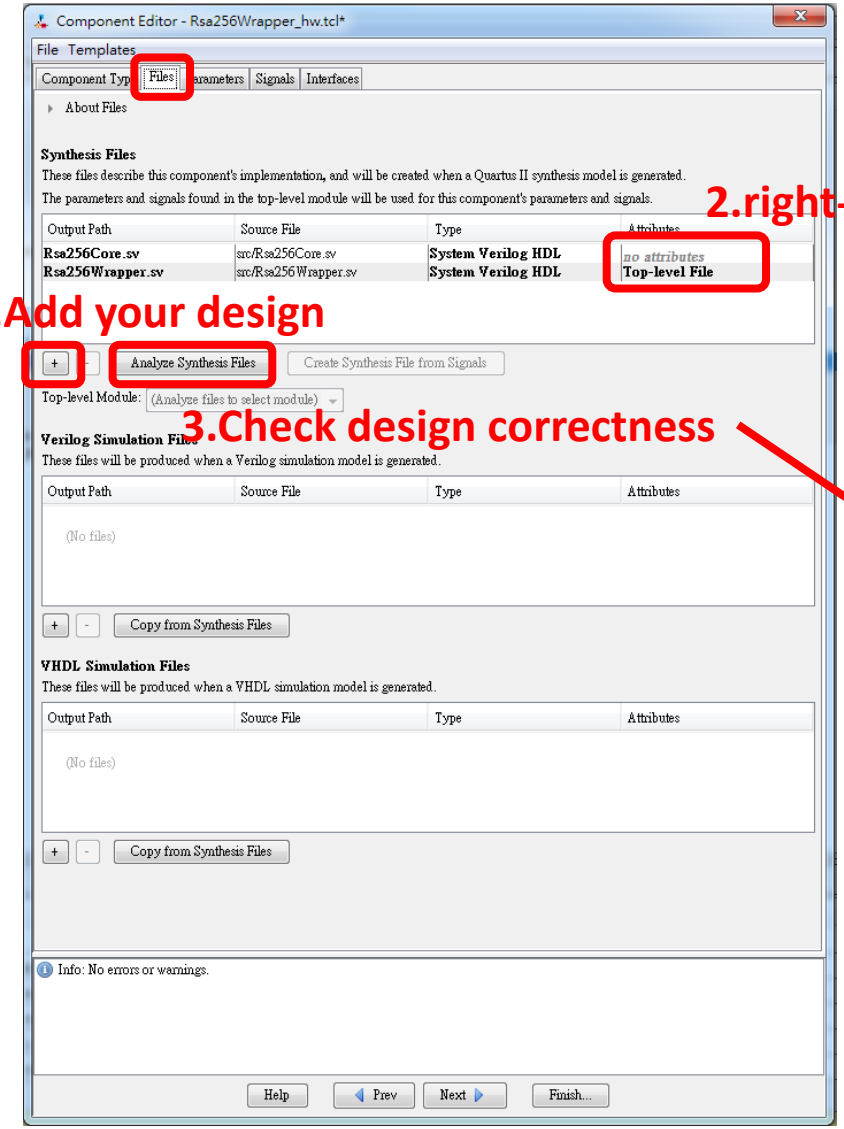


Add custom module to Qsys

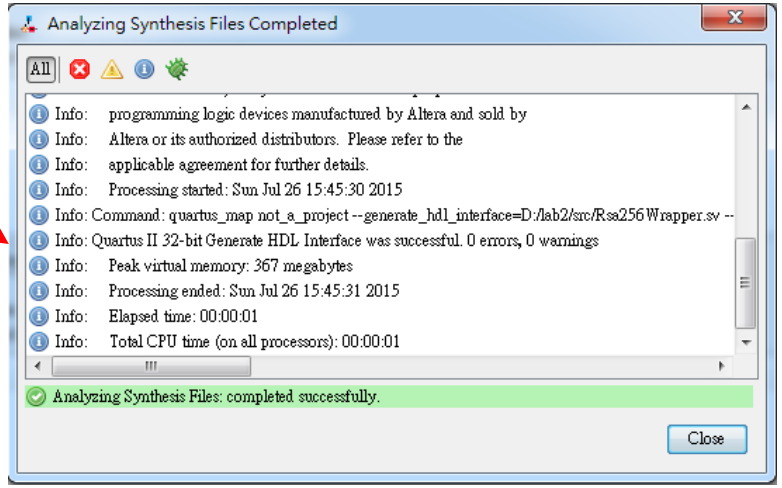




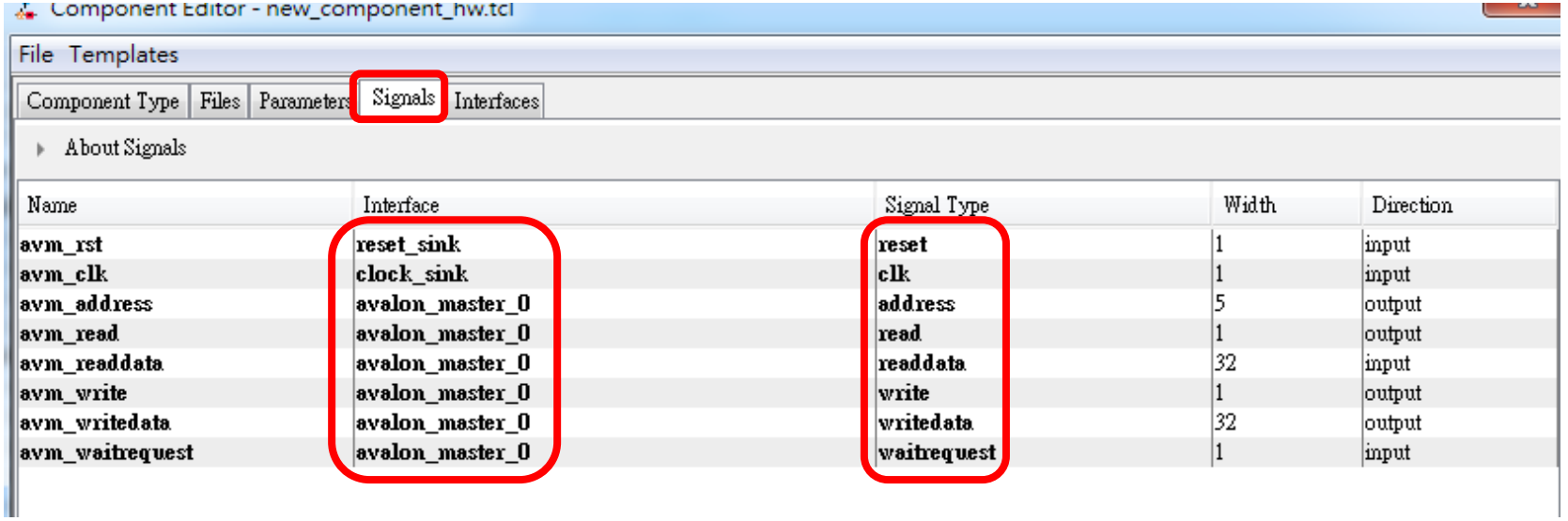
Add custom module to Qsys



2.right-click here to set top-level file if necessary

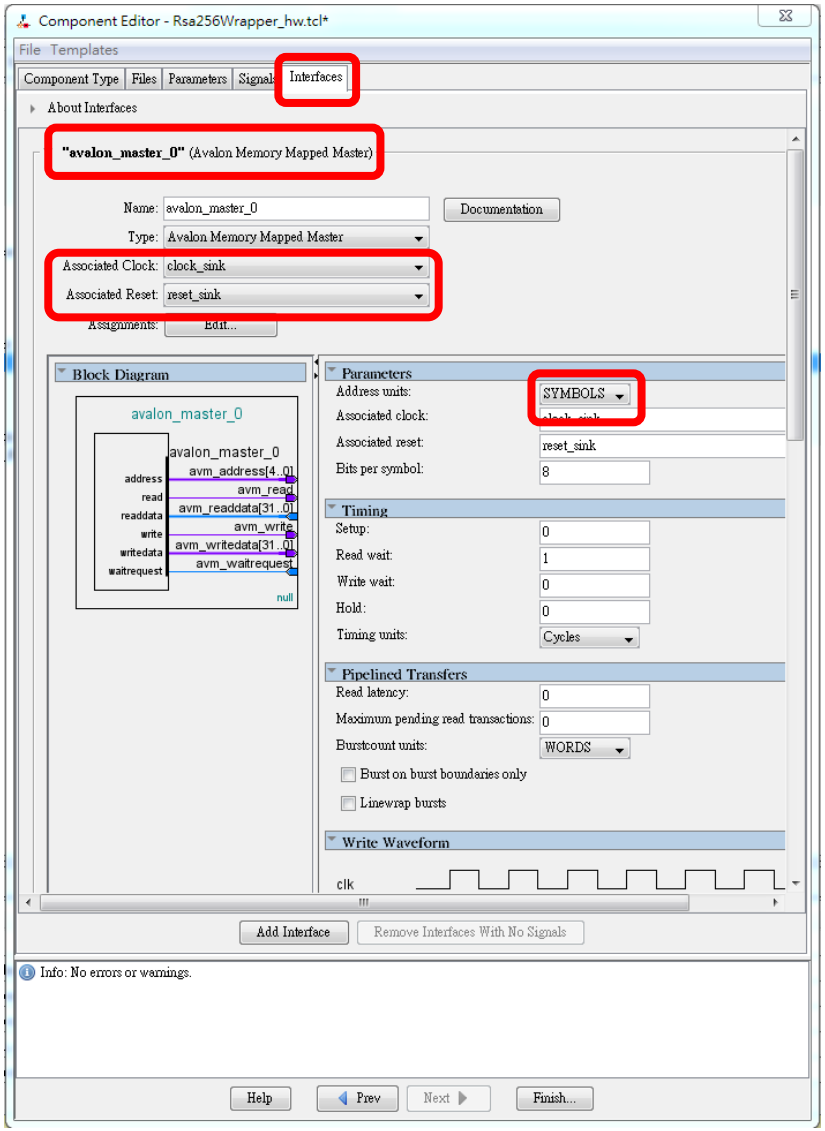


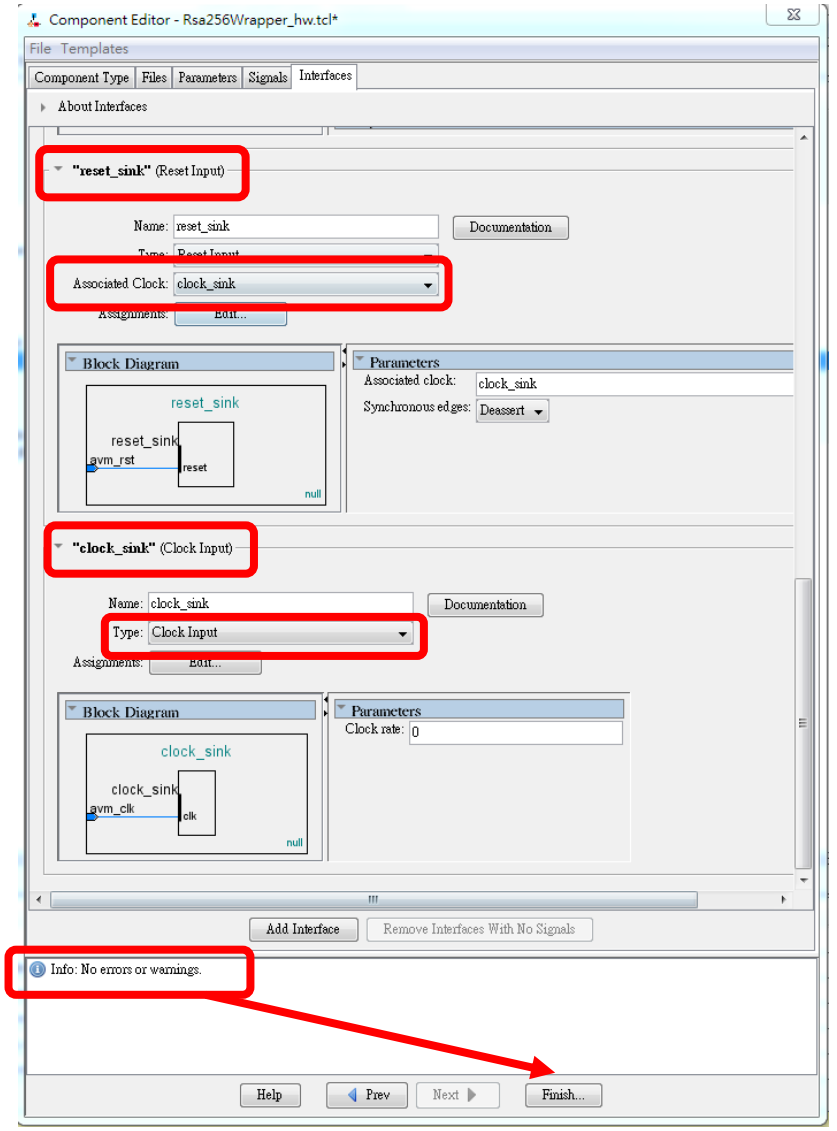
Add custom module to Qsys



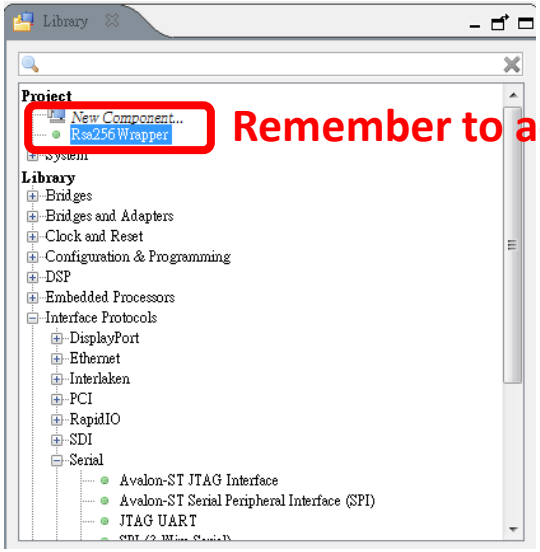
Check the interface & signal type are consistent with Avalon-MM master protocol

Add custom module to Qsys





Qsys system & connection

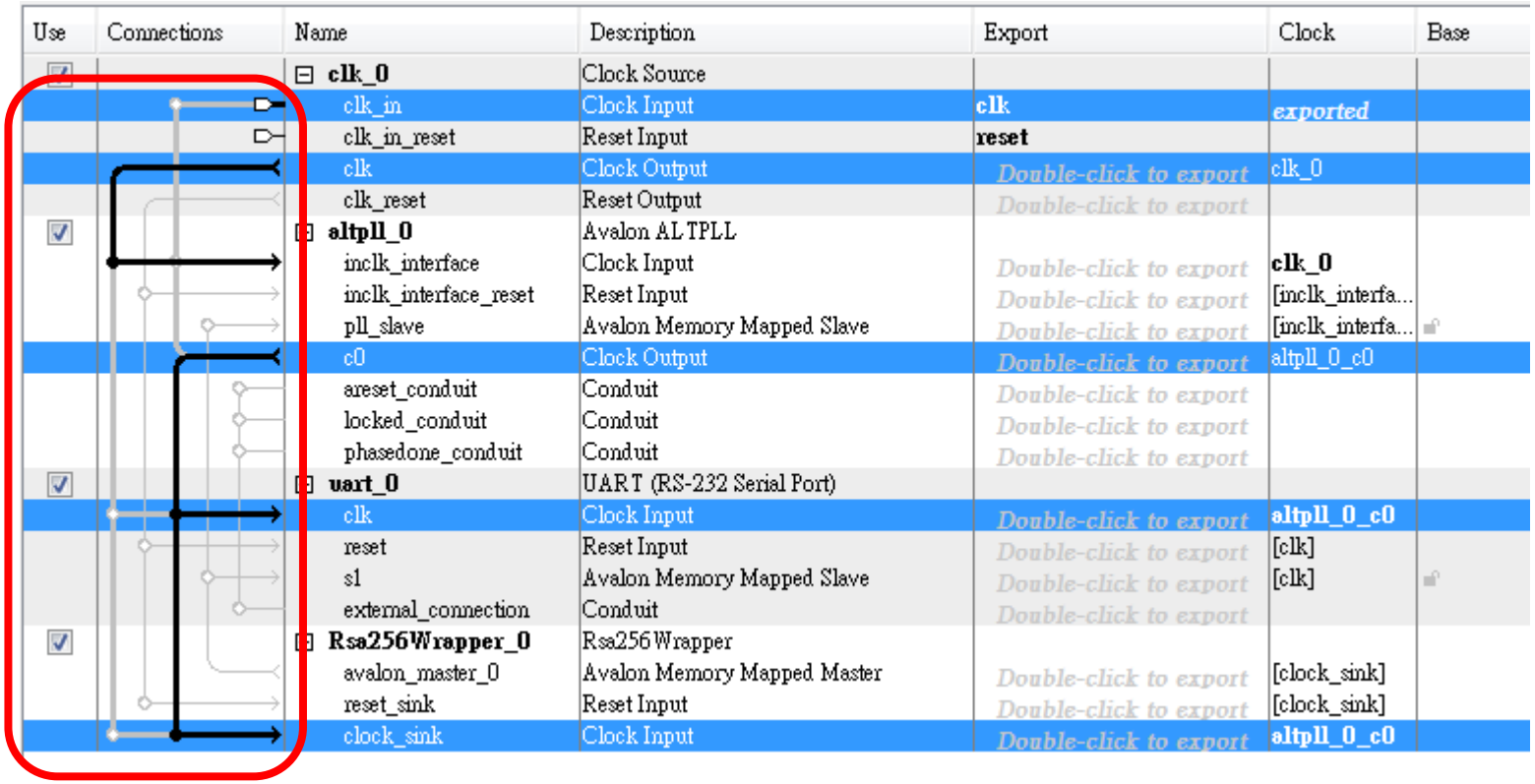


Remember to add your module

Signals here will be in the I/O list of the generated Qsys module

Use	Connections	Name	Description	Export	Clock	Base
☑		**clk_0**	Clock Source			
		clk_in	Clock Input	**clk**	***exported***	
		clk_in_reset	Reset Input	**reset**		
		clk	Clock Output	Double-click to export	clk_0	
		clk_reset	Reset Output	Double-click to export		
☑		**altpll_0**	Avalon ALTPLL			
		inclk_interface	Clock Input	Double-click to export	***unconnected***	
		inclk_interface_reset	Reset Input	Double-click to export	[inclk_interfa...	
		pll_slave	Avalon Memory Mapped Slave	Double-click to export	[inclk_interfa...	
		c0	Clock Output	Double-click to export	altpll_0_c0	
		areset_conduit	Conduit	Double-click to export		
		locked_conduit	Conduit	Double-click to export		
		phasedone_conduit	Conduit	Double-click to export		
☑		**uart_0**	UART (RS-232 Serial Port)			
		clk	Clock Input	Double-click to export	***unconnected***	
		reset	Reset Input	Double-click to export	[clk]	
		s1	Avalon Memory Mapped Slave	Double-click to export	[clk]	
		external_connection	Conduit	Double-click to export		
☑		**Rsa256Wrapper_0**	Rsa256Wrapper			
		avalon_master_0	Avalon Memory Mapped Master	Double-click to export	[clock_sink]	
		reset_sink	Reset Input	Double-click to export	[clock_sink]	
		clock_sink	Clock Input	Double-click to export	***unconnected***	

Qsys system & connection

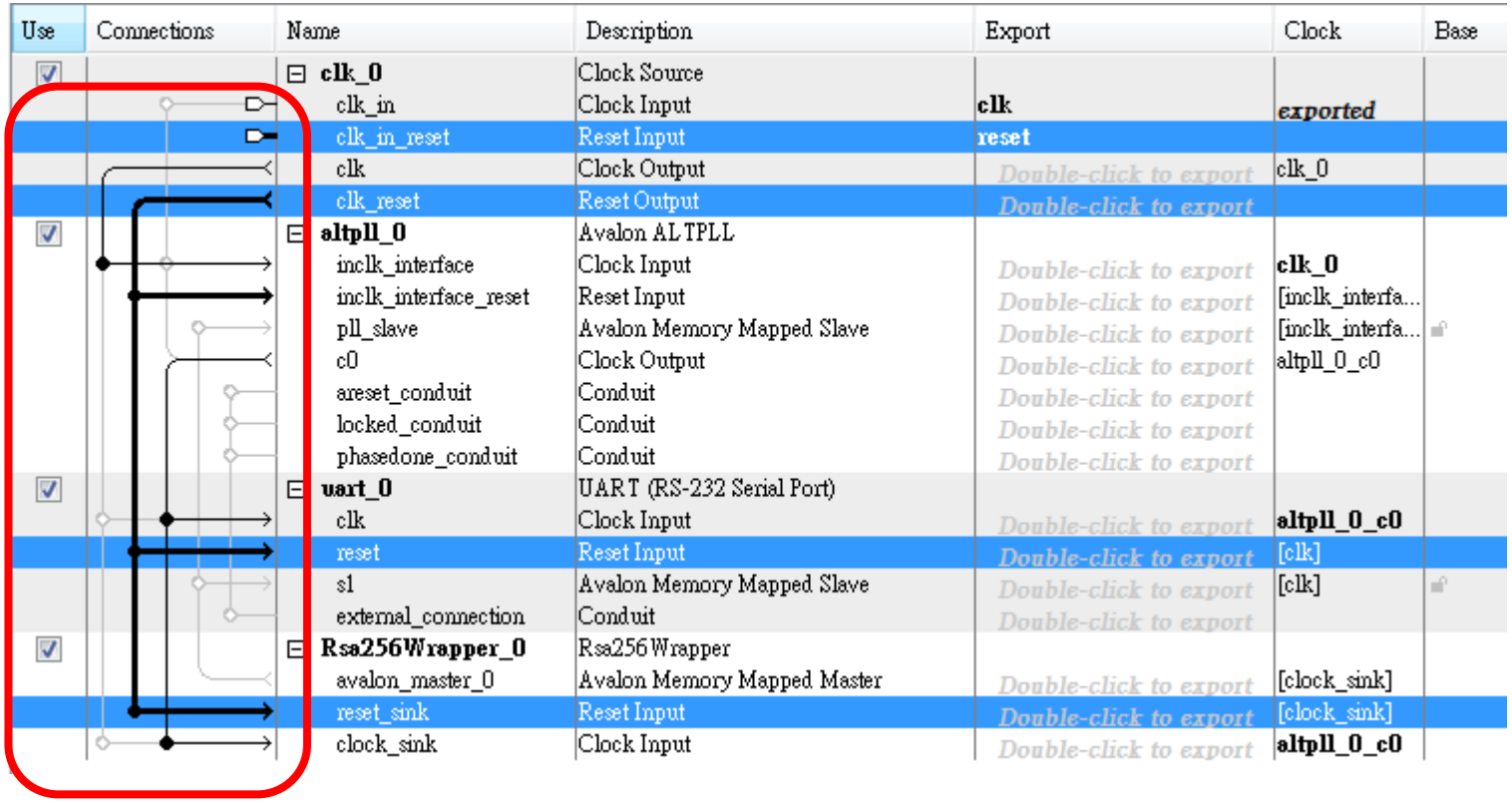


Use	Connections	Name	Description	Export	Clock	Base
☑		**clk_0**	Clock Source			
		clk_in	Clock Input	**clk**	*exported*	
		clk_in_reset	Reset Input	**reset**		
		clk	Clock Output	*Double-click to export*	clk_0	
		clk_reset	Reset Output	*Double-click to export*		
☑		**altpll_0**	Avalon ALTPLL			
		inclk_interface	Clock Input	*Double-click to export*	**clk_0**	
		inclk_interface_reset	Reset Input	*Double-click to export*	[inclk_interfa...	
		pll_slave	Avalon Memory Mapped Slave	*Double-click to export*	[inclk_interfa...	
		c0	Clock Output	*Double-click to export*	altpll_0_c0	
		areset_conduit	Conduit	*Double-click to export*		
		locked_conduit	Conduit	*Double-click to export*		
		phasedone_conduit	Conduit	*Double-click to export*		
☑		**uart_0**	UART (RS-232 Serial Port)			
		clk	Clock Input	*Double-click to export*	**altpll_0_c0**	
		reset	Reset Input	*Double-click to export*	[clk]	
		s1	Avalon Memory Mapped Slave	*Double-click to export*	[clk]	
		external_connection	Conduit	*Double-click to export*		
☑		**Rsa256Wrapper_0**	Rsa256Wrapper			
		avalon_master_0	Avalon Memory Mapped Master	*Double-click to export*	[clock_sink]	
		reset_sink	Reset Input	*Double-click to export*	[clock_sink]	
		clock_sink	Clock Input	*Double-click to export*	**altpll_0_c0**	

Idea of these connections:

DE2-115 clock (clk_in) is used by altpll to generate the clock (c0) for uart-rs232 & rsa module

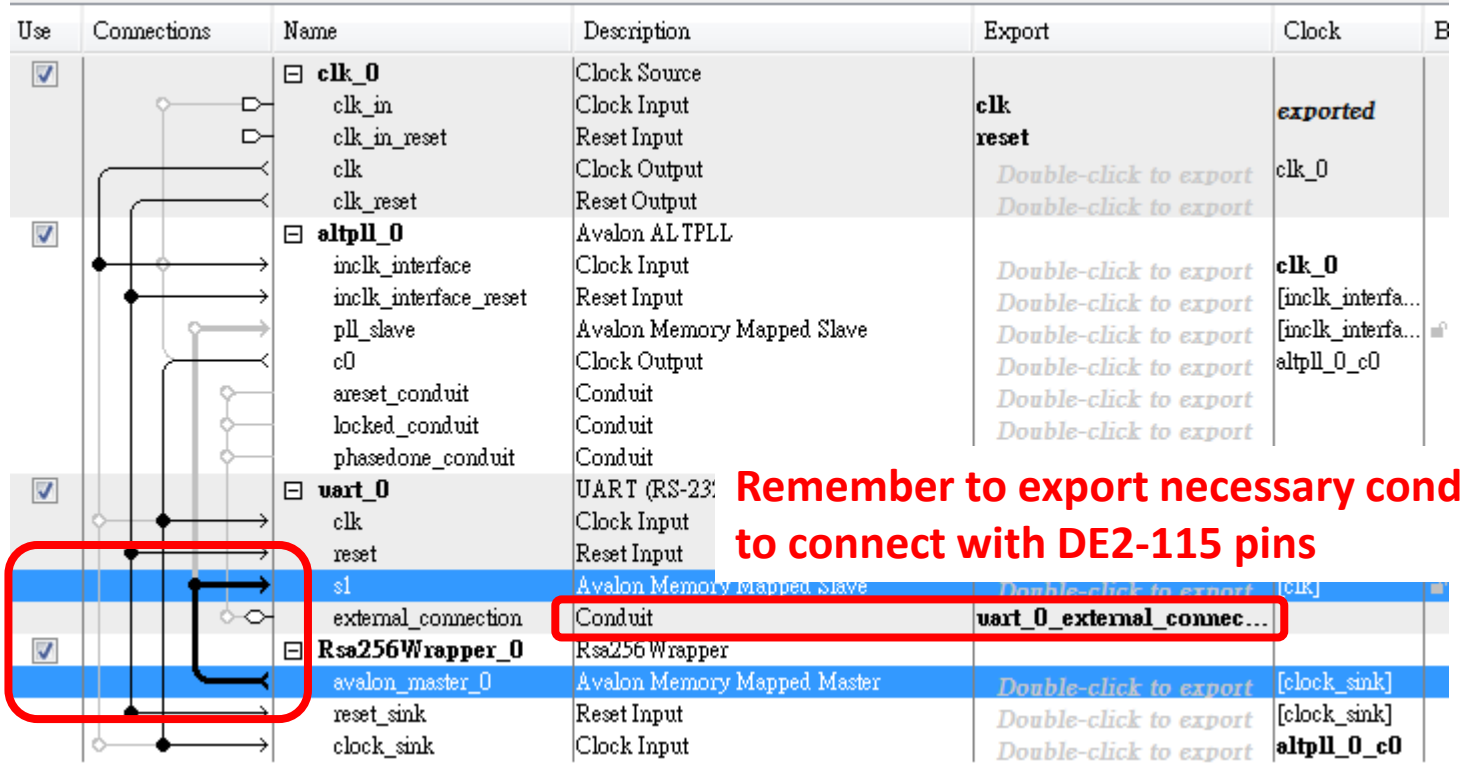
Qsys system & connection



Use	Connections	Name	Description	Export	Clock	Base
☑		**clk_0**	Clock Source			
		clk_in	Clock Input	**clk**	***exported***	
		clk_in_reset	Reset Input	**reset**		
		clk	Clock Output	*Double-click to export*	clk_0	
		clk_reset	Reset Output	*Double-click to export*		
☑		**altpll_0**	Avalon ALTPLL			
		inclk_interface	Clock Input	*Double-click to export*	**clk_0**	
		inclk_interface_reset	Reset Input	*Double-click to export*	[inclk_interfa...	
		pll_slave	Avalon Memory Mapped Slave	*Double-click to export*	[inclk_interfa...	
		c0	Clock Output	*Double-click to export*	altpll_0_c0	
		areset_conduit	Conduit	*Double-click to export*		
		locked_conduit	Conduit	*Double-click to export*		
		phasedone_conduit	Conduit	*Double-click to export*		
☑		**uart_0**	UART (RS-232 Serial Port)			
		clk	Clock Input	*Double-click to export*	**altpll_0_c0**	
		reset	Reset Input	*Double-click to export*	[clk]	
		s1	Avalon Memory Mapped Slave	*Double-click to export*	[clk]	
		external_connection	Conduit	*Double-click to export*		
☑		**Rsa256Wrapper_0**	Rsa256Wrapper			
		avalon_master_0	Avalon Memory Mapped Master	*Double-click to export*	[clock_sink]	
		reset_sink	Reset Input	*Double-click to export*	[clock_sink]	
		clock_sink	Clock Input	*Double-click to export*	**altpll_0_c0**	

Similarly, these connection means the reset signal from DE2-115 works as reset signal for altpll, uart-rs232, and rsa module.

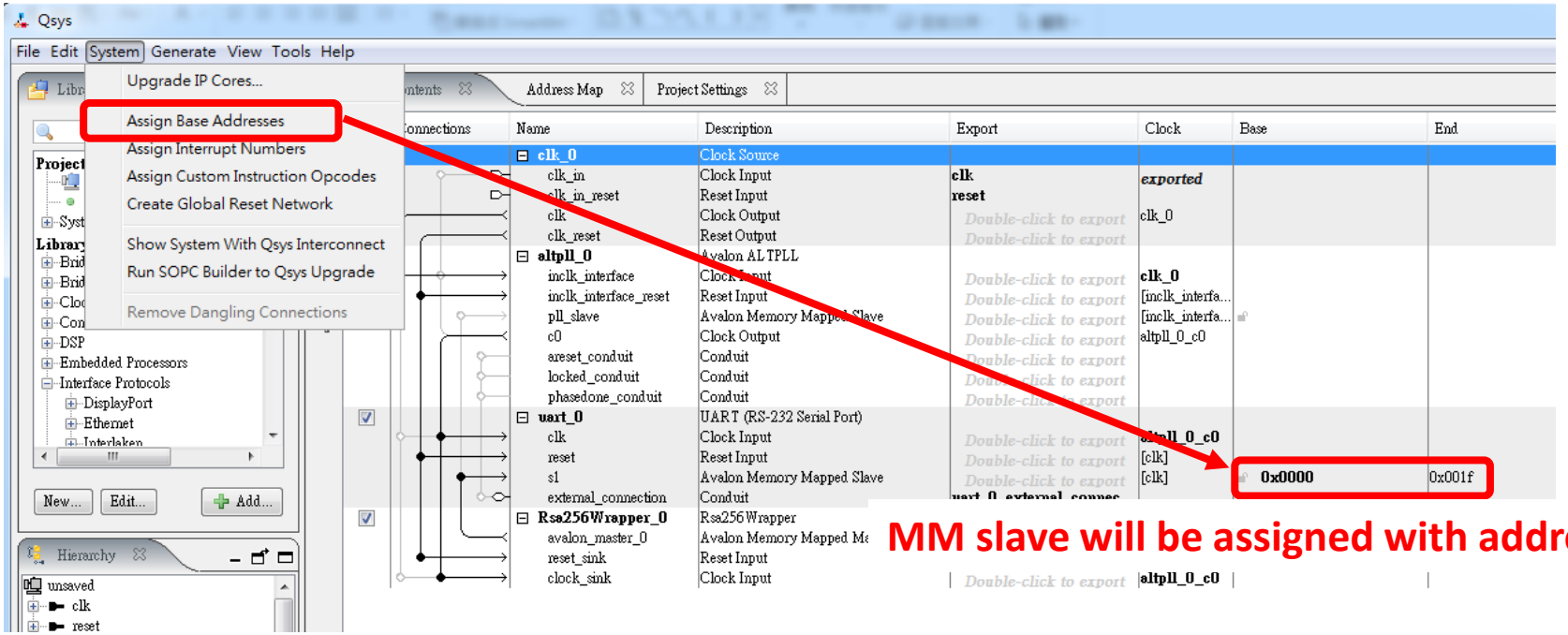
Qsys system & connection



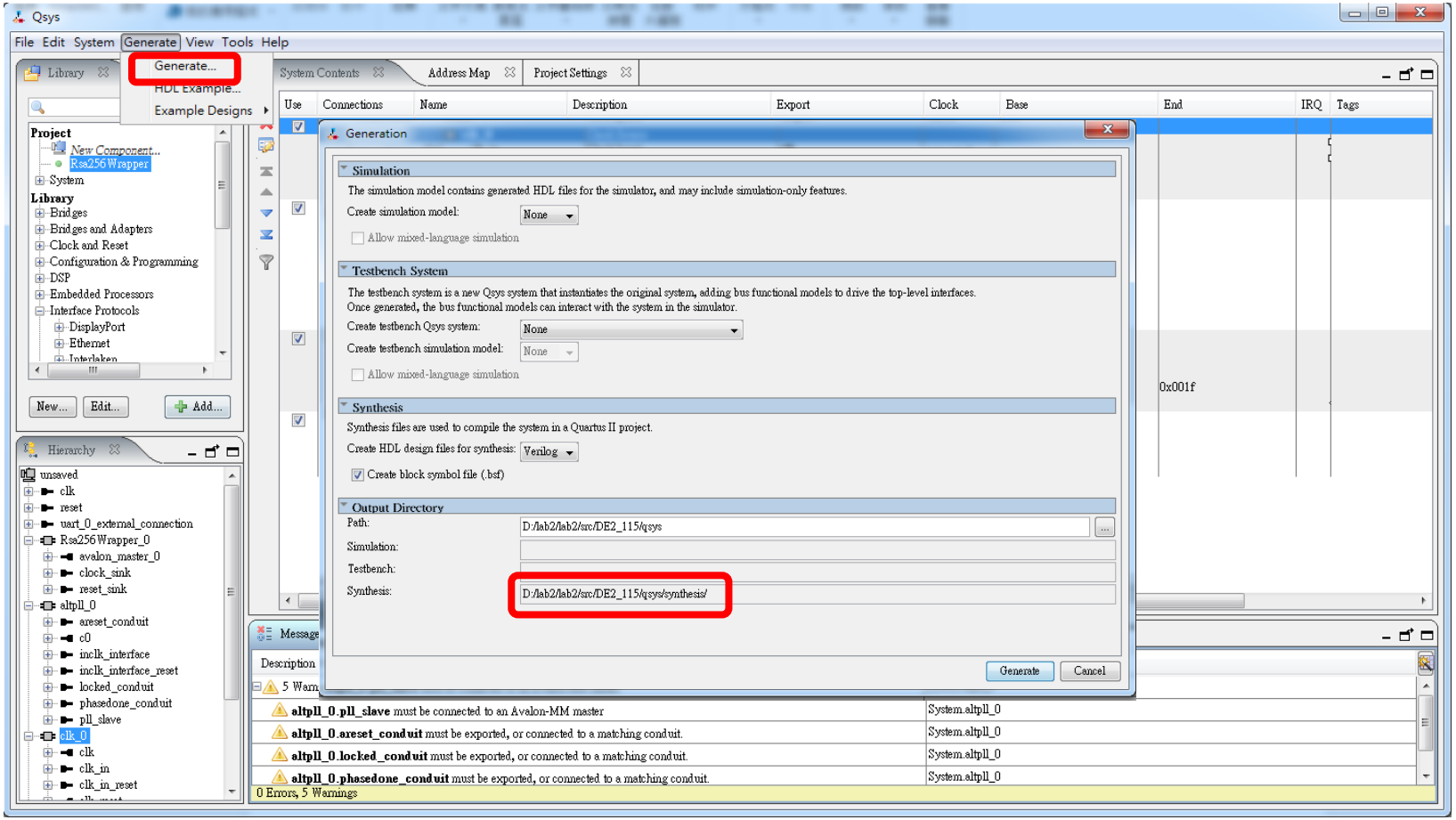
Remember to export necessary conduit signals to connect with DE2-115 pins

Master-slave pair in this lab

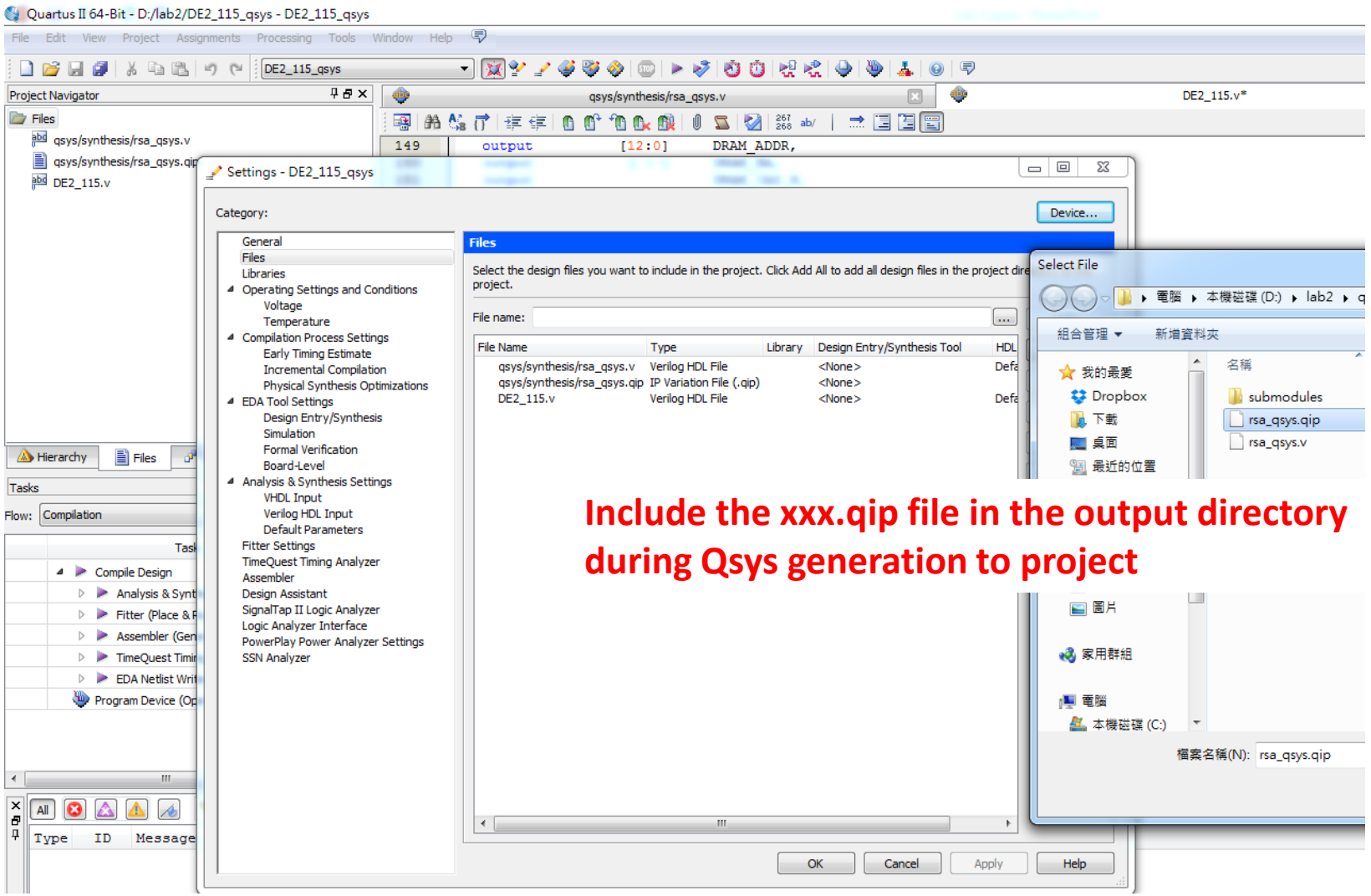
Generate Qsys module



Generate Qsys module



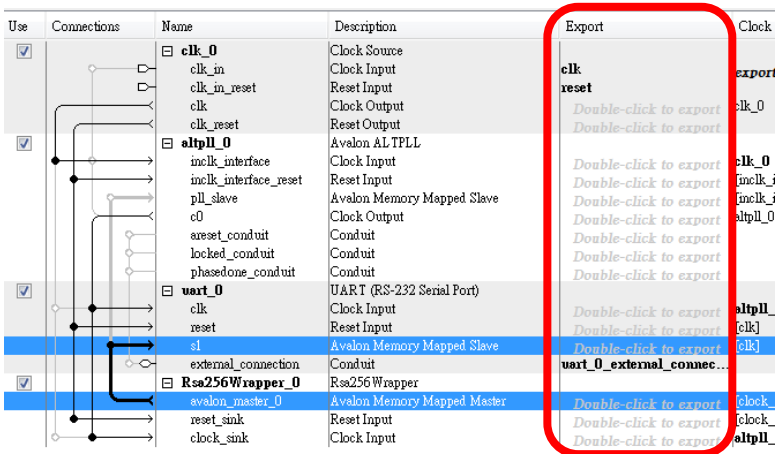
Include generated Qsys module



Instantiate Qsys module

- Finally, remember to instantiate the generate Qsys module.
  - For the module I/O, refer to xxx.v in the same directory of xxx.qip

```
`timescale 1 ps / 1 ps
module rsa_qsys (
      input  wire  clk_clk,                       //                    clk.clk
      input  wire  reset_reset_n,                 //                  reset.reset_n
      input  wire  uart_0_external_connection_rxd, // uart_0_external_connection.rxd
      output wire  uart_0_external_connection_txd  //                            .txd
   );
```



Generated module I/O should be the same as Exported signals in Qsys