

ASIC WiMax Phy: Phase 2

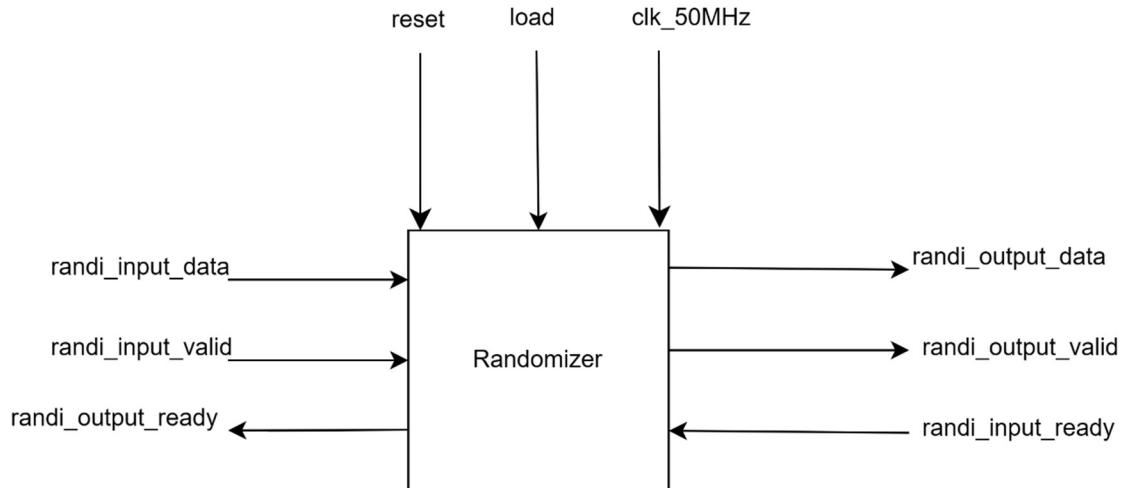
Randomizer

Main File Name	Randi.vhd
Testbench File Name	Randi_tb.vhd

Ports:

Signals	In/Out	Type	Width
Clk_50MHz	In	std_logic	1
Reset	In	std_logic	1
randi_input_data	In	std_logic	1
randi_input_ready	In	std_logic	1
randi_input_valid	In	std_logic	1
randi_output_valid	Out	std_logic	1
randi_output_data	Out	std_logic	1
randi_output_ready	Out	std_logic	1

Block Diagram Sketch:

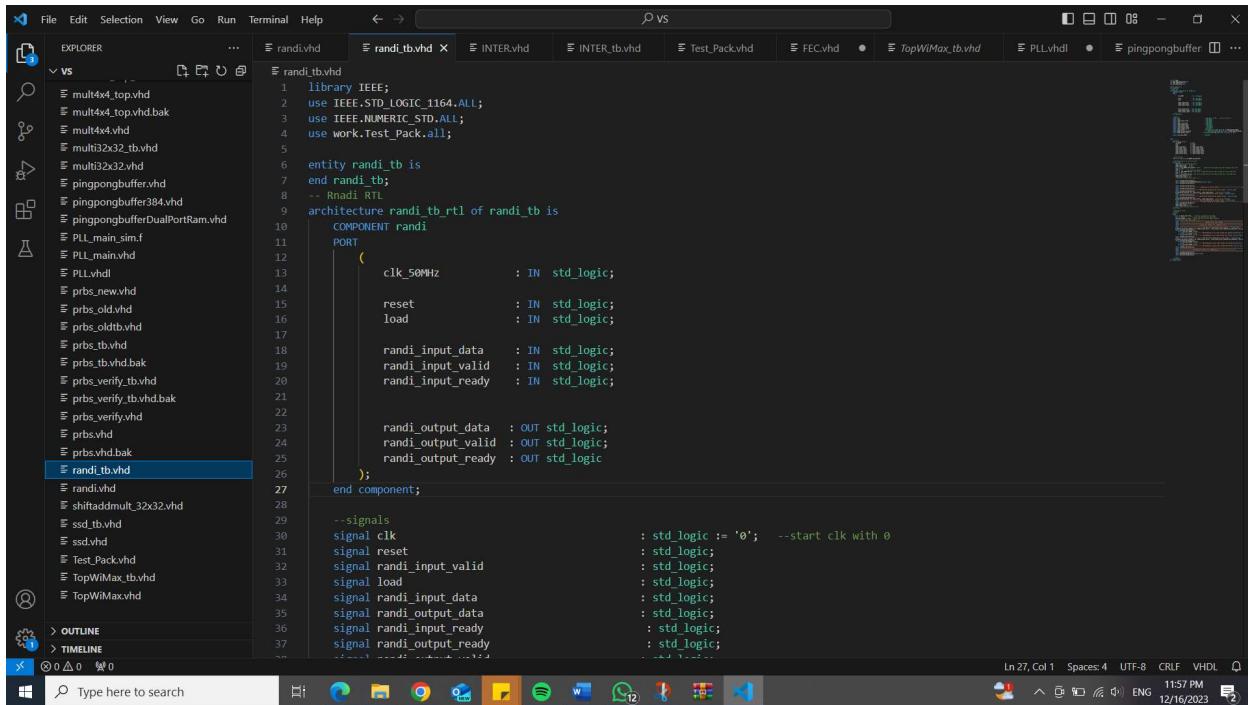


Testing and Functionality

- We have tested on 2 Input streams
- We have 2x96 cycles, with a period of 20 ns = (3940 ns runtime in testbench).
- No Setup nor warm-up cycles

- Handshakes (ready and valid signals) are in phase with inputs/outputs.
- Test bench utilizes Test_Pack package to compare bit vectors and ensure correct module functionality and uses procedures

Package:



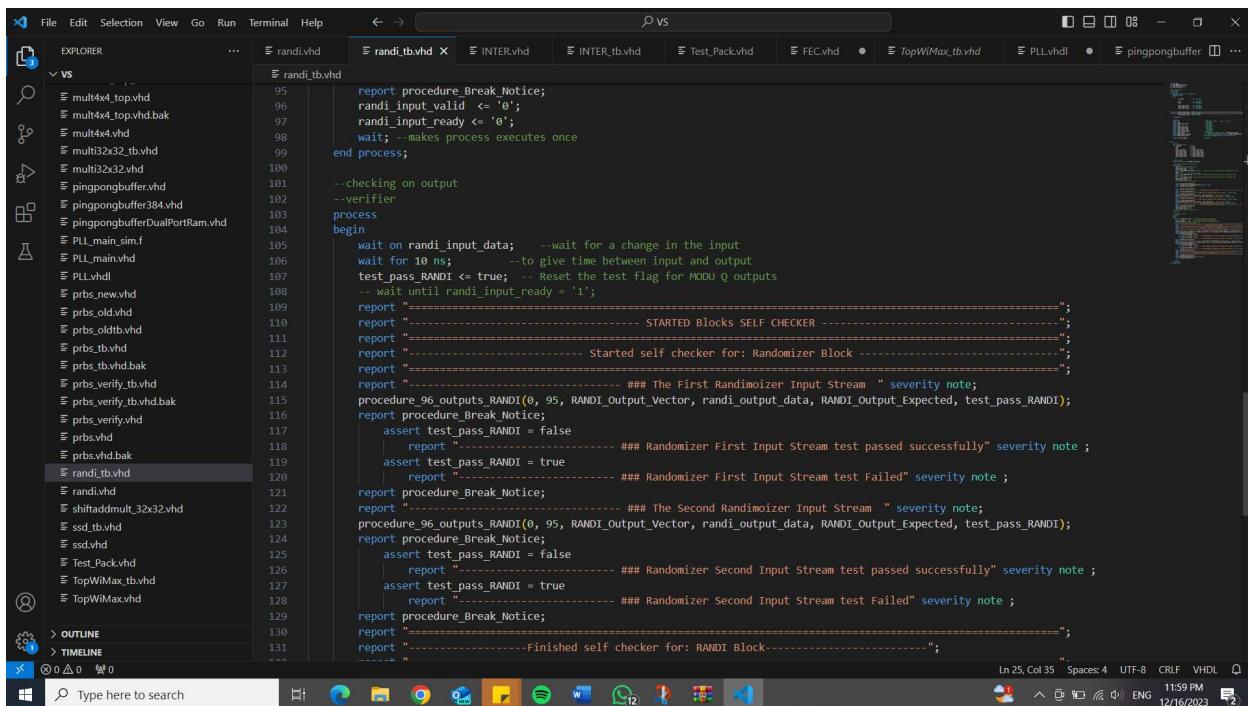
```

EXPLORER
vs
randi_tb.vhd
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.NUMERIC_STD.ALL;
use work.Test_Pack.all;

entity randi_tb is
end randi_tb;
-- Rndi RTL
architecture randi_tb_rtl of randi_tb is
COMPONENT randi
PORT(
    clk_50MHz : IN std_logic;
    reset : IN std_logic;
    load : IN std_logic;
    randi_input_data : IN std_logic;
    randi_input_valid : IN std_logic;
    randi_input_ready : IN std_logic;
    randi_output_data : OUT std_logic;
    randi_output_valid : OUT std_logic;
    randi_output_ready : OUT std_logic
);
end component;
--signals
signal clk : std_logic := '0'; --start clk with 0
signal reset : std_logic;
signal randi_input_valid : std_logic;
signal load : std_logic;
signal randi_input_data : std_logic;
signal randi_output_data : std_logic;
signal randi_input_ready : std_logic;
signal randi_output_valid : std_logic;
signal randi_output_ready : std_logic;
begin
end;

```

Procedure



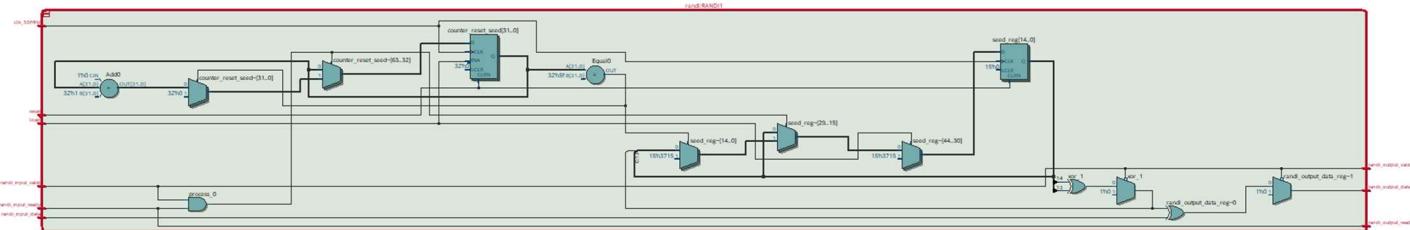
```

EXPLORER
vs
randi_tb.vhd
report procedure Break_Notify;
randi_input_valid <= '0';
randi_input_ready <= '0';
wait; --makes process executes once
end process;

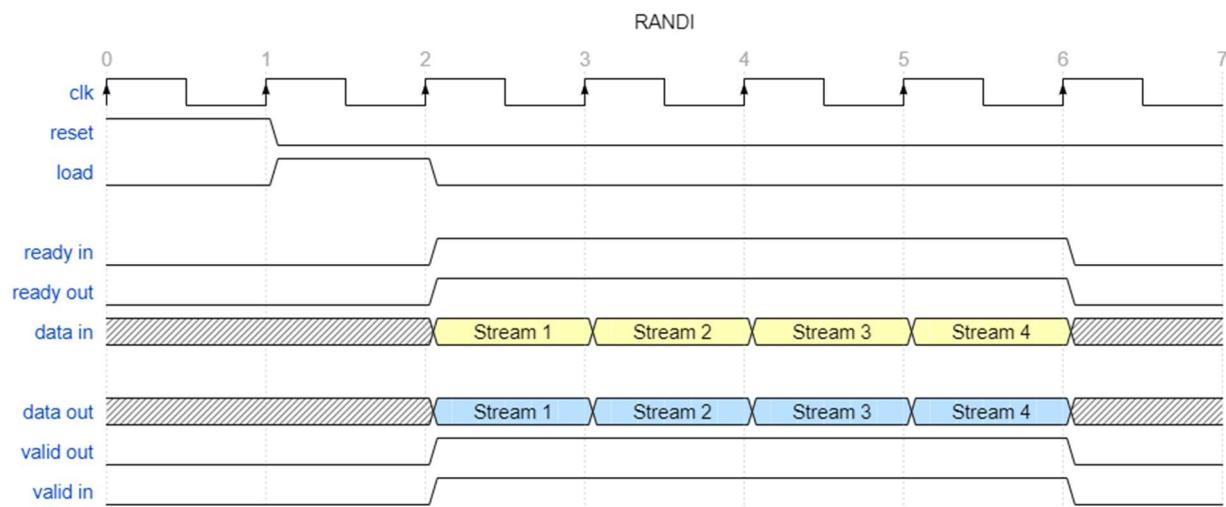
--checking on output
--verifier
process
begin
    wait on randi_input_data; --wait for a change in the input
    wait for 10 ns; --to give time between input and output
    test_pass_RANDI <= true; -- Reset the test flag for MODU Q outputs
    -- wait until randi_input_ready = '1';
    report "----- STARTED Blocks SELF CHECKER -----";
    report "----- Started self checker for: Randomizer Block -----";
    report "----- ### The First Randomizer Input Stream " severity note;
procedure 96_outputs_RANDI(o, 95, RANDI_Output_Vector, randi_output_data, RANDI_Output_Expected, test_pass_RANDI);
procedure procedure_Break_Notify;
    assert test_pass_RANDI = false
        report "----- Randomizer First Input Stream test passed successfully" severity note ;
    assert test_pass_RANDI = true
        report "----- Randomizer First Input Stream test Failed" severity note ;
report procedure_Break_Notify;
report "----- ### The Second Randomizer Input Stream " severity note;
procedure 96_outputs_RANDI(o, 95, RANDI_Output_Vector, randi_output_data, RANDI_Output_Expected, test_pass_RANDI);
report procedure_Break_Notify;
    assert test_pass_RANDI = false
        report "----- Randomizer Second Input Stream test passed successfully" severity note ;
    assert test_pass_RANDI = true
        report "----- Randomizer Second Input Stream test Failed" severity note ;
report procedure_Break_Notify;
report "----- Finished self checker for: RANDI Block-----";
report "----- ";

```

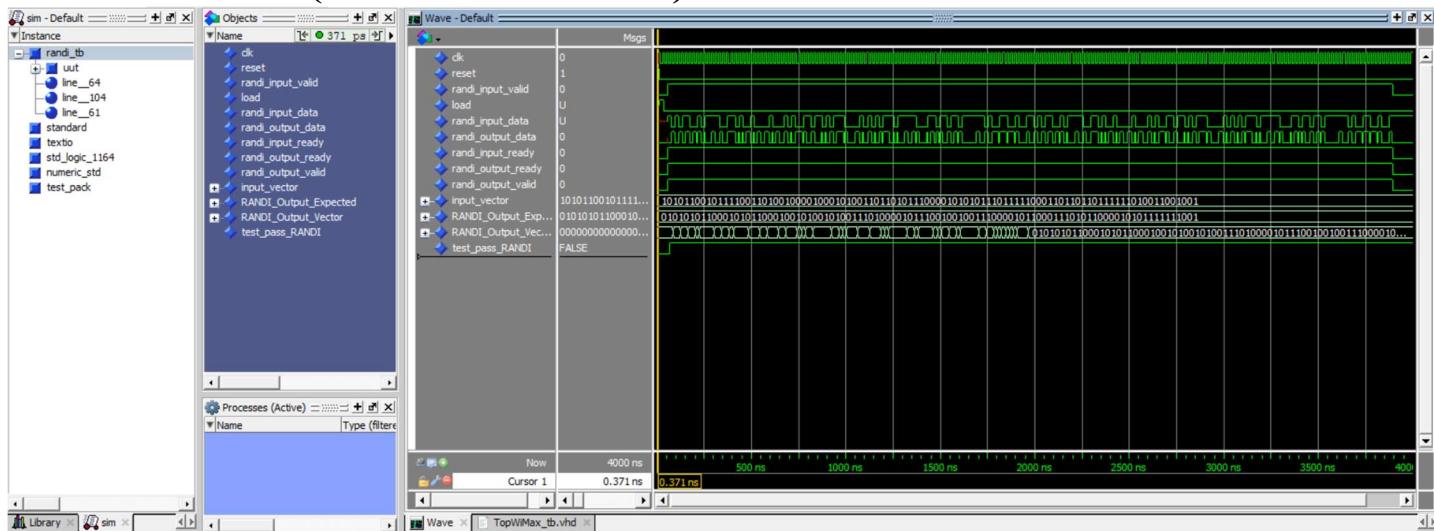
RTL (Quartus)



Waveform (Sketch Expect)



Randi Results (ModelSim Altera)



Randi Self-Check Transcript (ModelSim Altera)

The screenshot shows the ModelSim Altera transcript window. It displays a log of events from the simulation, including notes about input streams, randomizer tests, and the completion of the self-checker for the RANDI Block. The transcript ends with a note indicating the simulation has finished.

```
*** Note: ----- ## The Second Randomizer Input Stream
# Time: 1985 ns Iteration: 0 Instance: /randi_tb
*** Note: ----- ## Done Inputting the Second stream:
# Time: 3895 ns Iteration: 0 Instance: /randi_tb
*** Note: ----- Finished Inputting [2] Input Streams
# Time: 3895 ns Iteration: 0 Instance: /randi_tb
*** Note: -----
# Time: 3895 ns Iteration: 0 Instance: /randi_tb
*** Note: -----
# Time: 3905 ns Iteration: 0 Instance: /randi_tb
*** Note: ----- ## Randomizer Second Input Stream test passed successfully
# Time: 3905 ns Iteration: 0 Instance: /randi_tb
*** Note: -----
# Time: 3905 ns Iteration: 0 Instance: /randi_tb
*** Note: -----
# Time: 3905 ns Iteration: 0 Instance: /randi_tb
*** Note: -----
# Time: 3905 ns Iteration: 0 Instance: /randi_tb
*** Note: -----
# Time: 3905 ns Iteration: 0 Instance: /randi_tb
*** Note: -----Finished self checker for: RANDI Block-----
# Time: 3905 ns Iteration: 0 Instance: /randi_tb
*** Note: -----
# Time: 3905 ns Iteration: 0 Instance: /randi_tb
*** Note: -----
# Time: 3905 ns Iteration: 0 Instance: /randi_tb
*** Note: ----- Simulation Finished
# Time: 3905 ns Iteration: 0 Instance: /randi_tb
*** Note: -----
# Time: 3905 ns Iteration: 0 Instance: /randi_tb
```

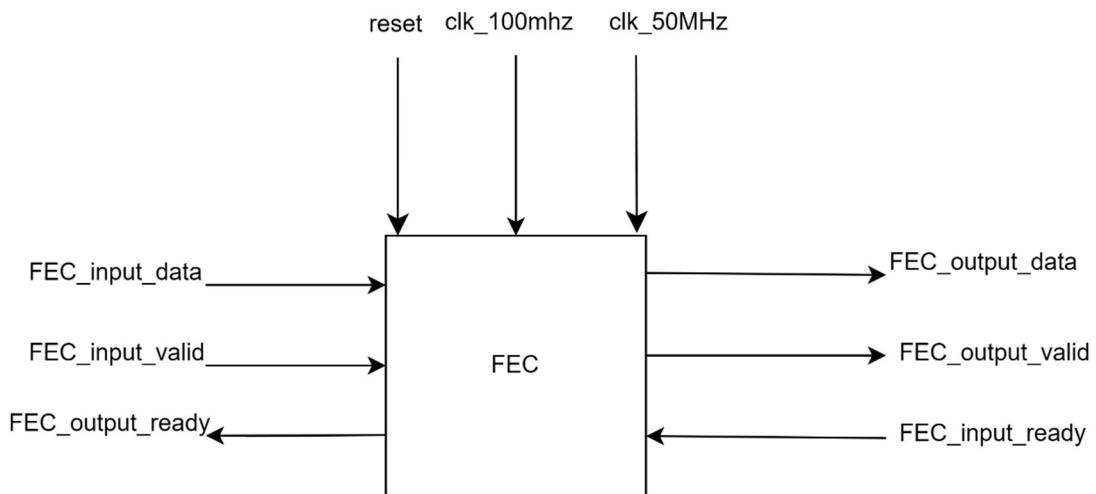
FEC

Main File Name	FEC.vhd
Testbench File Name	FEC_tb.vhd
Extra File	FEC_RAM_2PORTS.vhd

Ports:

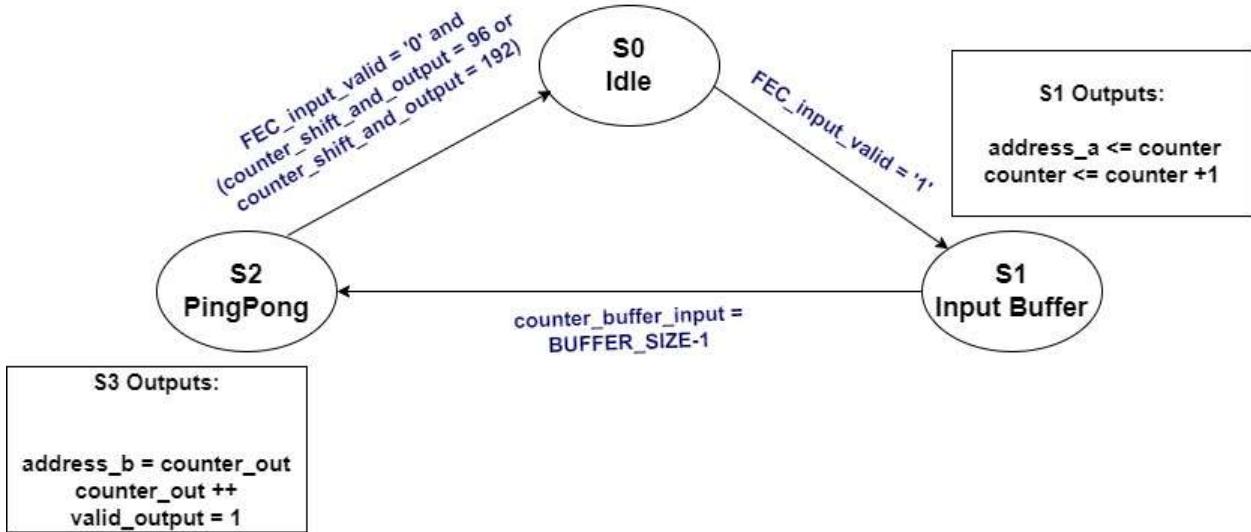
Signals	In/Out	Type	Width
clk_50MHz	In	STD_LOGIC	1
clk_100MHz	In	STD_LOGIC	1
reset	In	STD_LOGIC	1
FEC_input_data	In	STD_LOGIC	1
FEC_input_ready	In	STD_LOGIC	1
FEC_input_valid	In	STD_LOGIC	1
FEC_output_ready	Out	STD_LOGIC	1
FEC_output_valid	Out	STD_LOGIC	1
FEC_output_data	Out	STD_LOGIC_VECTOR	1

Block Diagram Sketch:

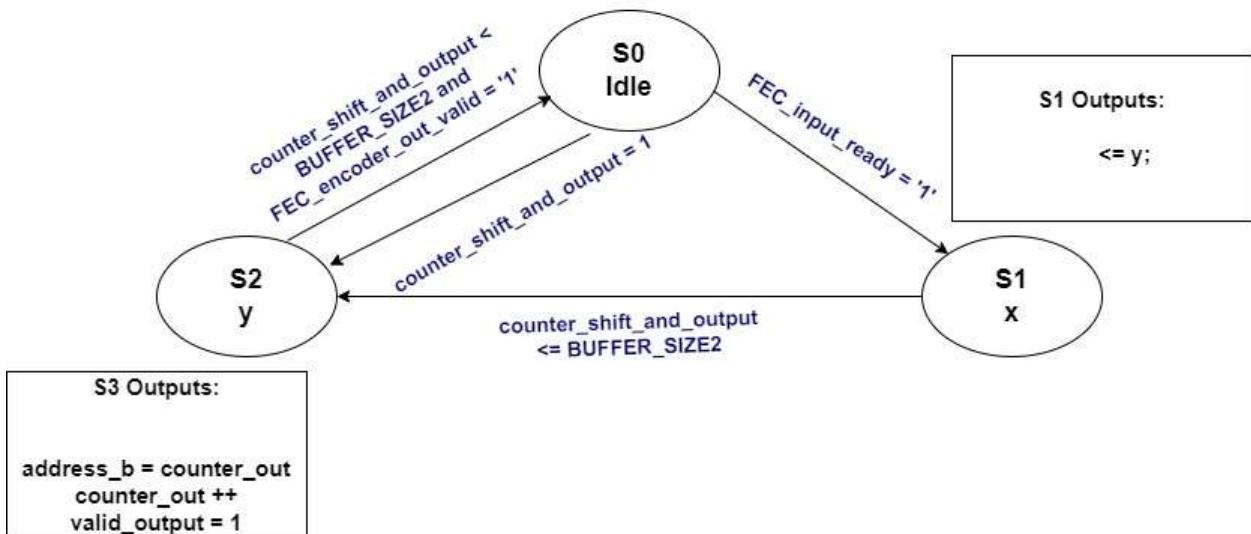


Finite State Machine Diagram

FEC Input: Finite State Machine



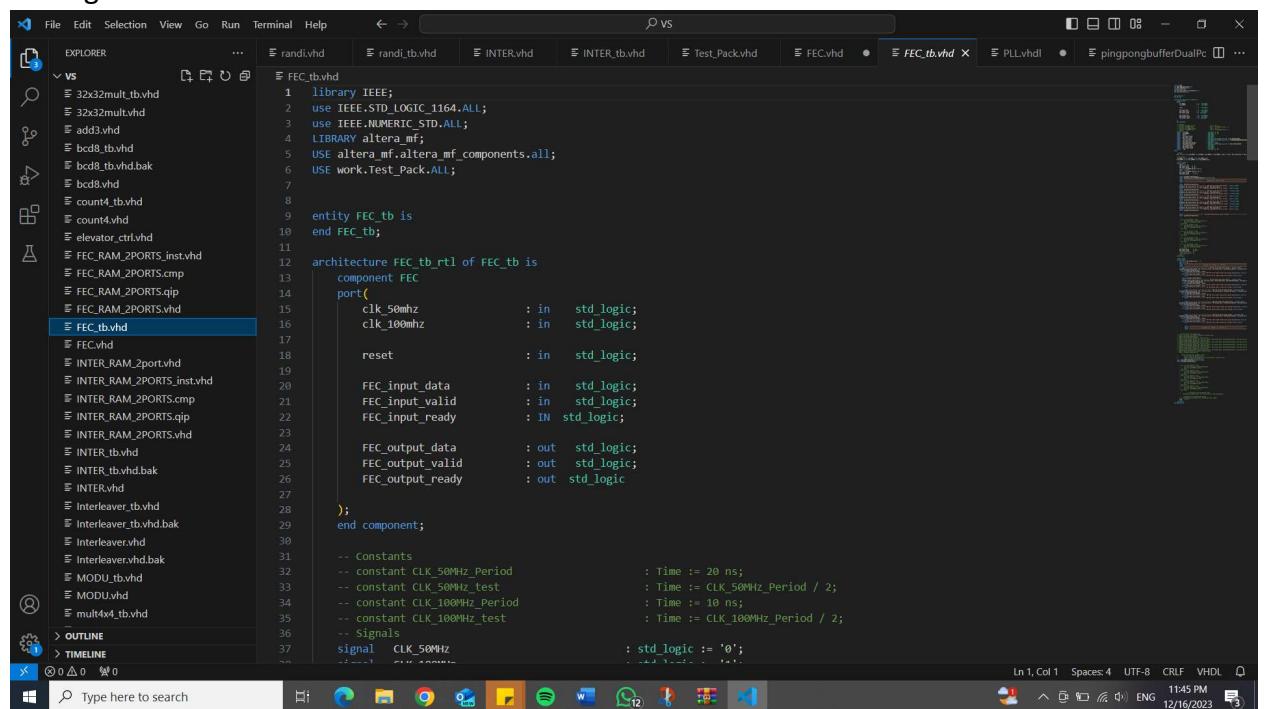
FEC Output: Finite State Machine



Testing and Functionality

- 1- 5 streams of Data, around 10 us of inputting, and 10 us of outputting but delayed by 2 us. So total run time is a round 12 us.
- 2- We receive 96 data bits at 50 MHz, at period 20 ns. They are processed in parallel and sent to our PLL_BLOCK (RAM).
- 3- Then we transmit 192 bits of data a faster rate of 100 MHz.
- 4- We need to either send or receive not both. And repeats the cycle again.
- 5- The test bench employs a package Test_Pack, and the built-in Altera package with procedures inside that package .

Packages used



```
File Edit Selection View Go Run Terminal Help < > VS
EXPLORER randi.vhd randi_tb.vhd INTER.vhd INTER_tb.vhd Test_Pack.vhd FEC.vhd FEC_tb.vhd PLLvhdl pingpongbufferDualPc ...
FEC.tb.vhd
1 library IEEE;
2 use IEEE.STD.LOGIC_1164.ALL;
3 use IEEE.NUMERIC_STD.ALL;
4 LIBRARY altera_mf;
5 USE altera_mf.altera_mf_components.all;
6 USE work.Test_Pack.ALL;
7
8
9 entity FEC_tb is
10 end FEC_tb;
11
12 architecture FEC_tb_rtl of FEC_tb is
13 component FEC
14 port(
15     clk_50MHz : in std_logic;
16     clk_100MHz : in std_logic;
17     reset : in std_logic;
18     FEC_input_data : in std_logic;
19     FEC_input_valid : in std_logic;
20     FEC_input_ready : IN std_logic;
21     FEC_output_data : out std_logic;
22     FEC_output_valid : out std_logic;
23     FEC_output_ready : out std_logic
24 );
25 end component;
26
27
28
29
30
31
32 -- Constants
33 -- constant CLK_50MHz_Period : Time := 20 ns;
34 -- constant CLK_50MHz_Test : Time := CLK_50MHz_Period / 2;
35 -- constant CLK_100MHz_Period : Time := 10 ns;
36 -- constant CLK_100MHz_Test : Time := CLK_100MHz_Period / 2;
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
59 signal CLK_50MHz : std_logic := '0';
59 end;
```

Procedures used

The screenshot shows a software interface for VHDL development. On the left is the Explorer pane, which lists various VHDL files and their versions. The main area is a code editor displaying a VHDL testbench for a FEC module. The code includes processes for generating input streams and checking output against expected values. A waveform viewer is visible on the right side of the interface.

```
File Edit Selection View Go Run Terminal Help < > J vs
```

```
EXPLORER
vs
32x32mult_tb.vhd
32x32mult.vhd
add3.vhd
bcd8_tb.vhd
bcd8_tb.vhd.bak
bcd8.vhd
count4_tb.vhd
count4.vhd
elevator_ctrl.vhd
FEC_RAM_2PORTS_inst.vhd
FEC_RAM_2PORTS.cmp
FEC_RAM_2PORTS.qip
FEC_RAM_2PORTS.vhd
FEC_tb.vhd
FEC.vhd
INTER_RAM_2portvhd
INTER_RAM_2PORTS_inst.vhd
INTER_RAM_2PORTS.cmp
INTER_RAM_2PORTS.qip
INTER_RAM_2PORTS.vhd
INTER_tb.vhd
INTER_tb.vhd.bak
INTER_tb.vhd
Interleaver_tb.vhd
Interleaver_tb.vhd.bak
Interleaver.vhd
Interleaver.vhd.bak
MODU_tb.vhd
MODU.vhd
mult4x1_tb.vhd

> OUTLINE
> TIMELINE
```

```
randi.vhd randi_tb.vhd INTER.vhd INTER_tb.vhd Test_Pack.vhd FEC.vhd FEC_tb.vhd PLL.vhd pingpongbufferDualPc.vhd
```

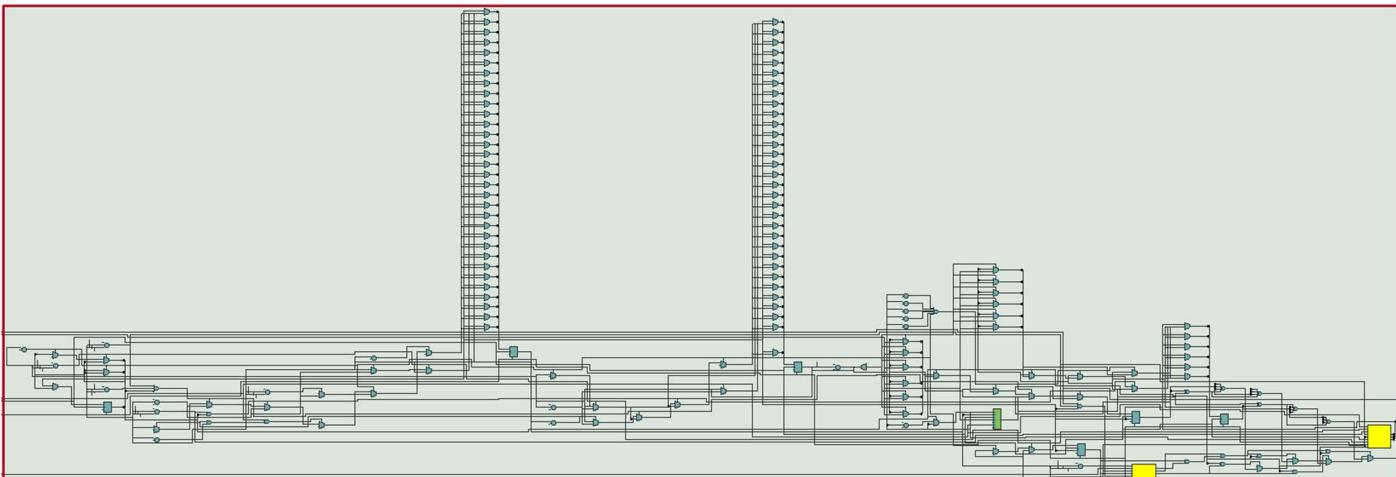
```
-- end loop;
FEC_input_valid      <= '0';
FEC_input_ready     <= '0';
--wait until flag = '1';
--FEC_input_data <= '0';
wait;
end process;

--check output
process begin
    wait until FEC_output_valid = '1';
    wait for 2 ns;
    report "-----";
    report "----- Started self checker for: FEC Block -----";
    report "-----";
    report "----- ##> The First FEC Input Stream " severity note;
procedure_192_outputs_FEC(0, 191, test_out_vector, FEC_output_data, FEC_Expected_Output, test_pass_fec_encoder);
report procedure_Break_Notify;
assert test_pass_fec_encoder = false
    report "----- ##> FEC First Input Stream test passed Successfully" severity note ;
assert test_pass_fec_encoder = true
    report "----- ##> FEC First Input Stream test Failed" severity note ;

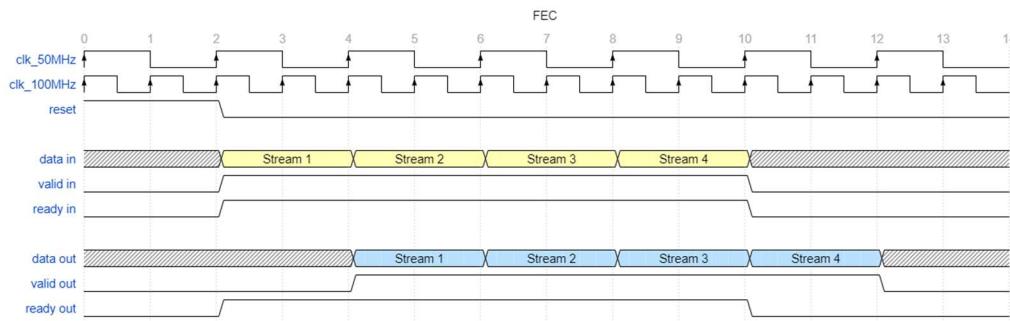
report procedure_Break_Notify;
report "----- ##> The Second FEC Input Stream " severity note;
procedure_192_outputs_FEC(0, 191, test_out_vector, FEC_output_data, FEC_Expected_Output, test_pass_fec_encoder);
report procedure_Break_Notify;
assert test_pass_fec_encoder = false
    report "----- ##> FEC Second Input Stream test passed Successfully" severity note ;
assert test_pass_fec_encoder = true
    report "----- ##> FEC Second Input Stream test Failed" severity note ;

report "----- ##> The Third FEC Input Stream " severity note;
procedure_192_outputs_FEC(0, 191, test_out_vector, FEC_output_data, FEC_Expected_Output, test_pass_fec_encoder);
report procedure_Break_Notify;
assert test_pass_fec_encoder = false
    report "----- ##> FEC Third Input Stream test passed Successfully" severity note ;
```

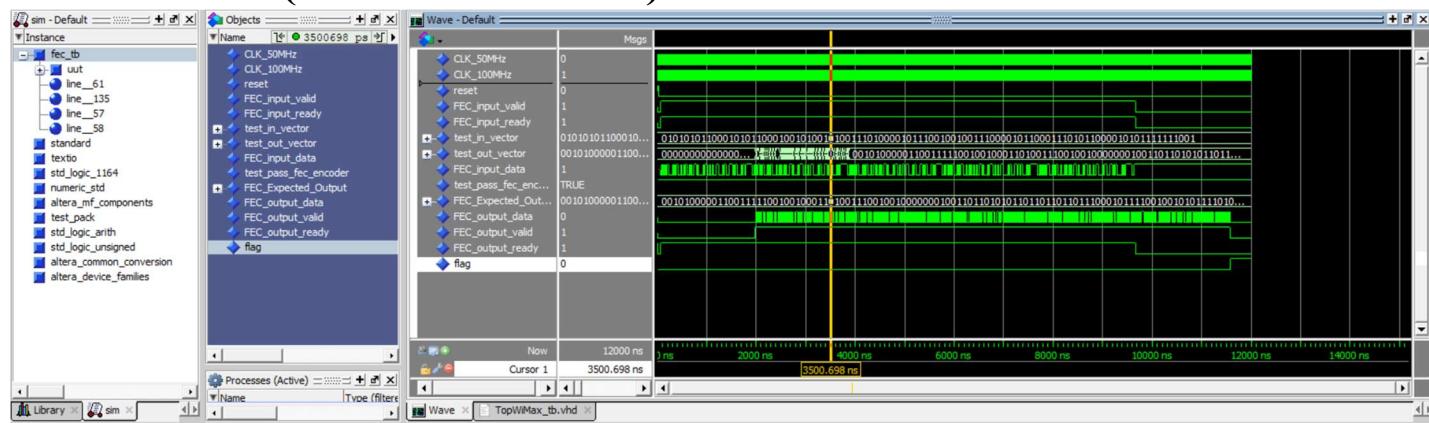
RTL (Quartus)



Waveform (Sketch Expect)



FEC Results (ModelSim Altera)



FEC Self Check (ModelSim Altera)

```
# ** Note: -----
#   Time: 7740 ns Iteration: 0 Instance: /fec_tb
# ** Note: ----- ## The Fifth FEC Input Stream
#   Time: 7740 ns Iteration: 0 Instance: /fec_tb
VSIM 39> run 2 us
# ** Note: -----
#   Time: 9652 ns Iteration: 0 Instance: /fec_tb
# ** Note: ----- ## FEC Fourth Input Stream test passed Successfully
#   Time: 9652 ns Iteration: 0 Instance: /fec_tb
# ** Note: ----- ## The Fifth FEC Input Stream
#   Time: 9652 ns Iteration: 0 Instance: /fec_tb
# ** Note: ----- ## Done Inputting the Fifth stream:
#   Time: 9660 ns Iteration: 0 Instance: /fec_tb
# ** Note: -----
#   Time: 9660 ns Iteration: 0 Instance: /fec_tb
# ** Note: ----- Finished Inputting (5) Input Streams -----
#   Time: 9660 ns Iteration: 0 Instance: /fec_tb
# ** Note: -----
#   Time: 9660 ns Iteration: 0 Instance: /fec_tb
VSIM 40> run 2 us
# ** Note: -----
#   Time: 11572 ns Iteration: 0 Instance: /fec_tb
# ** Note: ----- ## FEC Fifth Input Stream test passed Successfully
#   Time: 11572 ns Iteration: 0 Instance: /fec_tb
# ** Note: -----
#   Time: 11572 ns Iteration: 0 Instance: /fec_tb
# ** Note: -----Finished self checker for: FEC Block-----
#   Time: 11572 ns Iteration: 0 Instance: /fec_tb
# ** Note: -----
#   Time: 11572 ns Iteration: 0 Instance: /fec_tb
# ** Note: ----- Simulation Finished -----
#   Time: 11572 ns Iteration: 0 Instance: /fec_tb
# ** Note: -----
#   Time: 11572 ns Iteration: 0 Instance: /fec_tb
VSIM 40>
```

Dual Port Memory

(We would change its name from PPL_BLOCK to 2PORTSRAM in next phase).

Main File Name	PLL_main.vhd
Extra File	PLL_main_0002.v

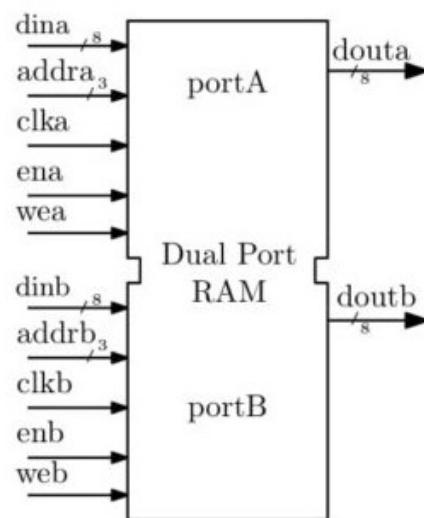
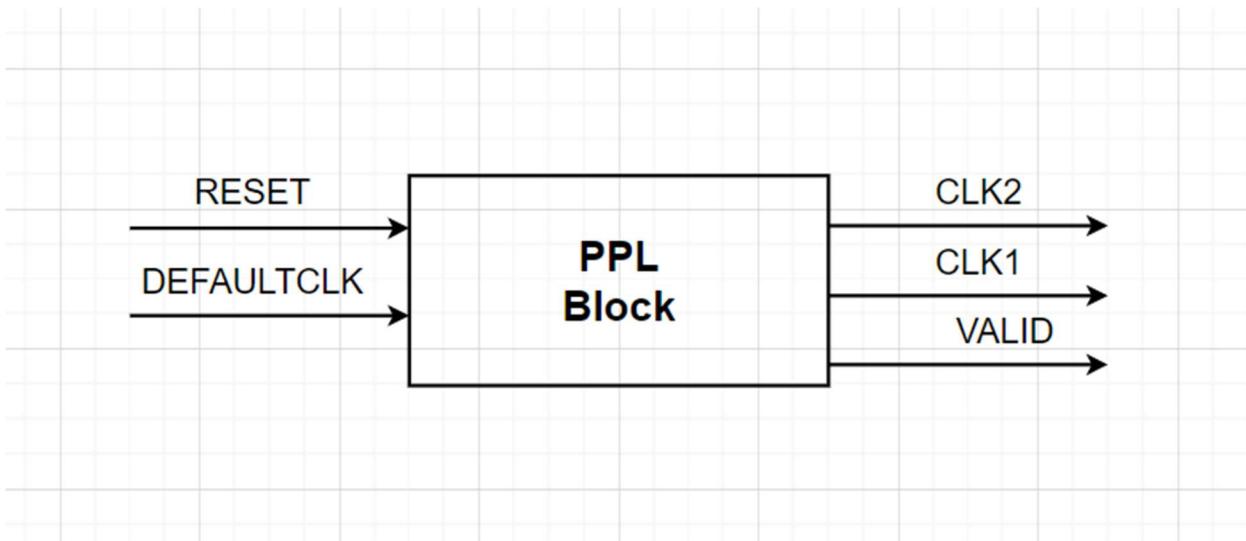


Fig.1

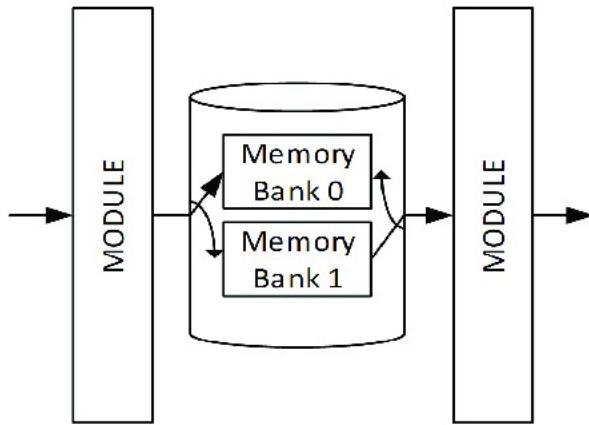


Fig 2.

Signals	In/Out	Type	Width
refclk	In	STD_LOGIC	1
rst	In	STD_LOGIC	1
outclk_0	Out	STD_LOGIC	1
outclk_1	Out	STD_LOGIC	1
locked	Out	STD_LOGIC	1

Functionality

PLL take 50Mhz clock as reference, and output two clocks.

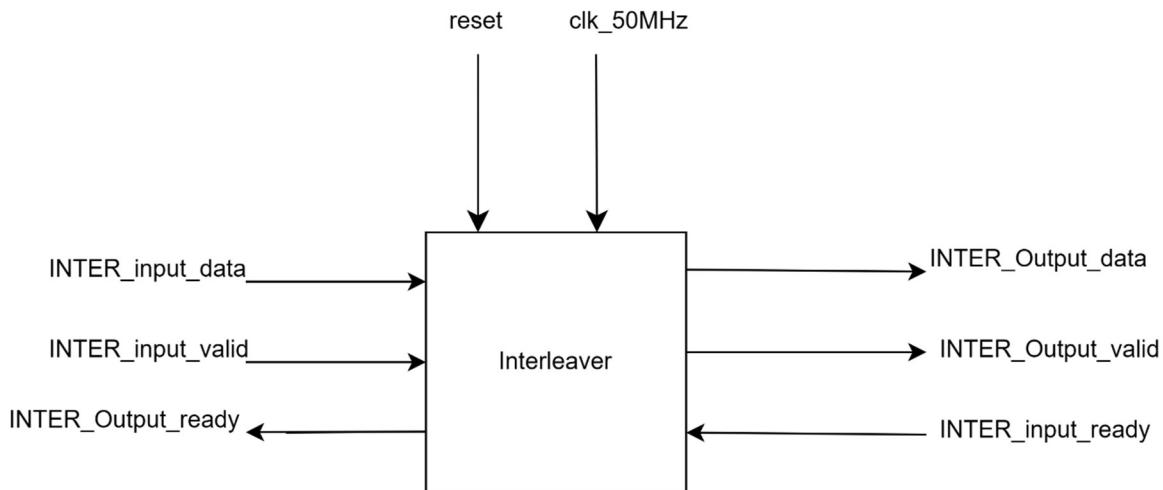
Interleaver

Main File Name	INTER.vhd
Testbench File Name	INTER_tb.vhd
Extra File 1	INTER_RAM_2port

Ports:

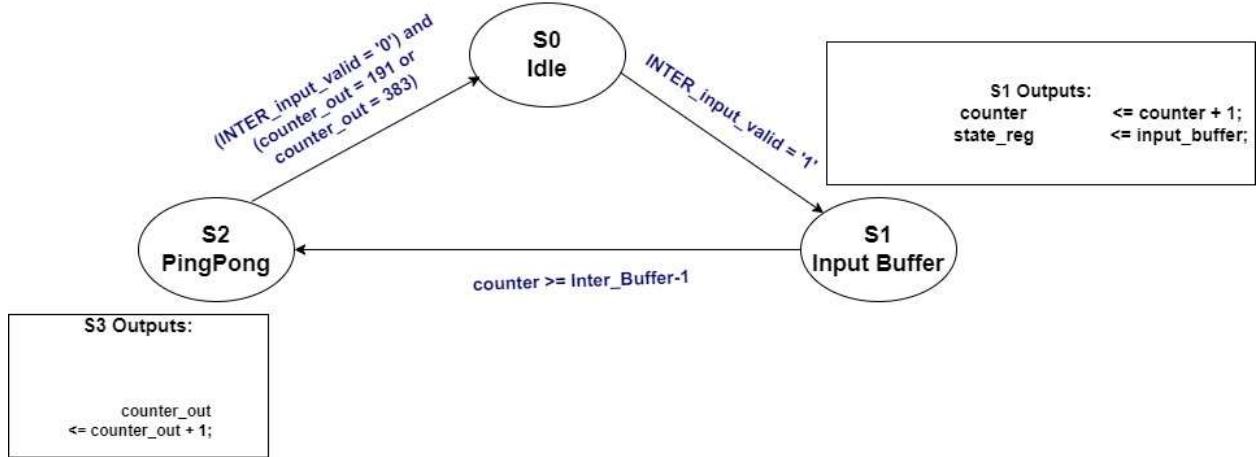
Signals	In/Out	Type	Width
INTER_Input_data	In	std_logic	1
INTER_Input_ready	In	std_logic	1
INTER_Input_valid	In	std_logic	1
Clk_100Mhz	In	std_logic	1
reset	In	std_logic	1
INTER_OUTPUT_data	Out	std_logic	1
INTER_OUTPUT_valid	out	std_logic	1
INTER_OUTPUT_ready	out	std_logic	1

Block Diagram Sketch:



Finite State Machine Diagram

INTER: Finite State Machine



Testing and Functionality

- 4 Input Streams, 2 us for each input. And 8 us for output streams, but shifted by 1 phase cycle. Thus, total run time is 10 us.
- For INTER, input data is Parallel, then wires permeate the input, then output it serially
- Total delay of the block is $n+2$ (where n is the number of cycles) ... and this two is mainly from shifting and loading.
- The test bench for INTER (INTER_tb.vhd) utilizes the package Test_Pack for the bit vectors the outputs are compared to ,and the built-in Alterna package.It also procedures for the INTER module.

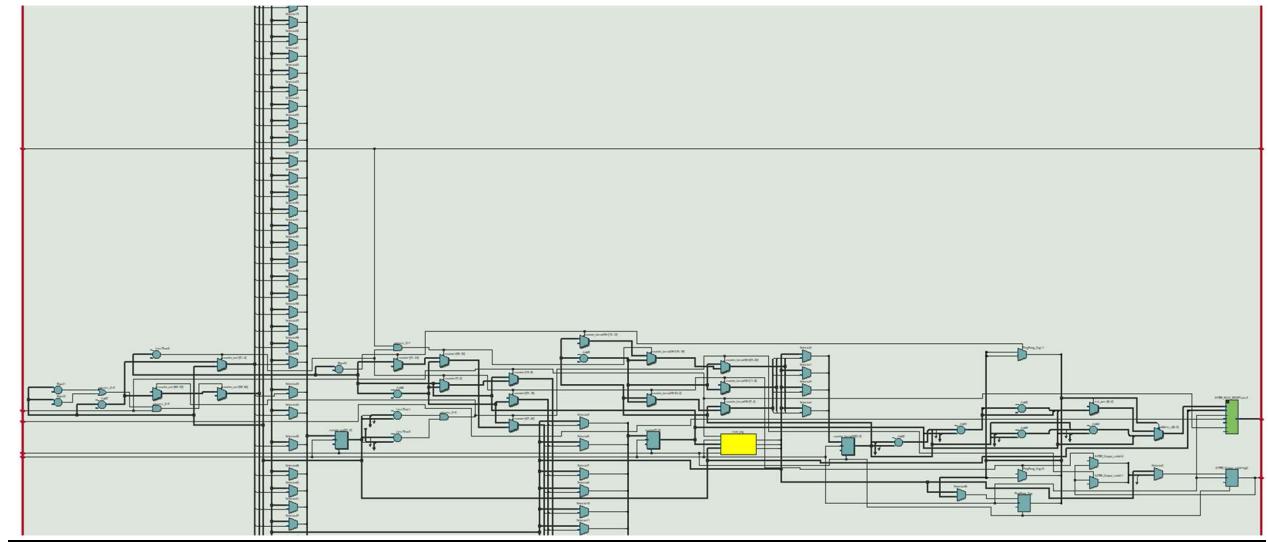
Packages used:

```
request_resolver.vhd    pingpongbuffer.vhd    randi.vhd    randi_tb.vhd    INTER.vhd    INTER_tb.vhd    Test_Pack.vhd    FEC.vhd    PLL.vhd
INTER_tb.vhd
1 library IEEE;
2 use IEEE.STD_LOGIC_1164.ALL;
3 use IEEE.NUMERIC_STD.ALL;
4 USE work.Test_Pack.ALL;
5
6 entity INTER_tb is
7 end INTER_tb;
8
9 architecture INTER_tb_rtl of INTER_tb is
10 component INTER
11 port(
12     clk_100MHz : in std_logic;
13     reset : in std_logic;
14     INTER_Input_data : in std_logic;
15     INTER_Input_valid : in std_logic;
16     INTER_Input_ready : in std_logic;
17     INTER_Output_data : out std_logic;
18     INTER_Output_valid : out std_logic;
19     INTER_Output_ready : out std_logic
20 );
21 end component;
22
23 --constants
24 constant CLK_100MHz_Period : Time := 10 ns;
25 constant CLK_100MHz_Period_HALF : Time := CLK_100MHz_Period / 2;
26
27 --signals
28 signal clk : std_logic := '0';
29 signal reset : std_logic;
30 signal INTER_Input_valid : std_logic;
31 signal INTER_Input_ready : std_logic;
32
33 signal test_in_vector : std_logic_vector(191 downto 0) := x"2833E48D39202605B6DC5E4AF47ADD2949486C89151348CA";
34 signal INTER_Input_data : std_logic;
35 signal INTER_Output_data : std_logic;
```

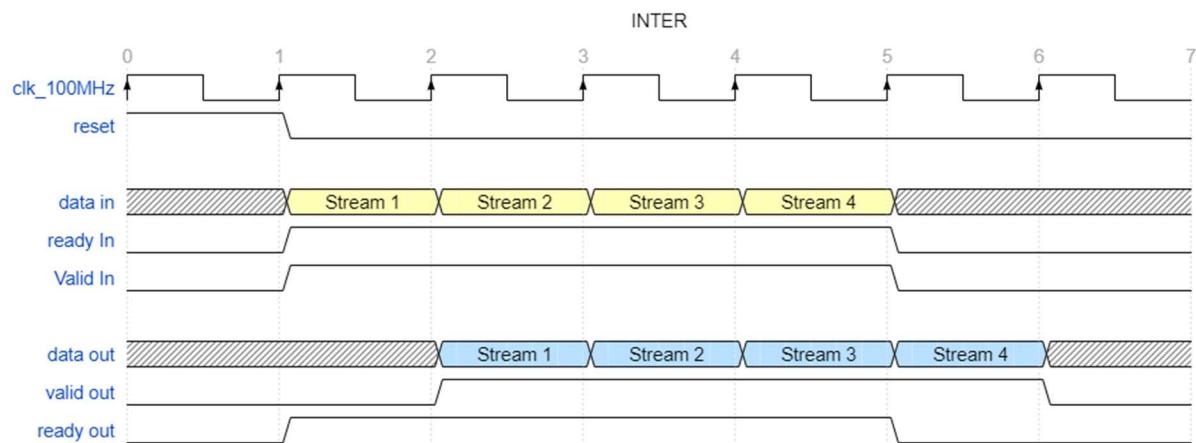
Procedures used:

RTL (Quartus)

1- RTL OF INTER (MAIN BLOCK)

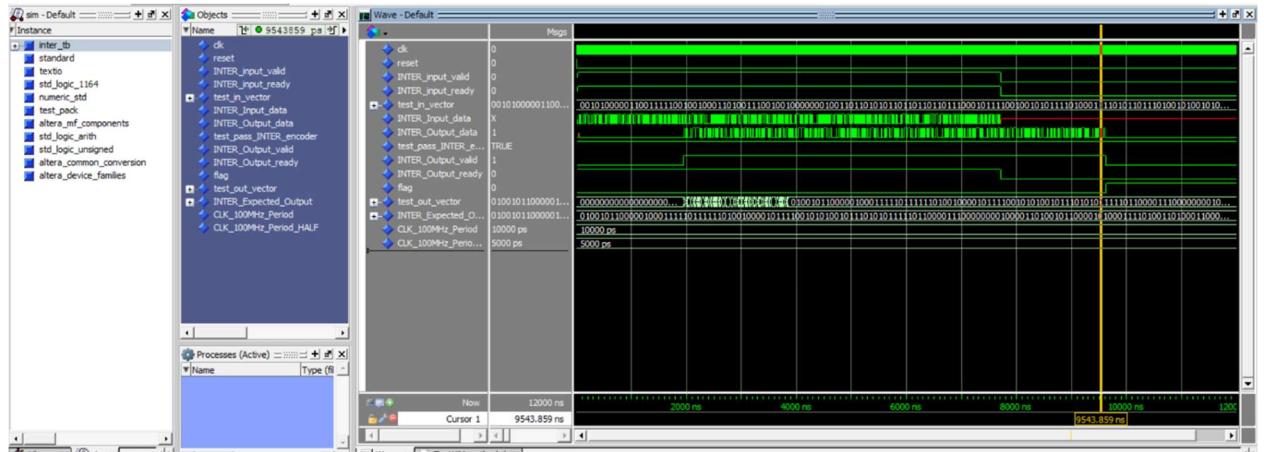


WaveForm (Sketch Expect)



Results (ModelSim Altera)

1- RESULTS INTER



2- INTER Self Check

The screenshot shows the 'Transcript' window of ModelSim Altera. It contains a log of simulation events and notes. Key entries include:

- Notes indicating the start of input streams: "## Note: ----- ## The Fourth INTER Input Stream", "## Note: ----- ## INTER Third Input Stream test passed Successfully", and "## Note: ----- ## INTER Second Input Stream test passed Successfully".
- A note about finishing the fourth input stream: "## Note: ----- ## Done Inputting the Fourth stream".
- A note about finishing self-checking: "## Note: ----- Finished self checker for: INTER Block".
- A note about simulation completion: "## Note: ----- Simulation Finished".
- Final notes at the end of the transcript: "## Note: ----- ## Simulation Finished", "## Note: ----- ## Simulation Finished", and "## Note: ----- ## Simulation Finished".

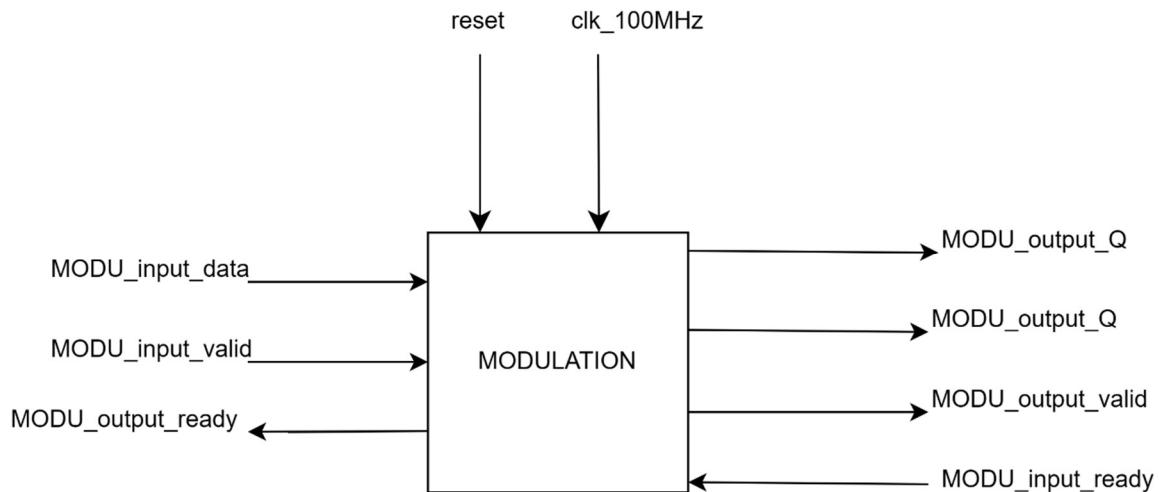
Modulation

Main File Name	MODU.vhd
Testbench File Name	MODU_tb.vhd

Ports:

Signals	In/Out	Type	Width
MODU_input_data	In	std_logic	1
MODU_input_ready	In	std_logic	1
MODU_input_valid	In	std_logic	1
clk_100MHz	In	std_logic	1
clk_50MHz	In	std_logic	1
reset	In	std_logic	1
MODU_output_valid	Out	std_logic	1
MODU_output_ready	Out	std_logic	1
MODU_output_Q	Out	std_logic_vector	16
MODU_output_I	Out	std_logic_vector	16

Block Diagram Sketch:



Testing and Functionality

- 2 inputs streams. Total runtime of around 4 us.

- Again, we use 100MHz for output and 50MHz for inputs.
- For handshaking, we have some delay to allow data to arrive as serial inputs (let's say late by x cycles).
- The total time needed is $n + x$ (where n might be calculated as $192*8$).
- The test bench The test bench for MODULATION (MODU_tb.vhd) utilizes the package Test_Pack for the bit vectors the outputs are compared to .It also has procedures for the MODU module

Package used:

```

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.NUMERIC_STD.ALL;
use work.Test_Pack.ALL;

entity MODU_tb is
end MODU_tb;

architecture MODU_tb_rtl of MODU_tb is
component MODU
PORT (
    clk_100MHz : IN std_logic;
    reset : IN std_logic;
    MODU_input_data : IN std_logic;
    MODU_input_valid : IN std_logic;
    MODU_input_ready : IN std_logic;
    MODU_output_Q : OUT std_logic_vector(15 DOWNTO 0);
    MODU_output_I : OUT std_logic_vector(15 DOWNTO 0);
    MODU_output_valid : OUT std_logic;
    MODU_output_ready : OUT std_logic
);
end component;

signal clk_100MHz : std_logic := '0';
signal reset : std_logic;
signal MODU_input_valid : std_logic;
signal MODU_input_ready : std_logic;
signal test_in_vector : std_logic_vector(191 DOWNTO 0) := MODU_VECTOR_INPUT;
signal test_out_vector_Q : std_logic_vector(1535 DOWNTO 0) := (others => '0');
signal test_out_vector_I : std_logic_vector(1535 DOWNTO 0) := (others => '0');
signal MODU_input_data : std_logic;
signal MODU_output_Q : std_logic_vector(15 DOWNTO 0);
signal MODU_output_I : std_logic_vector(15 DOWNTO 0);
signal MODU_output_valid : std_logic;

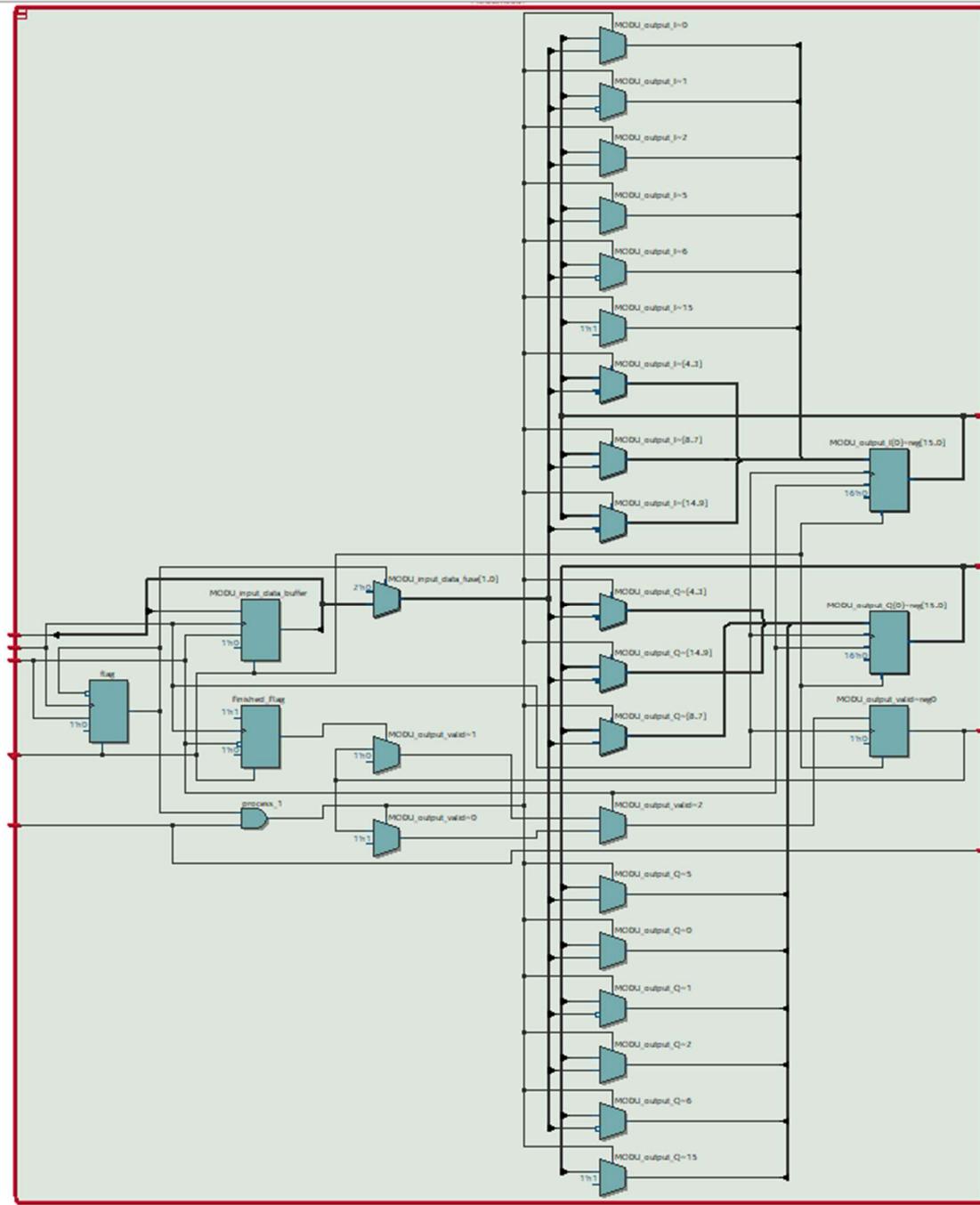
```

Procedure(s) used:

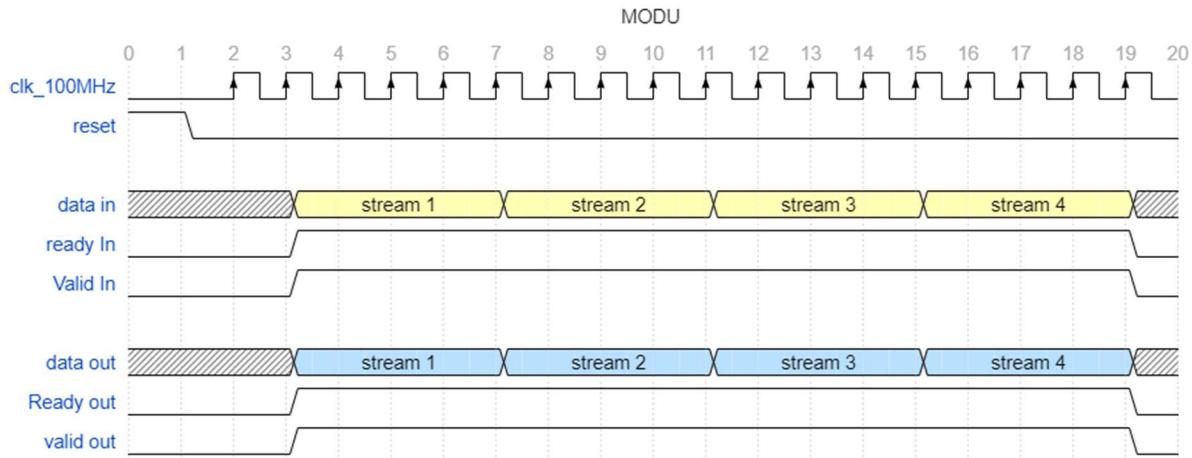
The screenshot shows a VHDL simulation environment with the following details:

- File Explorer:** Shows files under the `vs` folder, including `FEC_tb.vhd`, `FEC_tb.vhd`, `MODU_tb.vhd`, `elevator_ctrl.vhd`, `request_resolver.vhd`, `pingpongbuffer.vhd`, `randi.vhd`, `randi_tb.vhd`, `INTER.vhd`, `INTER_tb.vhd`, and `MODU_tb.vhd`.
- Code Editor:** The `MODU_tb.vhd` file is open, displaying VHDL code for a testbench. The code includes a process for serial input, a report procedure for break notices, and a report section for MODU input streams.
- Waveform View:** A large window on the right displays the waveform for the `MODU_tb.vhd` file, showing multiple signals over time.
- Bottom Bar:** Includes status indicators for line 84, column 66, spaces 4, UTF-8, CRLF, and various icons for file operations and help.

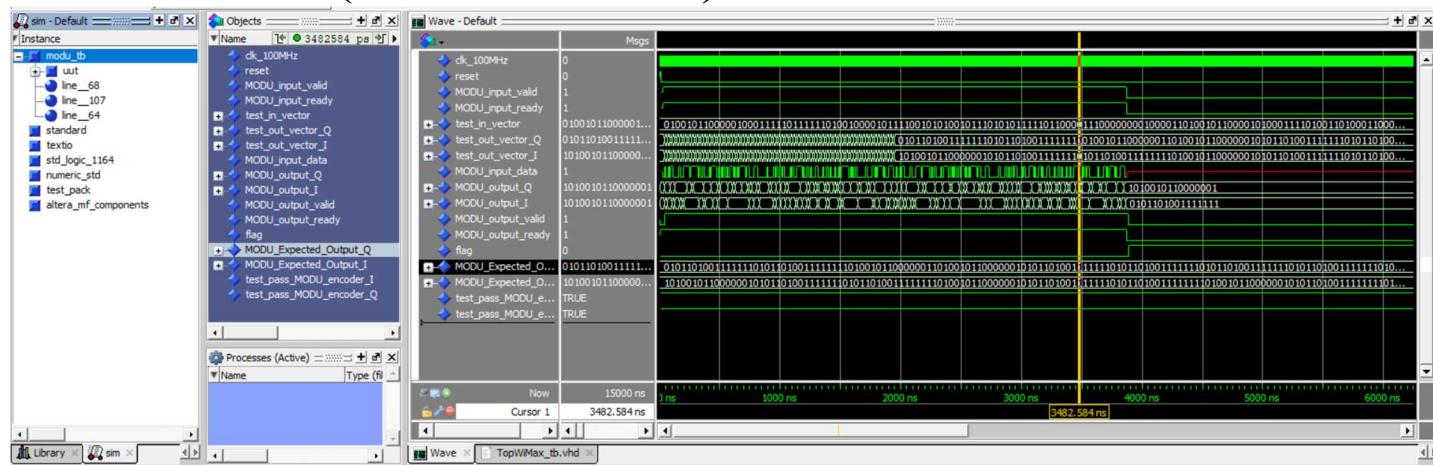
RTL (Quartus)



Waveform (Sketch Expect)



MODU Results (ModelSim Altera)



MODU Self Check (ModelSim Altera)

The screenshot shows the ModelSim Altera interface with the following details:

- Menu Bar:** File, Edit, View, Compile, Simulate, Add, Structure, Tools, Layout, Bookmarks, Window, Help.
- Toolbars:** Standard toolbar with icons for file operations, zoom, and simulation controls.
- Object Bars:** ColumnLayout, Objects, Wave.
- Simulator View:** sim - Default, Objects, Wave - Default.
- Library View:** Library, sim.
- Transcript Window:** Displays the simulation log output.
- Log Output:** The transcript window shows the following log entries:

```
# ** Note: ----- ### Ended Checking MODU First Output stream
# Time: 1966536 ps Iteration: 0 Instance: /modu_tb
# ** Note: -----
# Time: 1966536 ps Iteration: 0 Instance: /modu_tb
# ** Note: ----- ### Started Checking MODU Second Output stream:
# Time: 1966536 ps Iteration: 0 Instance: /modu_tb
# ** Note: ----- ## Done Inputting MODU the Second stream:
# Time: 3870 ns Iteration: 0 Instance: /modu_tb
# ** Note: ----- Finishehd Inputting MODU (2) Input Streams -----
# Time: 3870 ns Iteration: 0 Instance: /modu_tb
# ** Note: -----
# Time: 3870 ns Iteration: 0 Instance: /modu_tb
# ** Note: ----- ## Second MODU Output Stream: 1536 Q Values Successed
# Time: 3888072 ps Iteration: 0 Instance: /modu_tb
# ** Note: -----
# Time: 3888072 ps Iteration: 0 Instance: /modu_tb
# ** Note: ----- ## Second MODU Output Stream: 1536 I Values Successed
# Time: 3888072 ps Iteration: 0 Instance: /modu_tb
# ** Note: -----
# Time: 3888072 ps Iteration: 0 Instance: /modu_tb
# ** Note: ----- ## Ended Checking MODU Second Output stream
# Time: 3888072 ps Iteration: 0 Instance: /modu_tb
# ** Note: -----
# Time: 3888072 ps Iteration: 0 Instance: /modu_tb
# ** Note: ----- Ended MODU Checking (Demodulating) (2) output streams -----
# Time: 3888072 ps Iteration: 0 Instance: /modu_tb
# ** Note: -----
# Time: 3888072 ps Iteration: 0 Instance: /modu_tb
# ** Note: ----- Simulation Finished -----
# Time: 3888072 ps Iteration: 0 Instance: /modu_tb
# ** Note: -----
# Time: 3888072 ps Iteration: 0 Instance: /modu_tb
```

VSIM 50>

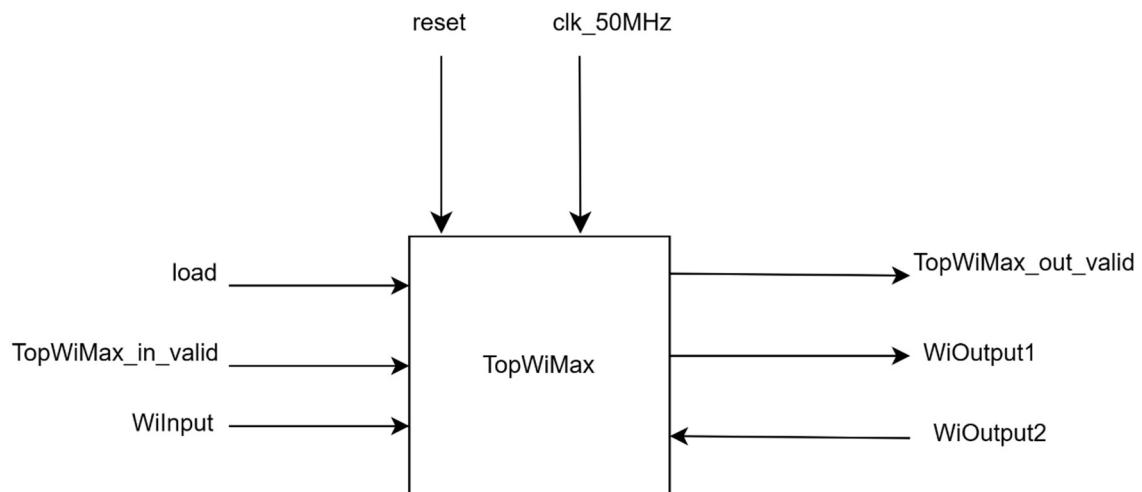
TopWiMax

Main File Name	TopWiMax.vhd
Testbench File Name	TopWiMax_tb.vhd

Ports:

Signals	In/Out	Type	Width
Wilnput	In	std_logic	1
TopWiMax_in_ready	In	std_logic	1
TopWiMax_in_valid	In	std_logic	1
load	In	std_logic	1
clk_50MHz	In	std_logic	1
reset	In	std_logic	1
TopWiMax_out_valid	Out	std_logic	1
TopWiMax_out_ready	Out	std_logic	1
WiOutput1	Out	std_logic_vector	16
WiOutput2	Out	std_logic_vector	16

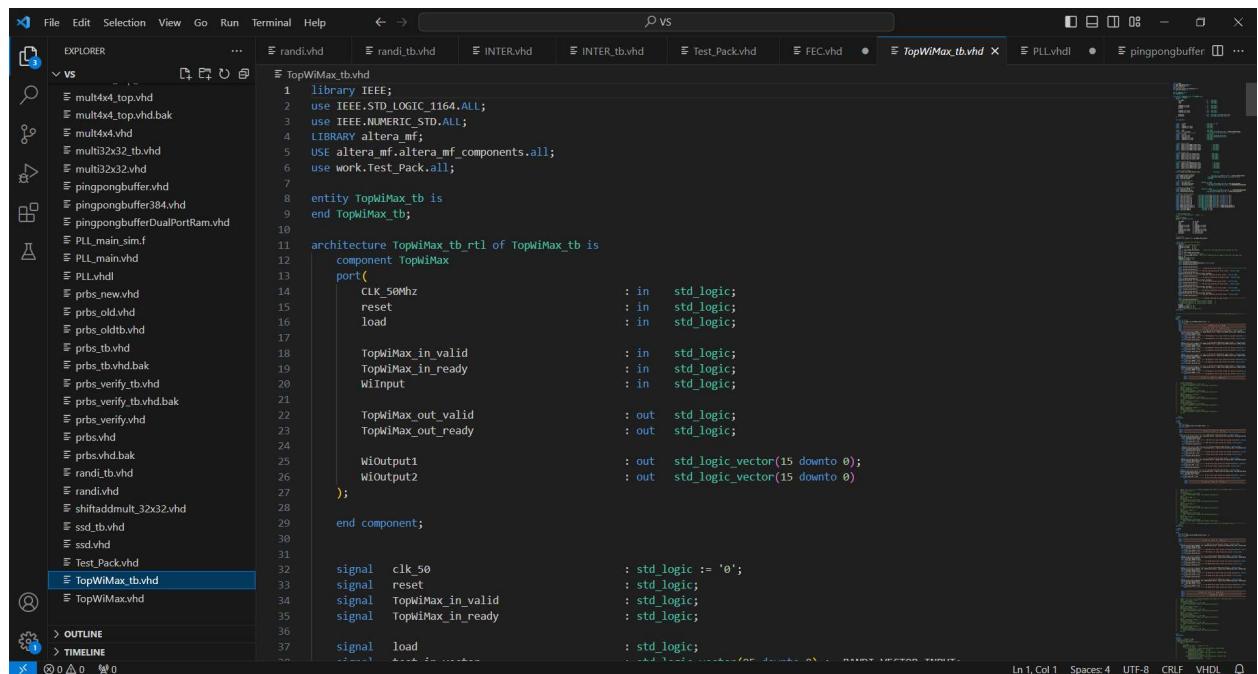
Block Diagram Sketch:



Testing and Functionality

- 2 inputs streams. Total runtime of around 4 us.
- Again, we use 100MHz for output and 50MHz for inputs.
 - For handshaking, we have some delay to allow data to arrive as serial inputs (let's say late by x cycles).
 - The total time needed is n +x (where n might be calculated as 192*8).
- The test bench uses Test_Pack package,built-in Alterna package , procedures and external names.

Packages used:



```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.NUMERIC_STD.ALL;
LIBRARY altera_mf;
USE altera_mf.altera_mf_components.all;
use work.Test_Pack.all;

entity TopWiMax_tb is
end TopWiMax_tb;

architecture TopWiMax_tb_rtl of TopWiMax_tb is
component TopWiMax
port(
    CLK_50MHz : in std_logic;
    reset : in std_logic;
    load : in std_logic;
    TopWiMax_in_valid : in std_logic;
    TopWiMax_in_ready : in std_logic;
    WiInput : in std_logic;
    TopWiMax_out_valid : out std_logic;
    TopWiMax_out_ready : out std_logic;
    WiOutput1 : out std_logic_vector(15 downto 0);
    WiOutput2 : out std_logic_vector(15 downto 0)
);
end component;

signal clk_50 : std_logic := '0';
signal reset : std_logic;
signal TopWiMax_in_valid : std_logic;
signal TopWiMax_in_ready : std_logic;
signal load : std_logic;
```

Procedural(s) used:

```

File Edit Selection View Go Run Terminal Help < - > OS vs
EXPLORER ... randi.vhd randi_tb.vhd INTER.vhd INTER_tb.vhd Test_Pack.vhd FEC.vhd TopWiMax_tb.vhd PLLvhdl pingpongbuffer ...
mult4x4_top.vhd 268
mult4x4_top.vhd.bak 269
mult4x4.vhd 270
multi32x32_tb.vhd 271
multi32x32.vhd 272
pingpongbuffer.vhd 273
pingpongbuffer384.vhd 274
pingpongbufferDualPortRam.vhd 275
PLL_main_sim.f 277
PLL_main.vhd 278
PLLvhdl 279
prbs_new.vhd 280
prbs_old.vhd 281
prbs_cldtb.vhd 282
prbs_tb.vhd 284
prbs_tb.vhd.bak 285
prbs_verify_tb.vhd 286
prbs_verify_tb.vhd.bak 287
prbs_verify.vhd 288
prbs.vhd 290
prbs.vhd.bak 291
randi_tb.vhd 292
randi.vhd 293
shiftaddmult_32x32.vhd 294
ssd_tb.vhd 295
ssd.vhd 296
Test_Pack.vhd 298
TopWiMax_tb.vhd 299
TopWiMax.vhd 300
301
302
303
304
report "----- Started self checker for: FEC Block -----";
report "----- The First FEC Input Stream " severity note;
procedure_192_outputs_FEC(0, 191, FEC_Output_Vector, signal_alias_fec_output_data, FEC_Expected_Output, test_pass_FEC);
report procedure_Break_Note;
assert test_pass_FEC = false
| report "----- FEC First Input Stream test passed Successfully" severity note ;
assert test_pass_FEC = true
| report "----- FEC First Input Stream test Failed" severity note ;
report procedure_Break_Note;

report "----- The Second FEC Input Stream " severity note;
procedure_192_outputs_FEC(0, 191, FEC_Output_Vector, signal_alias_fec_output_data, FEC_Expected_Output, test_pass_FEC);
report procedure_Break_Note;
assert test_pass_FEC = false
| report "----- FEC Second Input Stream test passed Successfully" severity note ;
assert test_pass_FEC = true
| report "----- FEC Second Input Stream test Failed" severity note ;
report procedure_Break_Note;

report "----- The Third FEC Input Stream " severity note;
procedure_192_outputs_FEC(0, 191, FEC_Output_Vector, signal_alias_fec_output_data, FEC_Expected_Output, test_pass_FEC);
report procedure_Break_Note;
assert test_pass_FEC = false
| report "----- FEC Third Input Stream test passed Successfully" severity note ;
assert test_pass_FEC = true
| report "----- FEC Third Input Stream test Failed" severity note ;
report procedure_Break_Note;

report "----- The Fourth FEC Input Stream " severity note;
procedure_192_outputs_FEC(0, 191, FEC_Output_Vector, signal_alias_fec_output_data, FEC_Expected_Output, test_pass_FEC);
report procedure_Break_Note;
assert test_pass_FEC = false
| report "----- FEC Fourth Input Stream test passed Successfully" severity note ;
assert test_pass_FEC = true
| report "----- FEC Fourth Input Stream test Failed" severity note ;
report procedure_Break_Note;

```

Ln 1, Col 1 Spaces: 4 UTF-8 CRLF VHDL

External names:

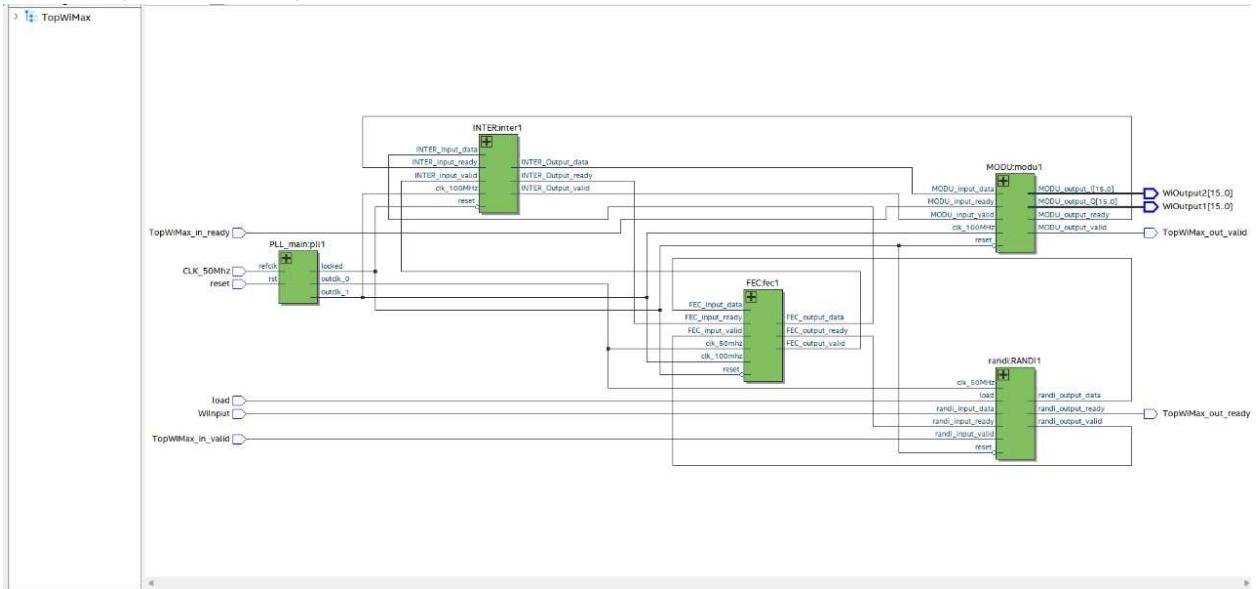
```

File Edit Selection View Go Run Terminal Help < - > OS vs
EXPLORER ... randi.vhd randi_tb.vhd INTER.vhd INTER_tb.vhd Test_Pack.vhd FEC.vhd TopWiMax_tb.vhd PLLvhdl pingpongbuffer ...
mult4x4_top.vhd 31
mult4x4_top.vhd.bak 32
mult4x4.vhd 33
multi32x32_tb.vhd 34
multi32x32.vhd 35
pingpongbuffer.vhd 36
pingpongbuffer384.vhd 37
pingpongbufferDualPortRam.vhd 38
PLL_main_sim.f 39
PLL_main.vhd 40
PLLvhdl 41
prbs_new.vhd 42
prbs_old.vhd 43
prbs_cldtb.vhd 44
prbs_tb.vhd 45
prbs_tb.vhd.bak 46
prbs_verify_tb.vhd 47
prbs_verify_tb.vhd.bak 48
prbs_verify.vhd 49
prbs.vhd 50
prbs.vhd.bak 51
randi_tb.vhd 52
randi.vhd 53
shiftaddmult_32x32.vhd 54
ssd_tb.vhd 55
ssd.vhd 56
Test_Pack.vhd 57
TopWiMax_tb.vhd 58
TopWiMax.vhd 59
60
61
62
63
64
65
66
67
signal clk_50 : std_logic := '0';
signal reset : std_logic;
signal TopWiMax_in_valid : std_logic;
signal TopWiMax_in_ready : std_logic;
signal load : std_logic;
signal test_in_vector : std_logic_vector(95 downto 0) := RANDI_VECTOR_INPUT;
-- signal demodulation_vector : std_logic_vector(191 downto 0) := (others => '0');
signal WiInput : std_logic;
signal test_out1_bit : std_logic_vector(15 downto 0) ;
signal test_out2_bit : std_logic_vector(15 downto 0) ;
signal TopWiMax_out_valid : std_logic;
signal TopWiMax_out_ready : std_logic;
signal TopWiMax_in_ready : std_logic;
-- alias signals RNDI
signal signal_alias_RANDI_output_data : std_logic;
signal signal_alias_RANDI_output_valid : std_logic;
signal signal_alias_RANDI_input_valid : std_logic;
signal signal_alias_RANDI_output_ready : std_logic;
signal signal_alias_RANDI_input_ready : std_logic;
-- alias signals FEC
signal signal_alias_fec_output_data : std_logic;
signal signal_alias_fec_output_valid : std_logic;
signal signal_alias_fec_input_valid : std_logic;
signal signal_alias_fec_output_ready : std_logic;
signal signal_alias_fec_input_ready : std_logic;
-- alias signals INTER
signal signal_alias_INTER_output_data : std_logic;
signal signal_alias_INTER_output_valid : std_logic;
signal signal_alias_INTER_input_valid : std_logic;
signal signal_alias_INTER_output_ready : std_logic;
signal signal_alias_INTER_input_ready : std_logic;

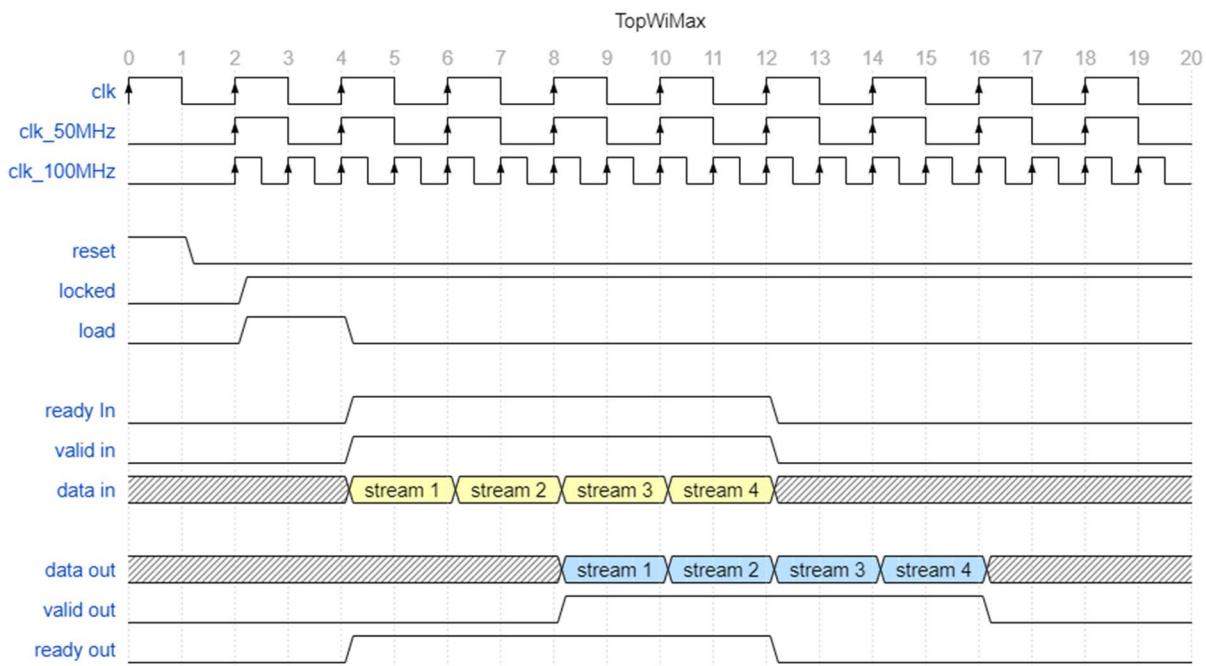
```

Ln 1, Col 1 Spaces: 4 UTF-8 CRLF VHDL

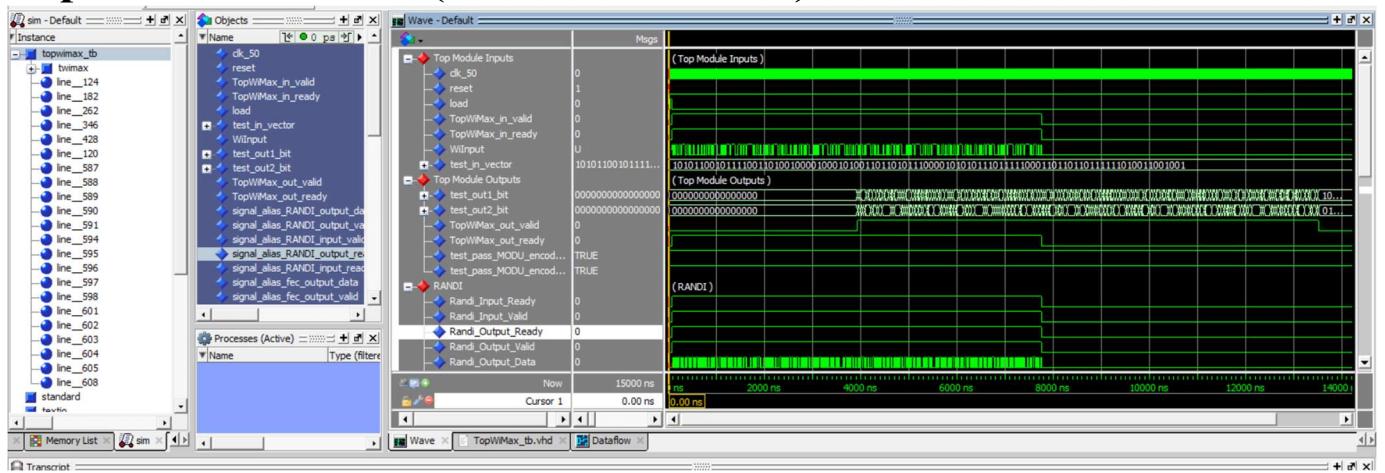
RTL (Quartus)

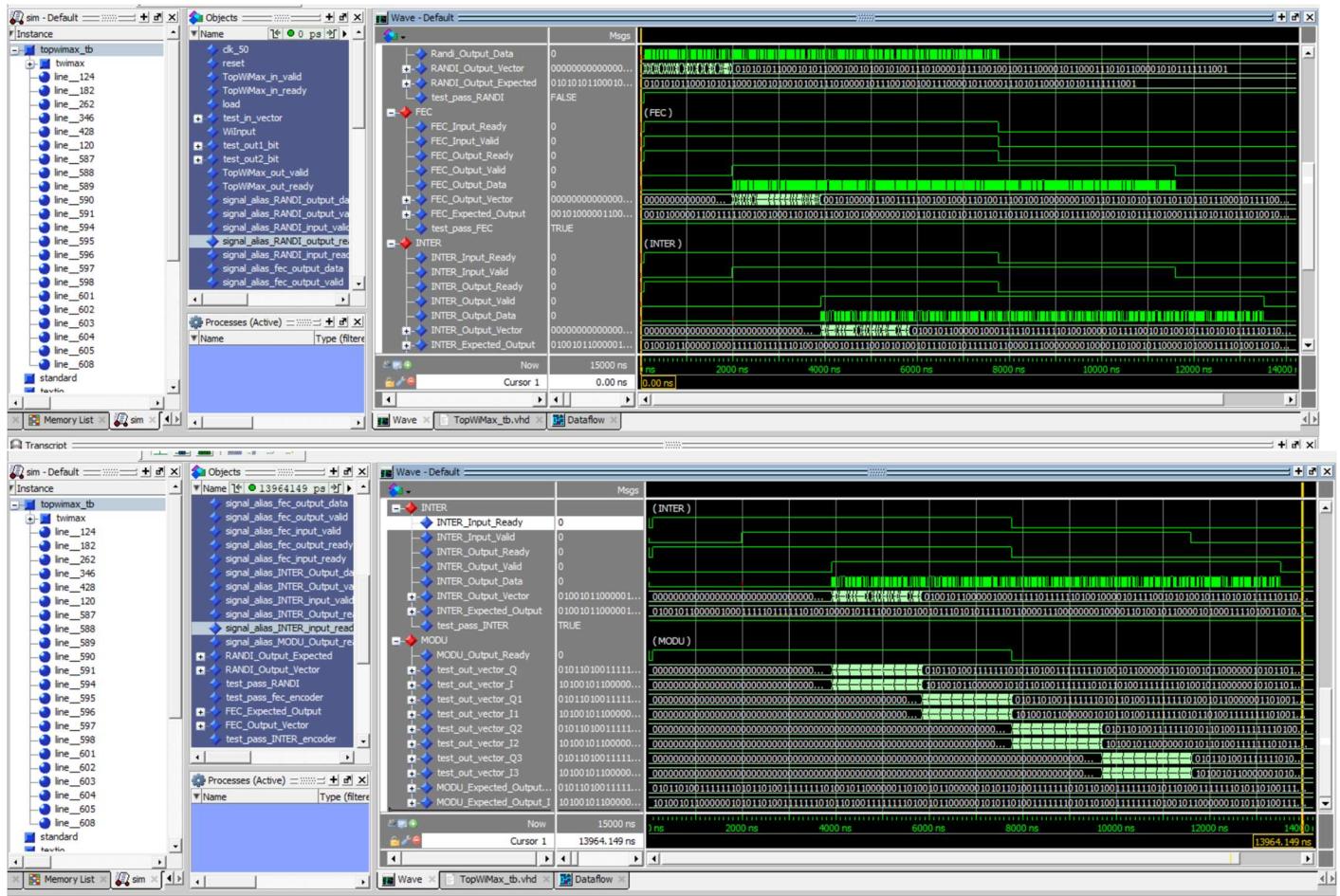


Waveform (Sketch Expect)



Top Module Results (ModelSim Altera)





MODU Self Check (ModelSim Altera)

MODU Self Check (ModelSim Altera) (With Error Injected)