The MLDesigner Optical Components Library

1. Introduction

The MLDesigner Optical Components Library is a collection of Discrete Event Data Structures and building blocks for the design and performance analysis of optical networking components and systems. The blocks can be combined to create models of optical network components (e.g., switches) and add optical interfaces to computer and server modules. The resulting models can be combined with upper layer protocol modules to do complete end-to-end optical network models.

The MLD Optical Components Library includes special optical library data structures, low level building blocks (e.g. laser sources, couplers, switching elements), high level building blocks (e.g., MUXers and DEMUXers, switches, inverters) and some sample optical system models (e.g., switch and switchless TDM and WDM Pixel Bus systems.

Physical layer models developed with the Optical Components Library can be extended with components from other MLDesigner libraries to model all layers of the OSI stack.

2. Modeling Approach

This library uses data structures to abstract component features so they are manageable in a network-level simulation while retaining sufficient physical effect detail to maintain accuracy.

Data structures are the foundation of the optical components library. All optical signals that pass through components in the optical network library are first and foremost a member of the *Optical Layer* (*OL*) data structure. This class definition is used to separate optical signals from others that may be used in the simulation (e.g. electrical, wireless, logical).

Optical signals are further classified into one of two types in this library version. The *Single_OL* class is used for optical signals that are represented as a single wavelength within optical components (disregarding center frequency spread),. The data members of this class include *Wavelength*, *Opt_Power_Level*, *Data* and *Number_Of_Bits*.

Optical signals of different wavelengths that pass through a component at the same time are classified as wave-division multiplexed (WDM) signals and are members of the *Multiple_OL* class. The Multiple_OL class is composed of a vector of *Single_OL* members.

Library components current model only simple physical layer effects such as time delay and signal power-level attenuation or amplifications. Future versions of the library will add the capability to distinguish between signal power and noise (so an optical signal-to-noise ratio can be calculated) as well as additional noise sources. Ongoing research is investigating techniques to modeling higher-order physical effects such as amplified spontaneous emissions, crosstalk, dispersion, temperature effects, source chirping and 4-wave mixing.

3. Sample System Models

Here are some top-level block diagrams of systems models created with the Optical Component Library. More detailed descriptions of three systems models are provided in Section 5 at the end of this document.

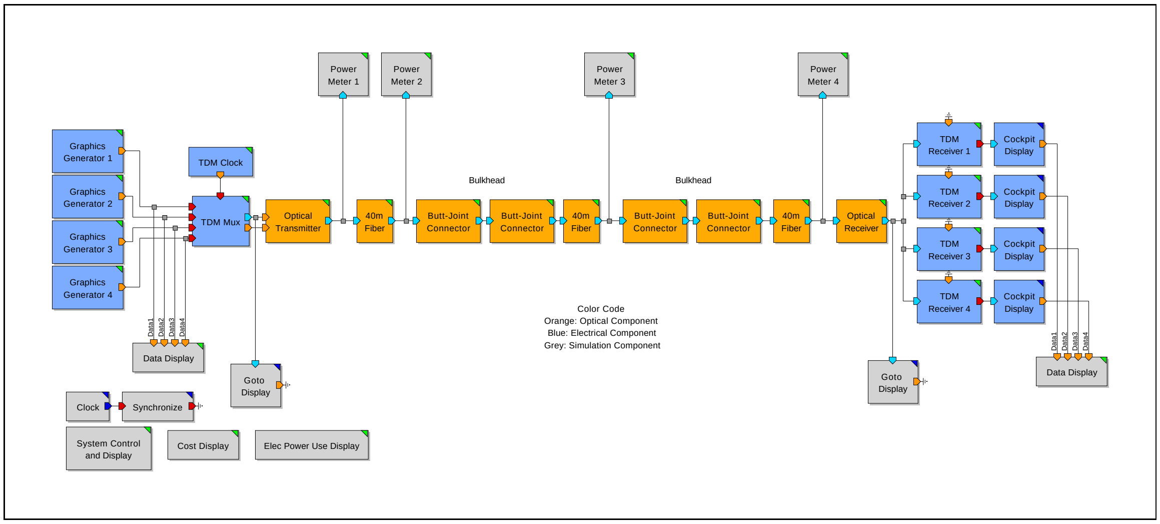

Figure 1: TDM Switchless Pixel Bus with •4 channels @ 2.5Gbps (10Gbps aggregate)

Figure 2: A Virtual prototype control panel to operate the TDM system model--assembled from MLDesigner TclTk animation component

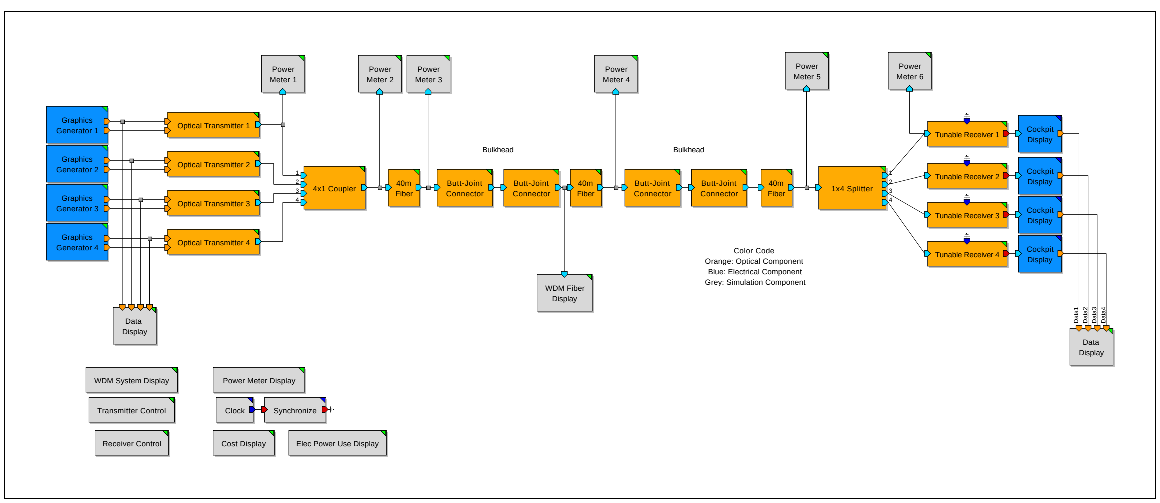

Figure 3: WDM Switchless Pixel; Bus •4 channels (4 independent wavelengths)@ 10Gbps (40Gbps aggregate)

Figures 2 and 4 show how the optical system models shown in Figures 1 & 3 can be adapted to serve as virtual prototypes to demonstrate the effects of changing parameters such as transmitter rate launch power, loss, power threshold on system operation. Readouts also show power meter readings from probes inserted into the network. The graphic control and monitoring panels were assembled with TclTk animation, dynamic control and dynamic output components from a prototype MLDesigner component library.

Figure 4: A virtual prototype control panel to operate the WDM model shown in Figure 3.

4. Detailed Description of Library Components

The complete library includes the data structures, low-level building blocks (internals), high level building blocks (components) and sample system models.

Data structures are hierarchical entities containing one or more data fields that can serve as a particle to carry data between blocks.

The low level building blocks are building blocks representing either primitives or block diagrams. A primitive is block that contains a coded algorithm with defined inputs and outputs .
A block diagram is a graphical representation of a functional model assembled from multiple blocks that are connected by net objects.

The high-level building blocks are all modules—blocks that contain block diagrams. They are the building blocks combined to form systems models.

Systems models are self-contained executable models/block diagrams. The contain sources to start the model and sinks to collect and display model outputs.

Top-level block diagrams are shown for the components and system models.

4.1 Data Structures

This section provides a brief description of the library data structures. Detailed descriptions are provided in the online documentation.

Root.OpticalLayer
- ID Number - ID number of the data structure
- Creation time - Time stamp showing ds creation time

OpticalLayer. Single_ OL
- ID Number - inherited from Root
- Creation Time - inherited from Root
- Data - Data in this packet
- Number_Of_Bits - Number of bits for data
- Wavelength - Wavelength of the single optical signal of this ds.
- Opt_Power_Level - Optical power level of this single wavelength of the ds

OpticalLayer. WDM_ OL
- ID Number - inherited from Root
- Creation Time - inherited from Root
- Active_Waves - A vector of Single_OL data structures. (A WDM_OL consists of multiple Single_OL data structures combined into a single ds.)

Power Selection
- Index 0 - Used to define power selection in dB.
- Index 1 - Used to define power selection as a percentage from an input parameter.

Switch_Control
- Switch type - Defines the switch dimensions
- Control_Vector - A vector with N elements where N is the number of input ports on the designated switch. Each vector elements is an integer from 1 to M. M is equal to the number of output ports on the switch. The first vector el ement (index=0) contains the number of the output port to which input port 1 will be connected. The second vector element (index=1, if present) designates which number output the second input port is connected to. The pattern continues for any other input pots. Note: All input ports must be assigned a unique output port; multicasting is not supported.

Data Integer (- inf, inf) 0 Data in this packet on optical layer

Number of bits for data

Wavelength of single optical signal of this structure

Optical power level of single wavelength of this structure

4.2 Internals

1x2 Optical Switch Primitive in DE Domain

The function of this primitive is to pass an Input DS to one of the two output ports. The value of the "control" input determines which output is to be enabled. If the value of the "control" input is zero, the input DS is placed on the "falseOut" output port. The input DS is placed on the "trueOut" output port for non-zero values of the "control" input.

Inputs

anytype (DS to be placed on one of the two output ports)
control

Outputs

output1
output2

1x4_Logical_Switch primitive in DE domain

The function of this primitive is to pass an Input DS to one of the two output ports. The value of the "control" input determines which output is to be enabled. If the value of the "control" input is zero, the input DS is placed on the "falseOut" output port. The input DS is placed on the "trueOut" output port for non-zero values of the "control" input.

Inputs

input
control

Outputs

output1
output2
output3
output
Ctrl_error

1x4_Splitting_Module module in DE domain

This module splits optical signal either single or WDM OL from one input port (OL_ In) to four output ports (OL_ Out_ 1: OL_ Out_ 4). The power level of each output channel is defined by individual splitting ratio (CH1_ Ratio: CH4_ Ratio).

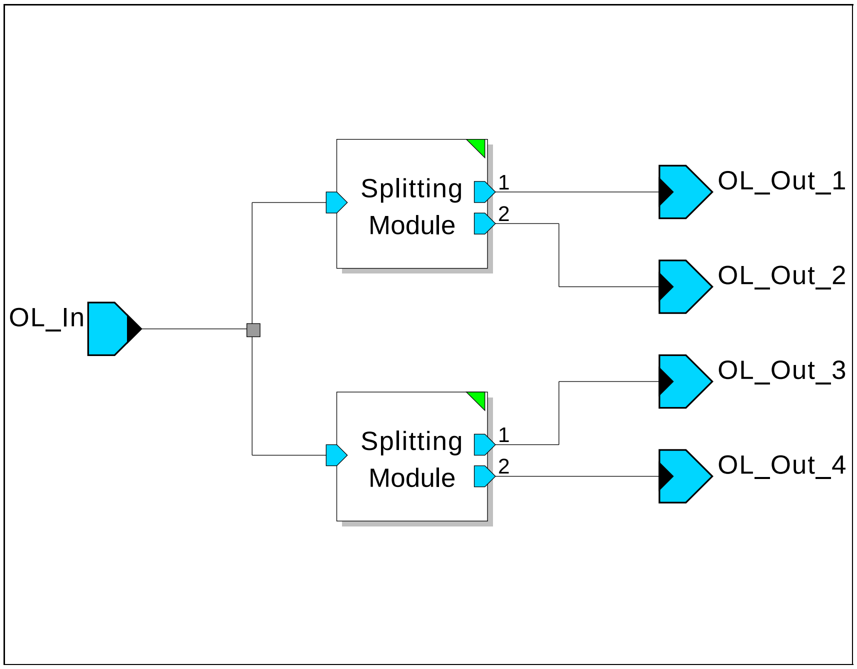

Figure 5: 1X4 Splitting module

Inputs
OL_In

Outputs
OL_Out_1
OL_Out_2
OL_Out_3
OL_Out_4

Parameters
CH1_Ratio
CH2_Ratio
CH3_Ratio
CH4_Ratio

1x8_Splitting_Module module in DE domain

This module splits optical signal either single or WDM OL from one input port (OL_ In) to eight output ports (OL_ Out_ 1: OL_ Out_ 8). The power level of each output channel is defined by individual splitting ratio (CH1_ Ratio: CH8_ Ratio).

Inputs
OL_In

Outputs
OL_Out_1

OL_Out_2
OL_Out_3
OL_Out_4
OL_Out_5
OL_Out_6
OL_Out_7
OL_Out_8

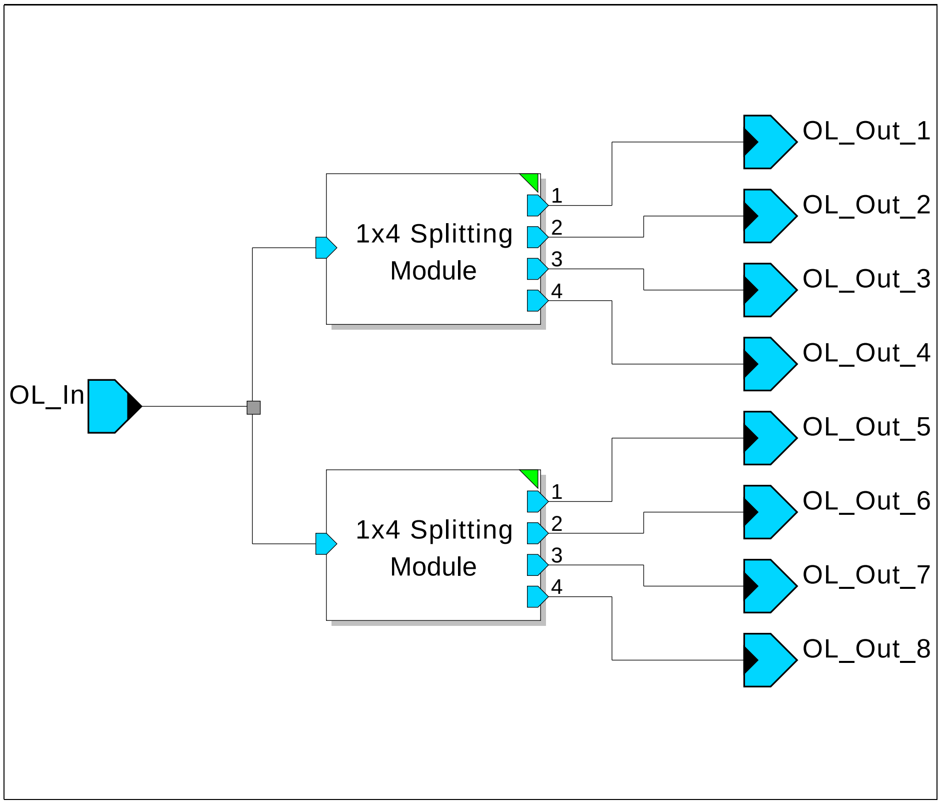

Figure 6: 1X8 Splitting Module

Parameters
CH1_Ratio
CH2_Ratio
CH3_Ratio
CH4_Ratio
CH5_Ratio
CH6 Ratio
CH7Ratio
CH8_Ratio

<u>2x1_Coupler_Module module in DE domain</u>

This module couples two input optical signals to WDM output signal. Input signal can be either single or WDM OL but output signal is always WDM OL. If only one signal at input port is received at particular time, this signal can pass through output without coupling. Each channel has coupling ratio which can be defined individually.

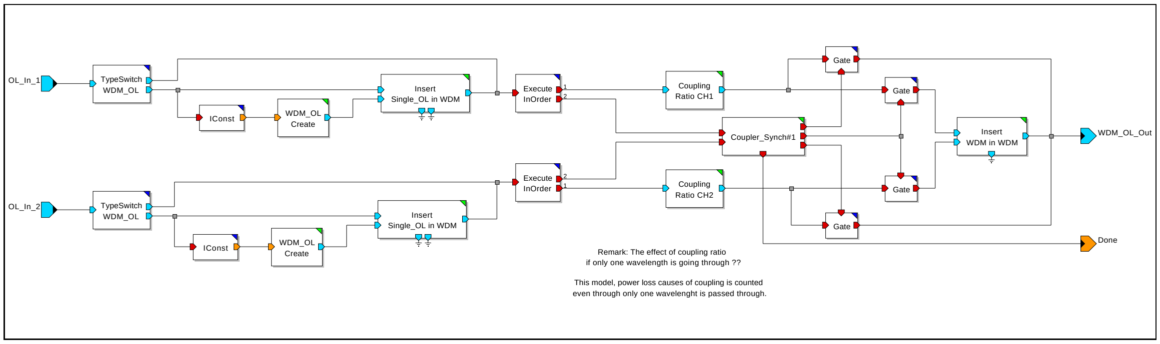

Figure 7: 2X1 Coupler

Inputs
OL_In_1
OL_In_2
Trigger

Outputs
WDM_OL_Out
Done

Parameters
Coupling_Ratio_CH1
Coupling_Ratio_CH2

4x1_Coupling_Module module in DE domain

This module couples four input optical signals to WDM output signal. Input signal can be either single or WDM OL but output signal is always WDM OL. If only one signal at input port is received at particular time, this signal can pass through output without coupling. Each channel has coupling ratio which can be defined individually.

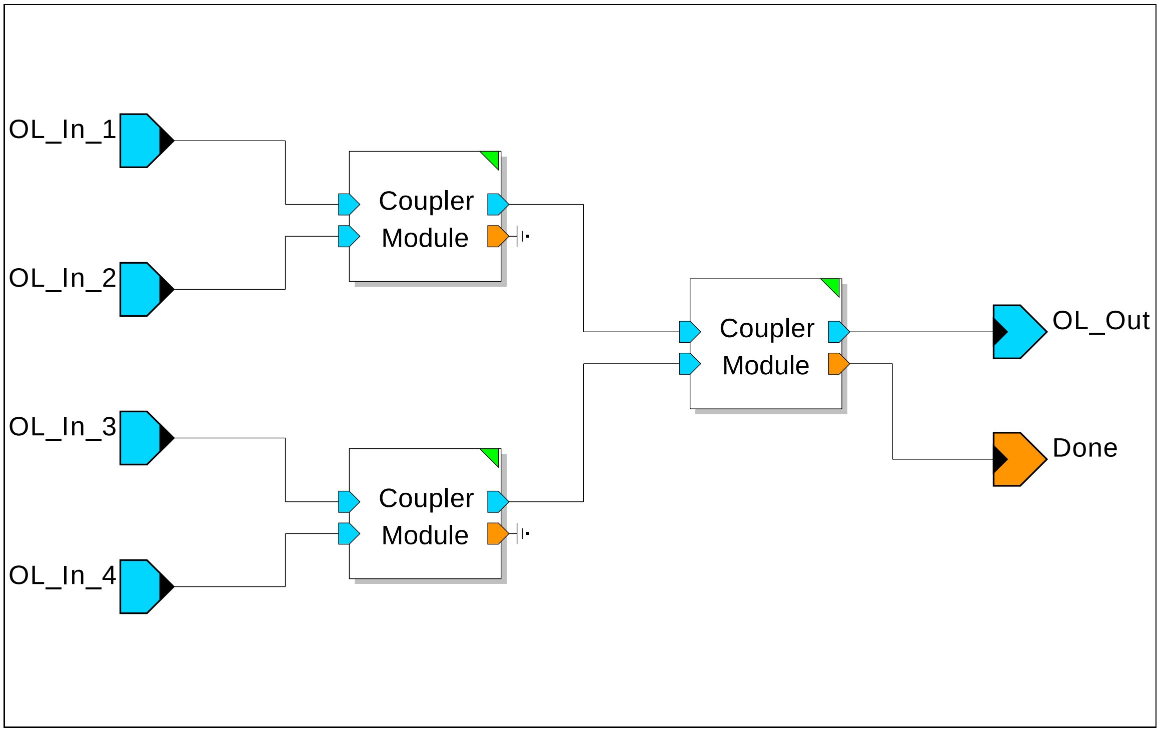

Figure 8: 4X1 Coupler

Inputs
OL_In_1
OL_In_2
OL_In_3
OL_In_4
Trigger

Outputs
OL_Out
Done

Parameters
Coupling_Ratio_CH1
Coupling_Ratio_CH2
Coupling_Ratio_CH3
Coupling_Ratio_CH4

8x1_Coupling_Module module in DE domain

This module couples eight input optical signals to WDM output signal. Input signal can be either single or WDM OL but output signal is always WDM OL. If only one signal at input port is received at particular time, this signal can pass through output without coupling. Each channel has coupling ratio which can be defined individually.

Inputs
OL_In_1
OL_In_2
OL_In_3
OL_In_4
OL_In_5
OL_In_6
OL_In_7
OL_In_8
Trigger

Outputs
OL_Out
Done

Parameters
Coupling_Ratio_CH1
Coupling_Ratio_CH2
Coupling_Ratio_CH3
Coupling_Ratio_CH4
Coupling_Ratio_CH5
Coupling_Ratio_CH6
Coupling_Ratio_CH7
Coupling_Ratio_CH8
Coupling_Ratio_CH4
Coupling_Ratio_CH5
Coupling_Ratio_CH6
Coupling_Ratio_CH7
Coupling_Ratio_CH8

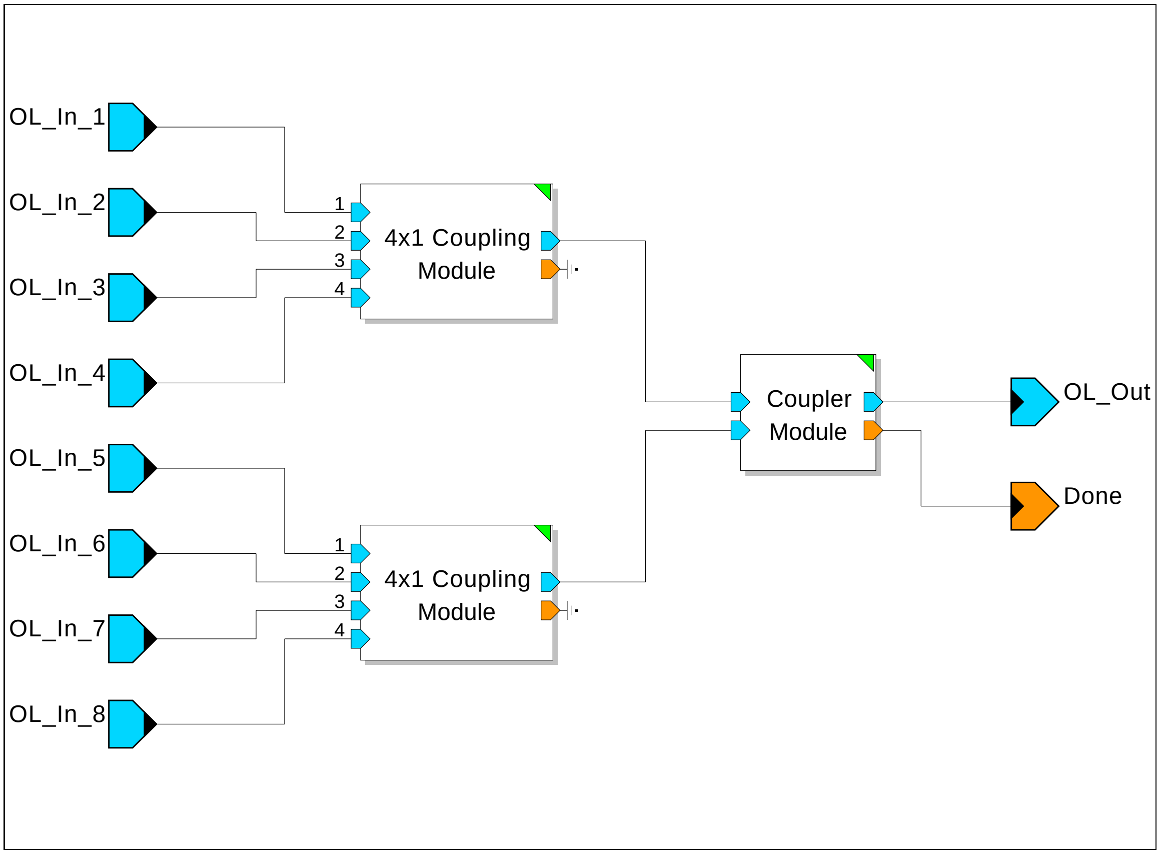

Figure 9: 1X8 Coupling Module

Add_Vector module in DE domain

This module adds any type of data structure at input data port (Data) into the existing vector (Vector_ In). The additional element will be added at the end of vector and the size of vector will increment by one. The vector with new element is given at the output port (Vector_ Out).

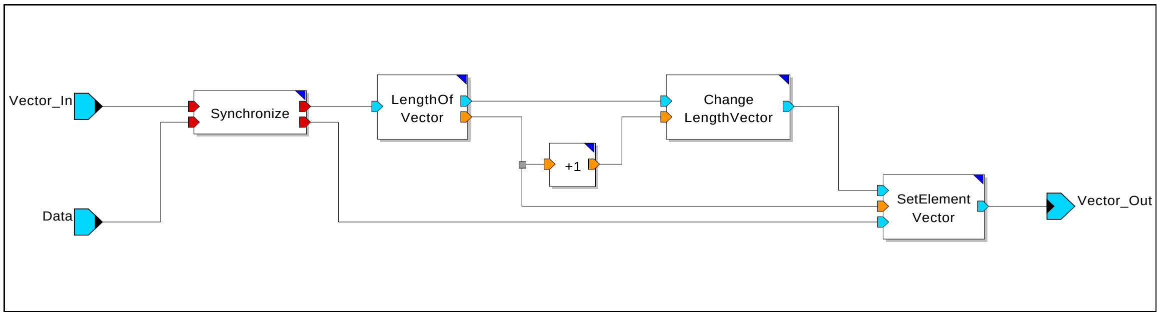

Figure 10: Add Vector

Inputs

Vector_In (datastruct:Root.Vector)
Data (datastruct:Root)

Outputs
Vector_Out (datastruct:Root.Vector)

Blank_Signal__ module in DE domain
This module checks the Single OL signal at input port (Single_ OL_ In) whether or not it contains any wavelength. By checking the wavelength value, if wavelength is less than or equal zero, TRUE (1) is given at output port (Blank). If not, FALSE (0) is given.

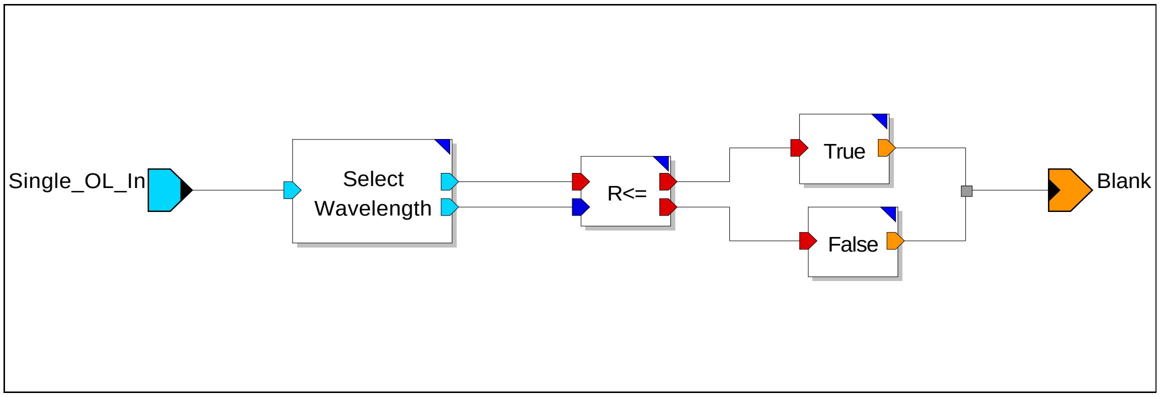

Figure 11: Blank Signal

Inputs
Single_OL_In
datastruct:OPN_Version_1_0:Root.OpticalLayer.Single_OL

Outputs
Blank

Coupling_Channel module in DE domain
This module uses to couple two signals (Input1, Input2) according to channel control parameter. If control data has bit matching with channel parameter, Input2 will be coupled with Input1. Otherwise, Input1 will pass through output port (Output). The expression used to determine channel matching is defined as following:
Coupling channel = 2 ([P] Channel – 1).AND. (Control data)

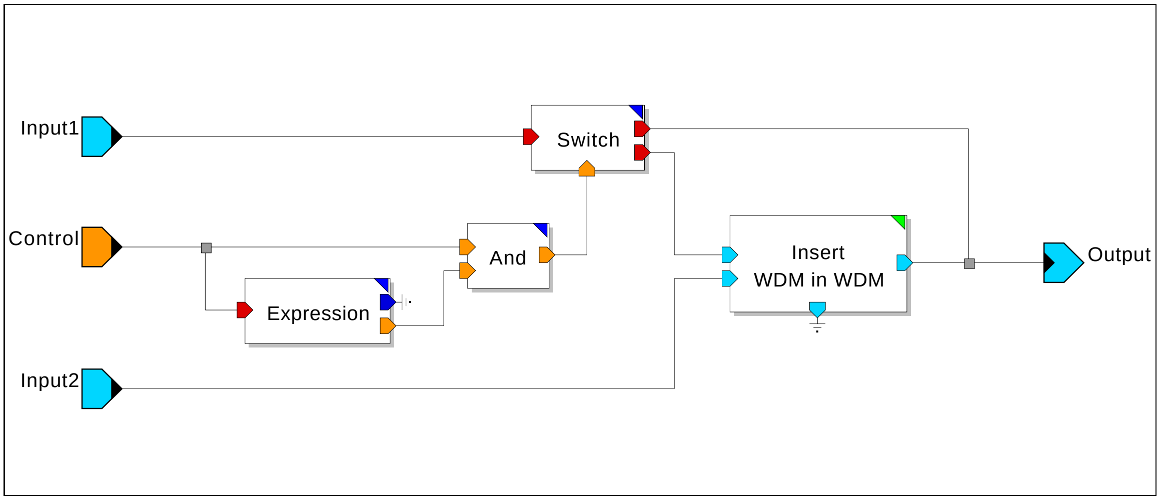

Figure 12: Coupling Channel

Inputs
Input1
Input2
Control

Outputs
Output

Parameters
Channel

Create_Switch_Control_DS_Primitive in DE domain

This internal creates a Switch Control Data Structure given the Switch Type and a Switch Control Vector. See documentation on the Switch Control DS for information on setting these inputs.

Inputs
Switch_Type
Ctrl_Vector

Outputs
Switch_Control_DS

CrossWave_Interference Primitive in DE domain

Injects crosswave interference into the data structure.

Inputs
OL_In

Outputs
OL_Out

DelElementVector primitive in DE domain

This primitive deletes the element in vector specified by the "VectorIn" input. The deleting element is specified by "Index" value. The index value must be less than or equal to maximum size of VectorIn. The vector size will be decrement by one after deleting and all following elements behind deleted element will be shifted up replacing the deleted element entry.

Inputs
VectorIn
datastruct
Index
int

Outputs
VectorOut
datastruct

Discrete_Tunable_Laser_Source module in DE domain

This module is used in the Discrete Tunable Optical Transmitter Module. Upon receiving a trigger this internal places values for the Opt_ Power and Wavelength on the output ports. The optical power is set by the parameter "Optical_ Power_ Level" and can be fluctuated by the parameter "Optical_ Power_ Fluctuation_ Vector". Refer to DE/ Sources/ WaveForm for more details on how to set this parameter. The output wavelength can be changed between a set of discrete values contained within the memory module "Discrete_ Wavelength_ Vector". The number and value of the wavelengths in the table can be altered by clicking on the memory module and changing the Value field {# of WL: WL 0 WL 1 …}. The initial output wavelength is the first value in the table. A new wavelength can be selected by placing its index (0, 1, 2 …) on the Change_ Transmit_ Wavelength port. Indices are stored in memory and remain until replaced.

Inputs
Trigger
Scalar anytype
Change_Transmit_Scalar int

Outputs
Opt_Power
Wavelength

Parameters
Switching_Time

Optical_Power_Level
The constant value.
Optical_Power_Fluctuation_Vector
One period of the output waveform.

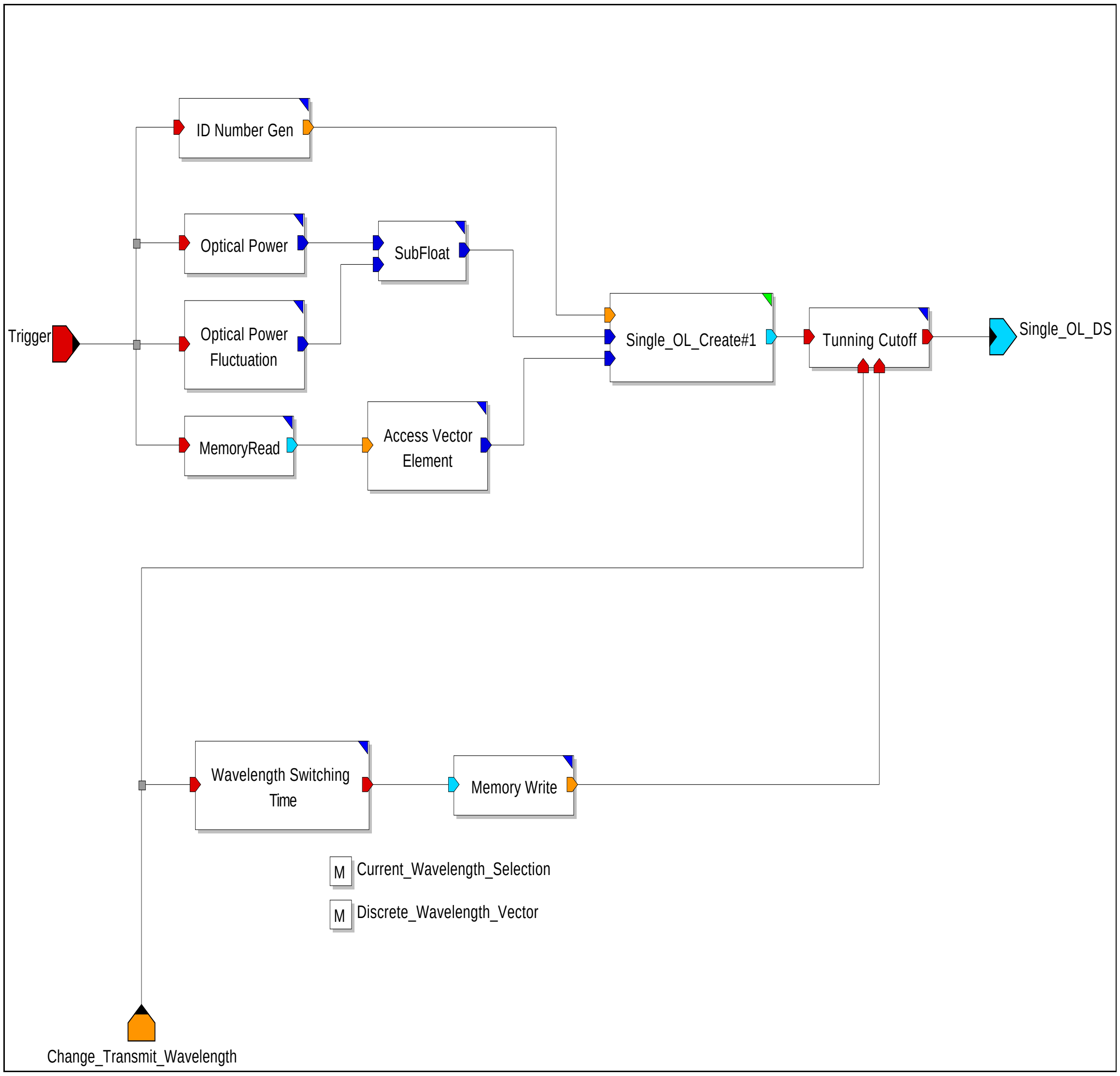

Figure 13: Tunable Laser Source Module

Memories
Discrete_Wavelength_Vector
Current_Wavelength_Selection

Divide_WDM_by_Wavelength module in DE domain

This module divides WDM OL signal (WDM_ OL_ In) into two WDM OL signals (WDM_ OL_ Upper, WDM_ OL_ Lower) by specified threshold wavelength. Any optical wavelengths which are higher than or equal to threshold wavelength are placed at

“WDM_ OL_ Upper” port as WDM signal. The other wavelengths which are less than or equal to threshold wavelength are placed at “WDM_ OL_ Lower” port as WDM signal. If no wavelength at any of output port, the empty WDM will be placed at that port. Also, if WDM input is empty wavelength, the empty WDM wavelength will be placed at both output ports.

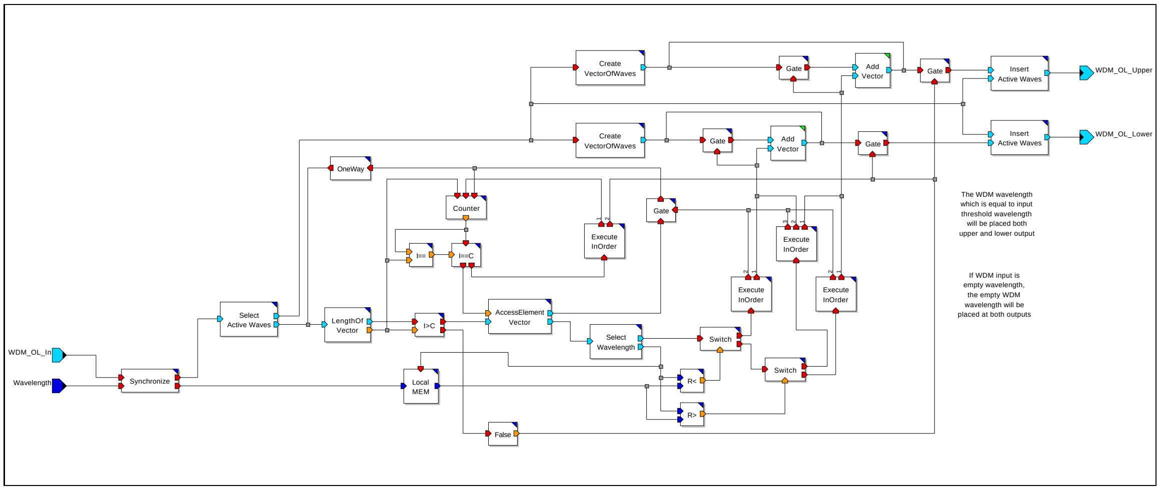

Figure 14: Divide WDM by Wavelength Module

Inputs

WDM_OL_In (datastruct:OPN_Version_2:Root.OpticalLayer.WDM_OL)
Wavelength

Outputs

WDM_OL_Upper (datastruct:OPN_Version_2:Root.OpticalLayer.WDM_OL)
WDM_OL_Lower (datastruct:OPN_Version_2:Root.OpticalLayer.WDM_OL)

<u>Extract_Fields_Single_OL_DS module in DE domain</u>

This internal places the value of each field of an input Single Optical Layer Data Structure on a corresponding output. The data structure is passed through unaltered.

Inputs

Single_OL_DS_In

Outputs

Single_OL_DS_Out
ID_Number
Creation_Time
Data
Number_of_Bits
Wavelength
Opt_Power_Level

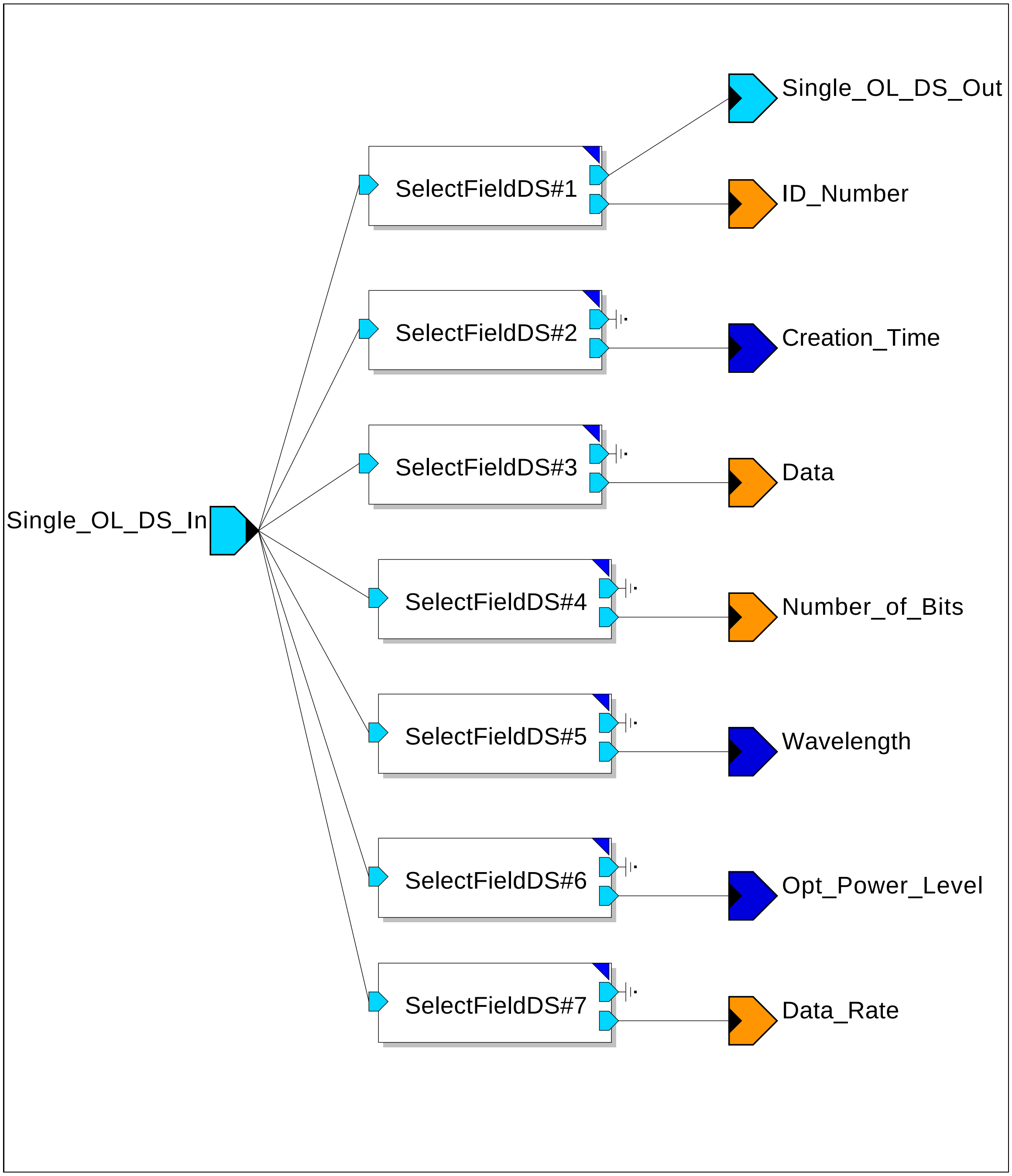

Figure 15: Extract Fields, Single OL DS Module

Extract_Fields_WDM_OL_DS module in DE domain

This internal places the value of each field of an input WDM Optical Layer Data Structure on a corresponding output. The data structure is passed through unaltered.

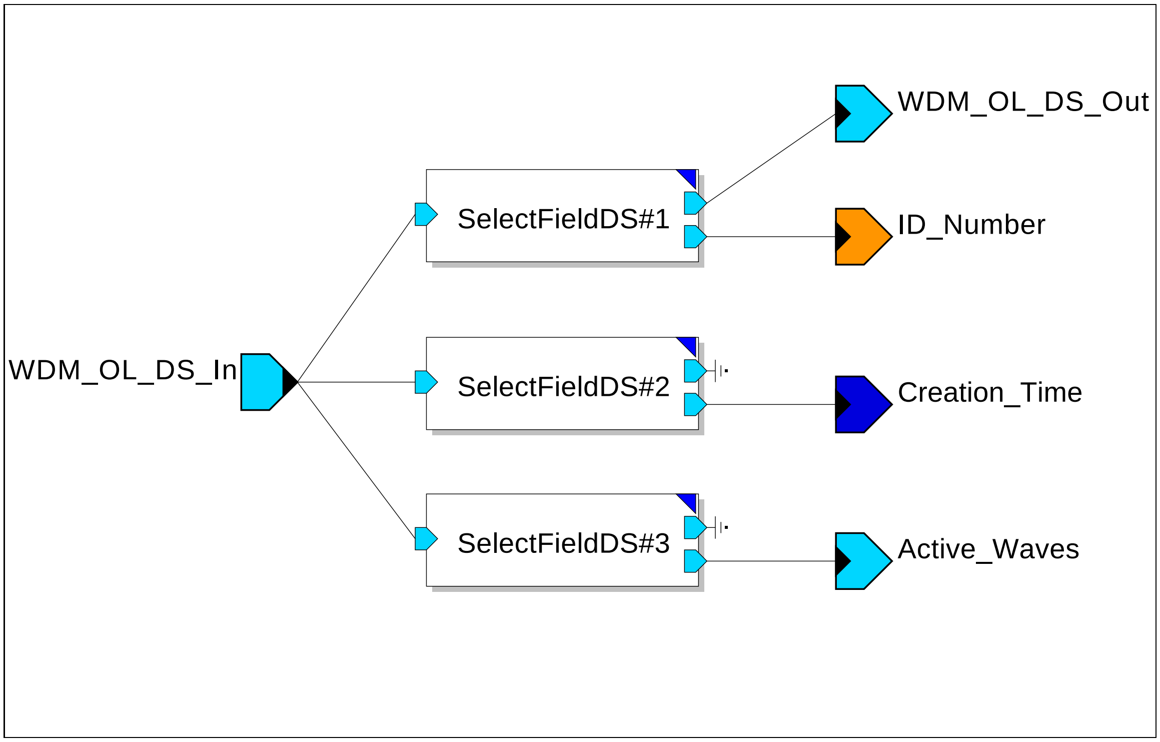

Figure 16: Extract Fields WDM OL DS Module

Inputs
WDM_OL_DS_In

Outputs
WDM_OL_DS_Out
ID_Number
Creation_Time
Active_Waves (datastruct:OPN_Version_1_0:Root.Vector.VectorOfWaves)

FIFOWithPeekReject module in DE domain

Module which implements a fixed length FIFO queue with a peek option.

Enabling the 'PeakTrigger' input causes a copy of the data structure at the head of the queue to be put on the output port 'PeekedOutput'. If nothing is in the queue when the PeekTrigger is enabled, the output 'NothingToSend' is enabled. Note that the peek option does not remove the data structure from the queue.

Data Structures enter the 'QueueInput' port and are queued in a First In First Out (FIFO) order. That is, the oldest data structures in this module leave first. When the 'ClearToSend' port is enabled the data structure at the head of the queue exits the 'QueueOutput' port. If the queue is empty when the 'ClearToSend' port is enabled the next arriving data structure will be allowed to flow through (no queueing delay). Any additional enables of the 'ClearToSend' port when the queue is empty are ignored.

If the parameter 'InitialQueueState' is set to 'InitialInputFlowsThrough' then the first data structure arriving to the queue flows through with no delay, else the first data structure waits for a 'ClearToSend' signal.

The first queue overflow displays a message and at the end of the simulation the total number of queue overflows for this queue will be displayed.

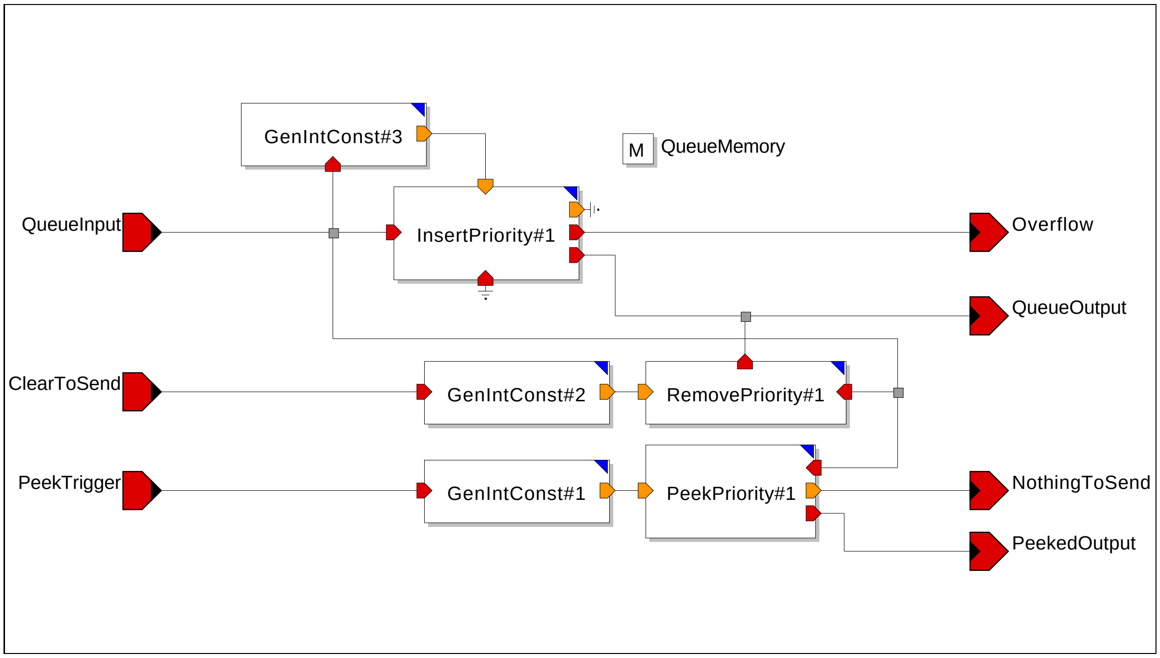

Figure 17: FIFO with Peek Reject Module

Inputs
ClearToSend
PeekTrigger
QueueInput

Outputs
NothingToSend
Overflow
PeekedOutput
QueueOutput

Parameters
InitialQueueState
MaximumQueueSize
QueueRejectMechanism

Memories
QueueMemory

Insert_Single_OL_in_WDM module in DE domain

This module provides function to add Single OL wavelength (Single_ OL_ In) into WDM OL (WDM_ OL_ In).If frequency of Single OL already exists in WDM OL, both input signals are dropped without insertion to “Single_ Interfere” and “WDM_ Interfere” ports. Otherwise, Single OL is inserted into WDM and new WDM is placed at output port (WDM_ OL_ Out).

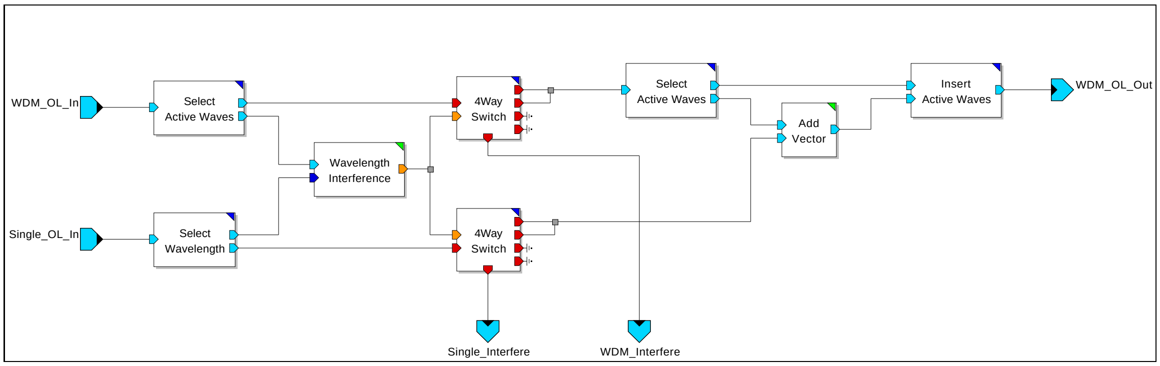

Figure 18: Insert Single OL in WDM Module

Inputs
WDM_OL_In
Single_OL_In

Outputs
WDM_OL_Out
WDM_Interfere
Single_Interfere

Laser_Source module in DE domain

This module creates fixed wavelength and power according to parameters setting. Wavelength can be specified by parameter “Laser_ Wavelength”. An associated power level can be set through parameter “Opt_ Power_ Level”. The power fluctuation is the ripple of the Opt_ Power_ Level. Fluctuation wave form is set through “Laser_ Power_ Fluctuation_ Vector”. Refer to /DE/ Sources/ WaveForm for more details how to set this parameter.

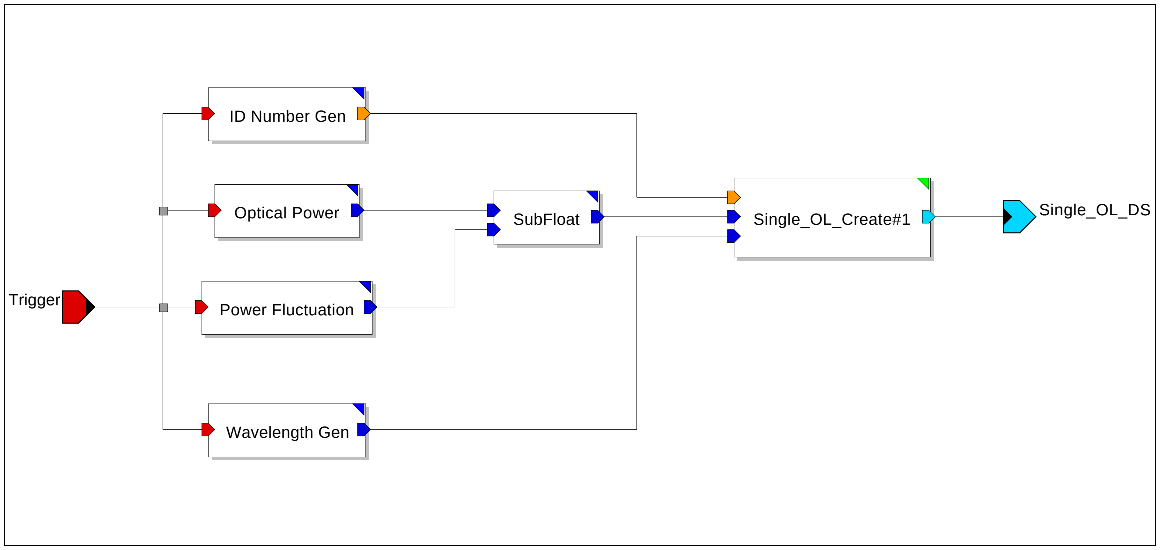

Figure 19: Laser Source Module

Inputs
Trigger
Scalar anytype

Outputs
Power
Wavelength

Parameters
Opt_Power_Level
Laser_Power_Fluctuation_Vector
Laser_Wavelength

Low_Power_Drop module in DE domain

This module provides capability to drop any wavelength which has power lower than power threshold value. Input signal at “OL_ In” port can be either Single OL or WDM OL. Any wavelength of input signal that has power less than “Power_ Threshold” value is placed at “Drop_ OL” port and others which are higher than or equal to power threshold are passed through “OL_ Out” port. If there is no wavelength less than power threshold value, “Drop_ OL” port is placed with empty WDM OL signal. In other hand, if no wavelength has power above or equal to threshold value, the empty WDM signal is placed at “OL_ Out” port.

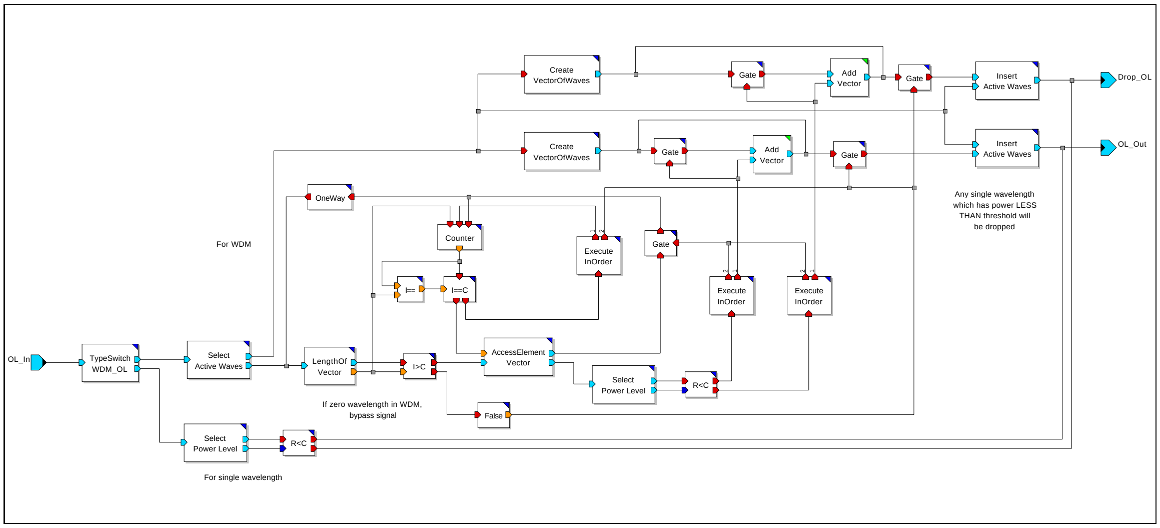

Figure 20: Low Power Drop Module

Inputs
OL_In

Outputs
OL_Out
Drop_OL

Parameters
Low_Power_Drop module in DE domain.

<u>Optical_Bit_Error_Injection module in DE domain</u>
Injects error into the DS.

Inputs
OL_In

Outputs
OL_Out

<u>Optical_Power_Gain module in DE domain</u>
This module provides functionality to increment power of each wavelength either Single OL or WDM OL. By using “Power Gain Expression” internal primitive, this module can increase power associated with wavelength individually. If empty WDM is received at input, the empty WDM output is placed at “OL_ Out” port. There are three parameters associated with this module, refer to “Power Gain Expression” for more details how to use these parameters.

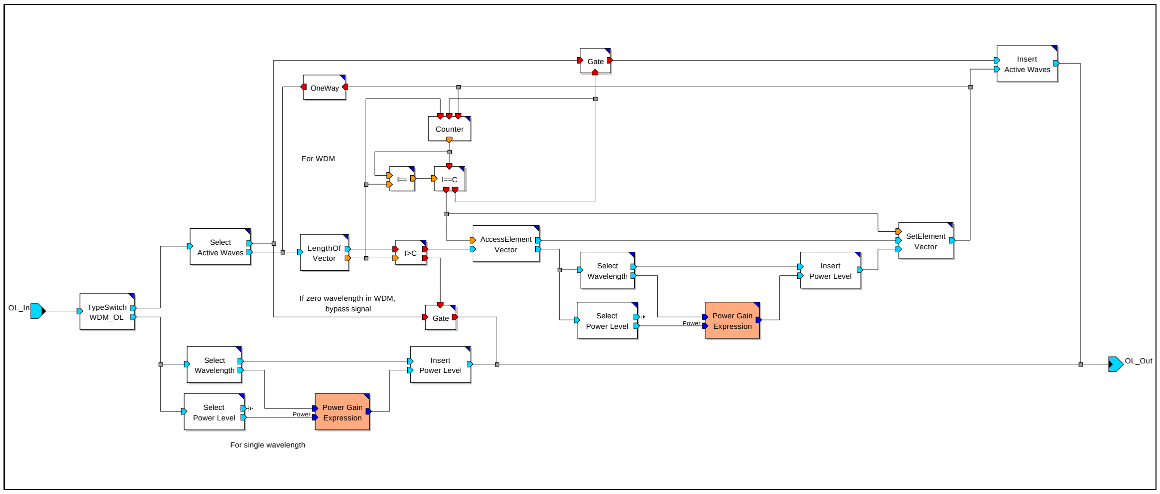

Figure 21: Optical Power Gain Module

Inputs
OL_In

Outputs
OL_Out

Parameters
Gain_dB
Gain_Factor
Gain_Selection (datastruct:OPN_Version_1_0:Root.ENUM.PowerSelection {dB})

Optical_Power_Loss module in DE domain

This module provides functionality to decrement power of each wavelength either Single OL or WDM OL. By using "Power Loss Expression" internal primitive, this module can decrease power associated with wavelength individually. If empty WDM is received at input, the empty WDM output is placed at "OL_ Out" port. There are three parameters associated with this module, refer to "Power Loss Expression" for more details how to use these parameters.

Inputs
OL_In

Outputs
OL_Out

Parameters
Loss_dB
Loss_Factor

Loss_Selection (datastruct:OPN_Version_1_0:Root.ENUM.PowerSelection {dB})

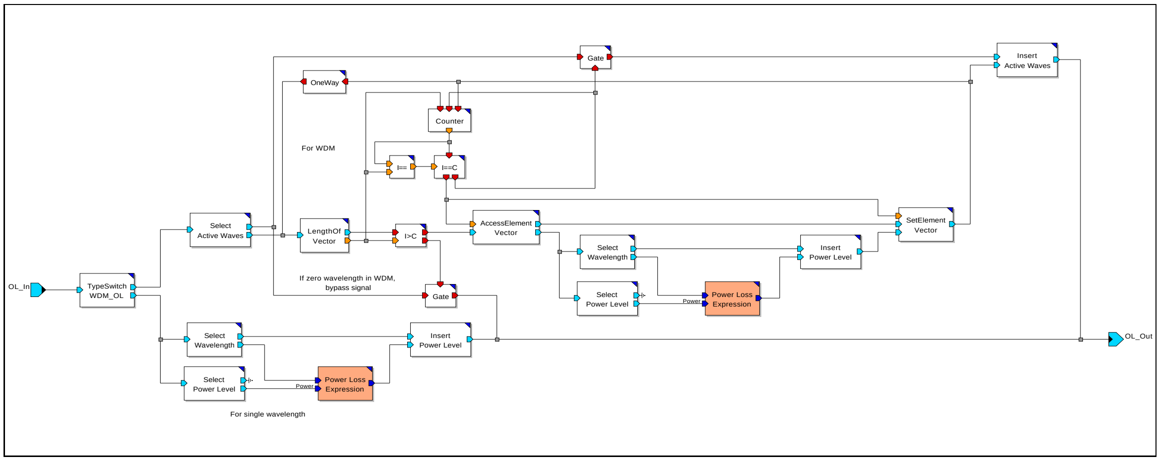

Figure 22: Optical Power Loss Module

Power_Gain_Expression __Single_ primitive in DE domain

This primitive provides capability to change power at “Power_ In” port with associated to “Freq” value. The equations given to “Power_ Out” are derived in primitive code. Refer to primitive code for equations. There are three parameters using in conjunction with primitive code. “Gain_ Selection” defined as enum type is type of calculation. If Gain_ Selection is “dB”, the power increment calculation is increased in amount of dB. If Gain_ Selection is “Factor”, the power increment calculation is increased in percentage of “Power_ In” value. “Gain_ dB” is amount of dB to be incremented. “Gain_ Factor” is percentage of “Power_ In” to be incremented. Note that “Gain_ Factor” is range between 0 and 1.

Inputs
Freq
Power_In

Outputs
Power_Out

Parameters
Gain_dB
Gain_Factor
Gain_Selection (datastruct OPN_Version_1_0:Root.ENUM.PowerSelection{dB})

Power_Loss_Expression __Single_ primitive in DE domain

This primitive provides capability to change power at “Power_ In” port with associated to “Freq” value. The equations given to “Power_ Out” are derived in primitive code. Refer to primitive code for equations. There are three parameters using in conjunction with primitive code. “Loss_ Selection” defined as enum type is type of calculation. If Loss_ Selection is “dB”, the power decrement calculation is decreased in amount of dB. If Loss_ Selection is “Factor”, the power decrement calculation is decreased in percentage of “Power_ In” value. “Loss_ dB” is amount of dB to be decremented. “Loss_ Factor” is percentage of “Power_ In” to be decremented. Note that “Loss_ Factor” is range between 0 and 1.

Inputs
Freq
Power_In

Outputs
Power_Out

Parameters
Loss_dB
Loss_Factor
Loss_Selection (datastruct OPN_Version_1_0:Root.ENUM.PowerSelection{dB})

<u>Remove_Single_OL_From_WDM module in DE domain</u>
This module provides function to remove any wavelength specified at “Wavelength” port from WDM OL (WDM_ OL_ In). If specified wavelength exists in WDM OL, its Single OL is removed from WDM OL and places at “Single_ OL_ Out” port. The remaining wavelengths are passed through “WDM_ OL_ Out” port. If specified wavelength does not exist in WDM OL, WDM OL input is passed through “WDM_ OL_ NotFound” port. If WDM OL is empty, input signal is passed through “WDM_ OL_ Empty” port.

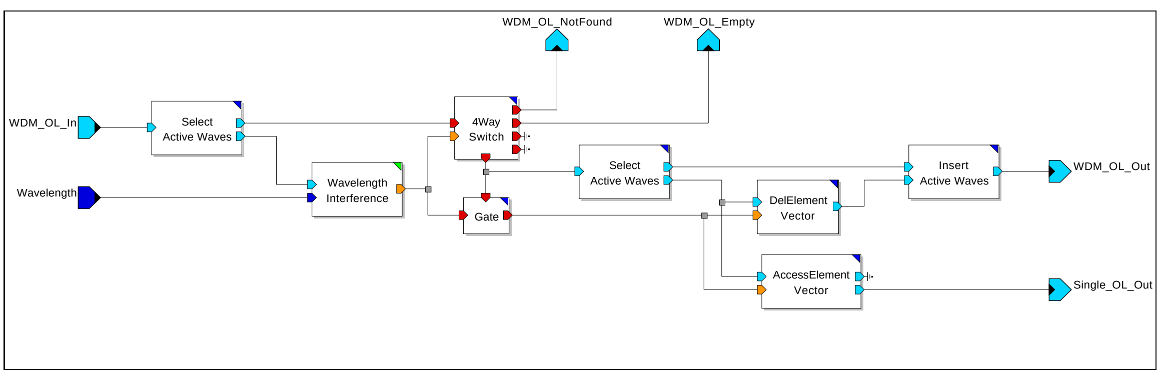

Figure 23: Remove Single OL from WDM Module

Inputs
WDM_OL_In
Wavelength

Outputs

WDM_OL_Out
WDM_OL_NotFound
WDM_OL_Empty
Single_OL_Out

Remove_Wavelength_from_Single_OL module in DE domain

This module removes Single OL wavelength by replacing all fields of Single OL structure with default values. These default values are considered as empty wavelength in Single OL.

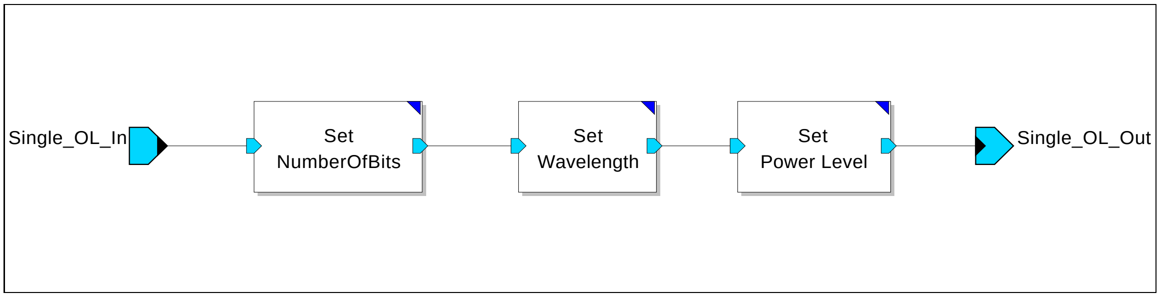

Figure 24: Remove Wavelength from Single OL Module

Inputs

Single_OL_In

Outputs

Single_OL_Out

Single_OL_Create module in DE domain

This module creates Single OL structure by giving ID, Wavelength and Power.

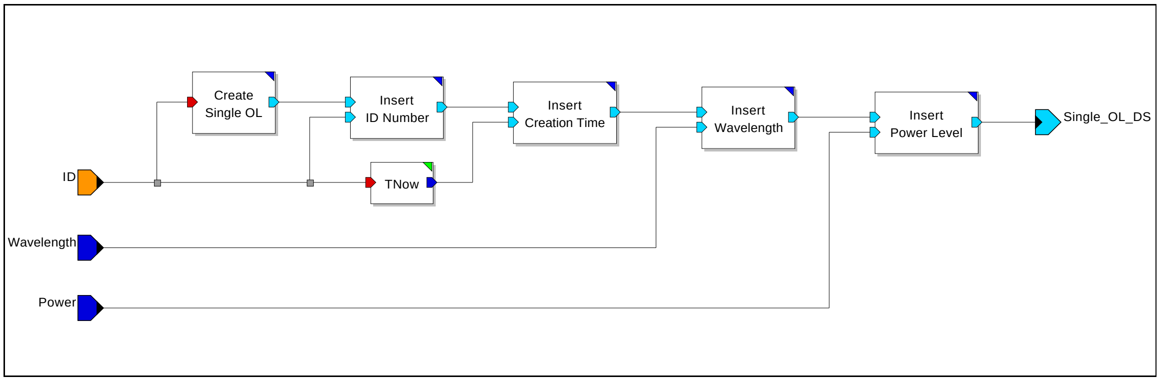

Figure 25: Single OL Create Module

Inputs

ID
Wavelength
Power

Outputs
Single_OL

Single_to_WDM_OL module in DE domain
This module converts Single OL structure to WDM OL structure. By inserting Single OL into new WDM OL, the output port (WDM_ OL) is placed WDM OL signal with one single wavelength corresponding to Single OL input.

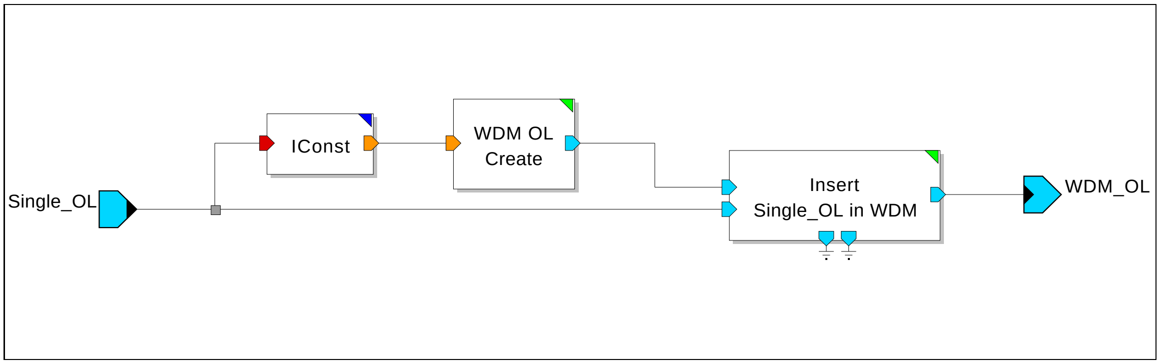

Figure 26: Single to WDM OL MOdule

Inputs
Single_OL

Outputs
WDM_OL

Splitting_Module module in DE domain
This module splits optical signal either single or WDM OL from one input port (OL_ In) to two output ports (OL_ Out_ 1, OL_ Out_ 2). The power level of each output channel is defined by individual splitting ratio (CH1_ Ratio, CH2_ Ratio).

Inputs
OL_In

Outputs
OL_Out_1
OL_Out_2

Parameters
CH1_Ratio (Splitting ratio is defined by loss factor of channel 1. Channel 2 is the remain ing of channel 1. The value of loss factor is between [0,1])
CH2_Ratio

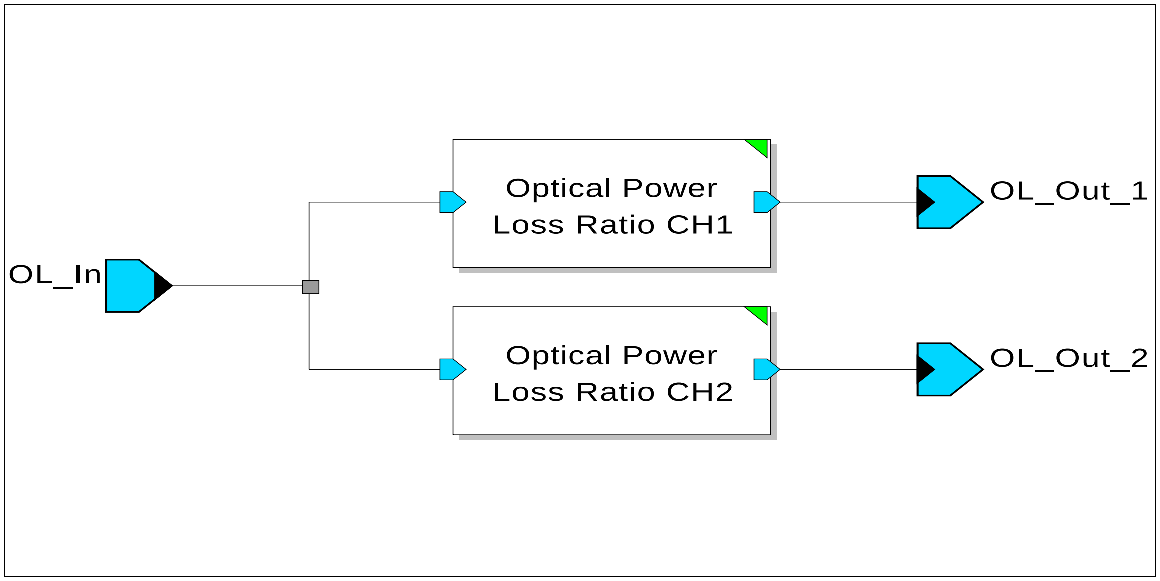

Figure 27: Splitting Module

String_to_DS primitive in DE domain

Converts the string representation of a data structure into the actual data structure.

Inputs

String

Type

Outputs

Data_Structure

Tunable_Laser_Source module in DE domain

This internal is used in the Discrete Tunable Optical Transmitter Module. Upon receiving a trigger this internal places values for the Opt_ Power and Wavelength on the output ports. The optical power is set by the parameter "Optical_ Power_ Level" and can be fluctuated by the parameter "Optical_ Power_ Fluctuation_ Vector". Refer to DE/ Sources/ WaveForm for more details on how to set this parameter. The output wavelength can be changed between a set of discrete values contained within the memory module "Discrete_ Wavelength_ Vector". The number and value of the wavelengths in the table can be altered by clicking on the memory module and changing the Value field {# of WL: WL 0 WL 1 …}. The initial output wavelength is the first value in the table.A new wavelength can be selected by placing its index (0, 1, 2 …) on the Change_ Transmit_ Wavelength port. Indices are stored in memory and remain until replaced.

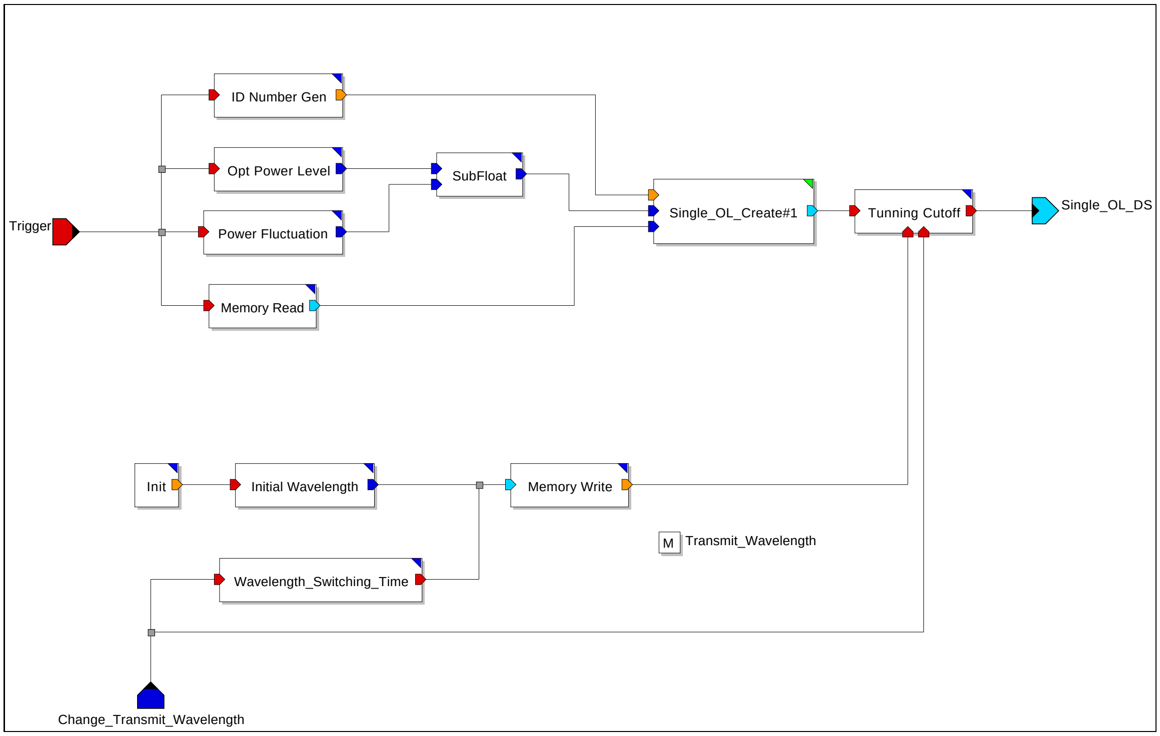

Figure 28: TUnable Laser Source Module

Inputs
Trigger
Change_Transmit_Wavelength

Outputs
Opt_Power
Wavelength

Parameters
Switching_Time
Initial_Wavelength
Laser_Power_Fluctuation_Vector
One period of the output waveform.
Opt_Power_Level

Memories
Transmit_Wavelength

<u>Wavelength_Interference module in DE domain</u>

This module determines the specified wavelength whether or not it already exists in WDM OL signal. By giving wavelength to test at "Wave_ Test" port, all wavelengths in WDM OL given at "Wave_ Vector" port are tested against Wave_ Test. The results of testing wavelength are given as following:

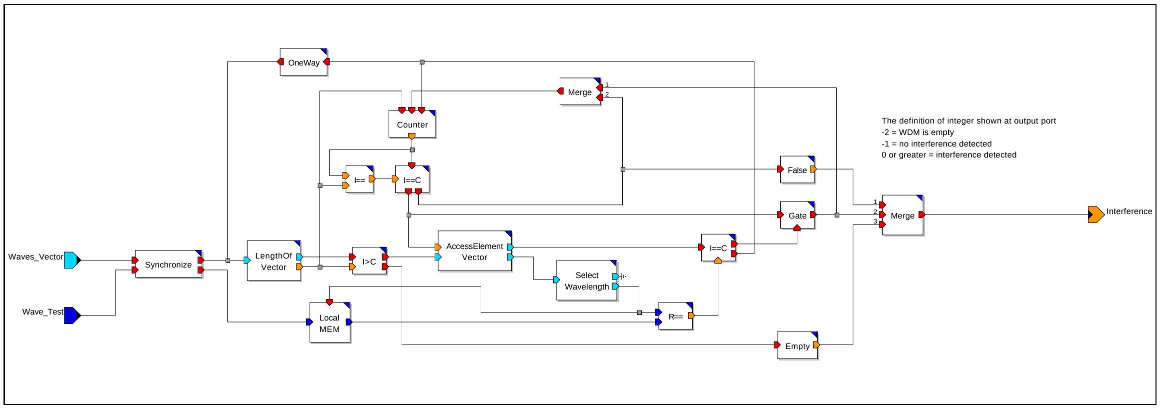

Figure 29: Wavelength Interference Module

Inputs
Waves_Vector
Wave_Test

Outputs
Interference

WDM_OL_Create module in DE domain

This module creates WDM OL structure with empty wavelength inside. The ID number of WDM is given at input port.

Inputs
ID

Outputs
WDM_OL

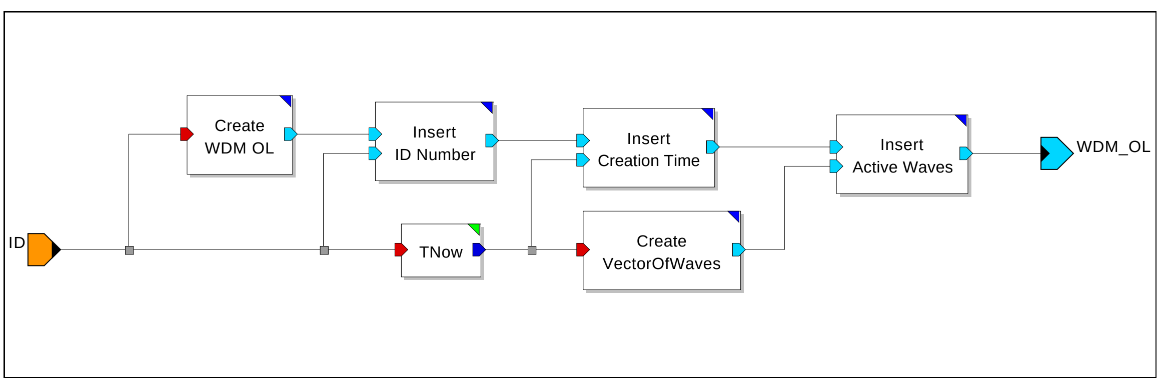

Figure 30: WDM OL Create Module

WDM_to_Single_OL module in DE domain

This module converts WDM OL signal to Single OL signal. If the given WDM OL at "WDM_ OL" port is empty, the input signal will be discarded and no event at "Single_

OL” output port. If the given WDM OL contains more than one wavelength, the input signal will be ignored and no event at output port 5 . Therefore, using this module must make sure that there is only one wavelength carrying in WDM OL. If only one wavelength in WDM OL input signal, the Single OL will be extracted from WDM OL and placed at “Single_ OL” output port.

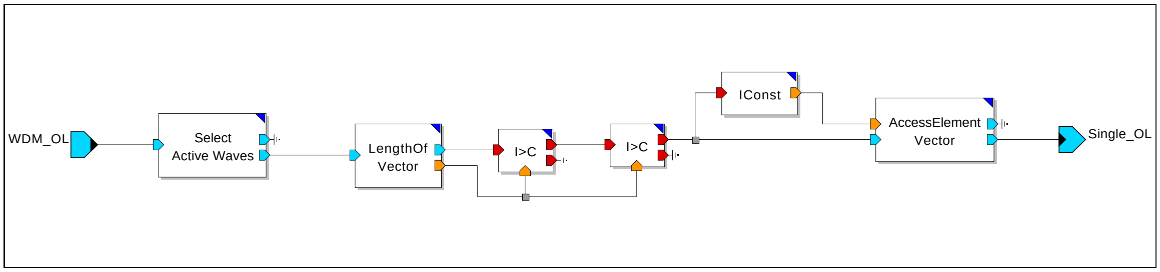

Figure 31: WDM to Single OL Module

Inputs
WDM_OL

Outputs
Single_OL

4.3 Components

4.3.1 OADM

1_Channel_OADM__Invert_Filter_ module in DE domain

This module performs optical add/ drop multiplexer (OADM) with adding and dropping one specified wavelength. This module exploits optical filters and invert filters to add and drop a specified wavelength. Adding and dropping frequency may or may not be the same frequency. If there are different, the adding wavelength is checked against input wavelengths whether it already exists. If adding wavelength already exists, the existing wavelength will be dropped by optical invert filter and the new wavelength signal will be replaced. In case of dropping wavelength, if the specified wavelength to be dropped does not exists in input signal, the empty WDM OL is created at "Drop_ OL". Refer to internal elements section for more details of each module.

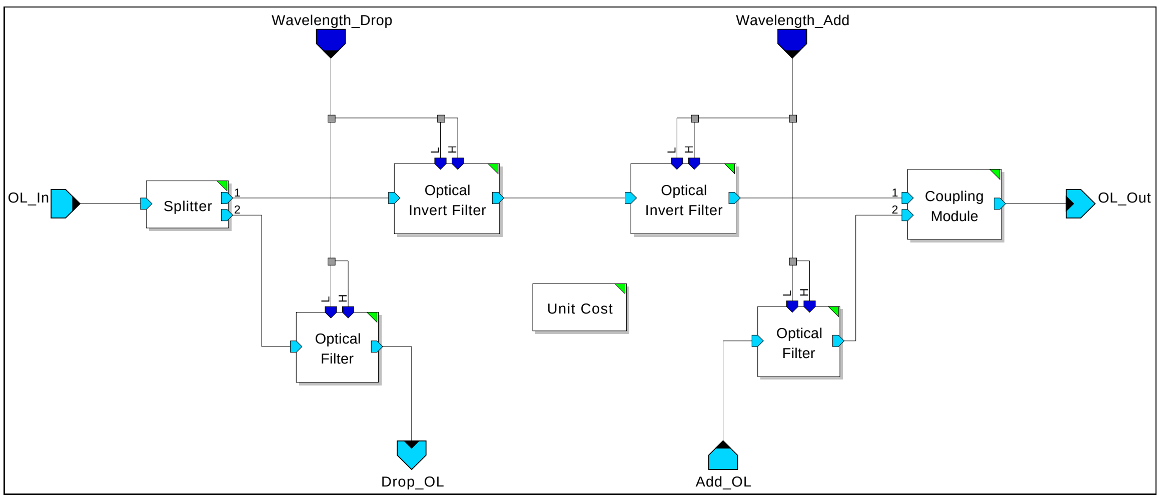

Figure 32: Channel OADM Invert Filter Module

Inputs

OL_In
Add_OL
Wavelength_Out
Wavelength_In

Outputs

Drop_OL
OL_Out

2_Channel_OADM__Invert_Filter_ module in DE domain

This module performs optical add/ drop multiplexer (OADM) with adding and dropping two specified wavelengths. This module exploits 1- Channel OADM module to add and drop specified wavelengths. Adding and dropping frequency may or may not be the same frequency. If there are different, the adding wavelength is checked against input wavelengths whether it already exists. If adding wavelength already exists, the existing wavelength will be dropped and the new wavelength signal will be replaced. In case of dropping wavelength, if the specified wavelength to be dropped does not exists in input. signal, the empty WDM OL is created at “Drop_ OL”. Refer to “1 Channel OADM” module for more details.

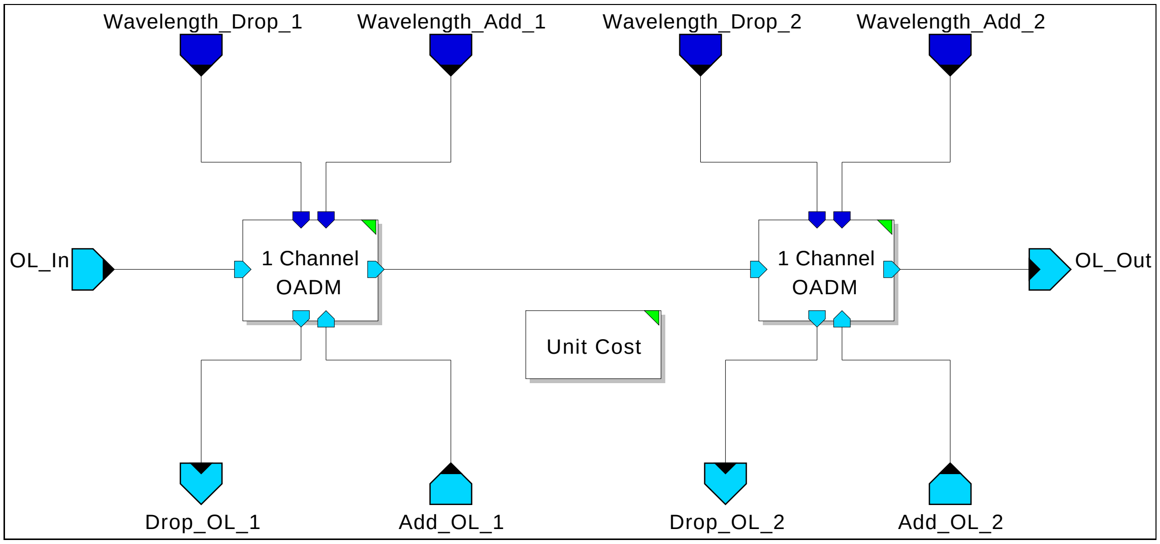

Figure 33: 2 Channel OADM Invert Filter Module

Inputs
OL_In
Add_OL_1
Wavelength_Out_1
Wavelength_In_1
Add_OL_2
Wavelength_Out_2
Wavelength_In_2

Outputs
Drop_OL_1
OL_Out
Drop_OL_2

4_Channel_OADM__Invert_Filter_ module in DE domain

This module performs optical add/ drop multiplexer (OADM) with adding and dropping two specified wavelengths. This module exploits 2 Channel OADM module to add and drop specified wavelengths. Adding and dropping frequency may or may not be the same frequency. If there are different, the adding wavelength is checked against input wavelengths whether it already exists. If adding wavelength already exists, the existing wavelength will be dropped and the new wavelength signal will be replaced. In case of dropping wavelength, if the specified wavelength to be dropped does not exists in input signal, the empty WDM OL is created at “Drop_ OL”. Refer to “2 Channel OADM” module for more details.

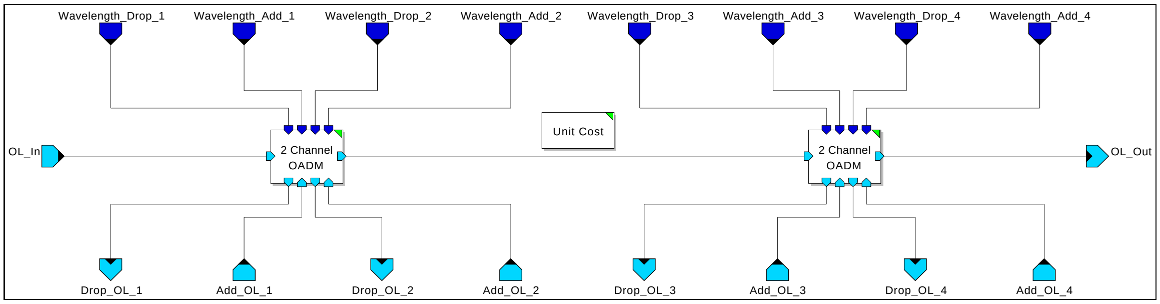

Figure 34: 4 Channel OADM Invert Filter Module

Inputs
Add_OL_1
Add_OL_2
Add_OL_3
Add_OL_4
OL_In
Wavelength_In_1
Wavelength_In_2
Wavelength_In_3
Wavelength_In_4
Wavelength_Out_1
Wavelength_Out_2
Wavelength_Out_3
Wavelength_Out_4

Outputs
Drop_OL_1
OL_Out
Drop_OL_2
Drop_OL_3
Drop_OL_4

4.3.2 Couplers

2x1_Coupler module in DE domain

This module couples two input optical signals to WDM output signal. Input signal can be either single or WDM OL but output signal is always WDM OL. If only one signal at input port is received at particular time, this signal can pass through output without coupling. Each channel has coupling ratio which can be defined individually. Each input port has input coupling loss which can be defined in dB as well as output coupling loss. The coupling module is exploited from internal library; refer to "2x1 Coupler Module" for more details.

Inputs
OL_In_1
OL_In_2

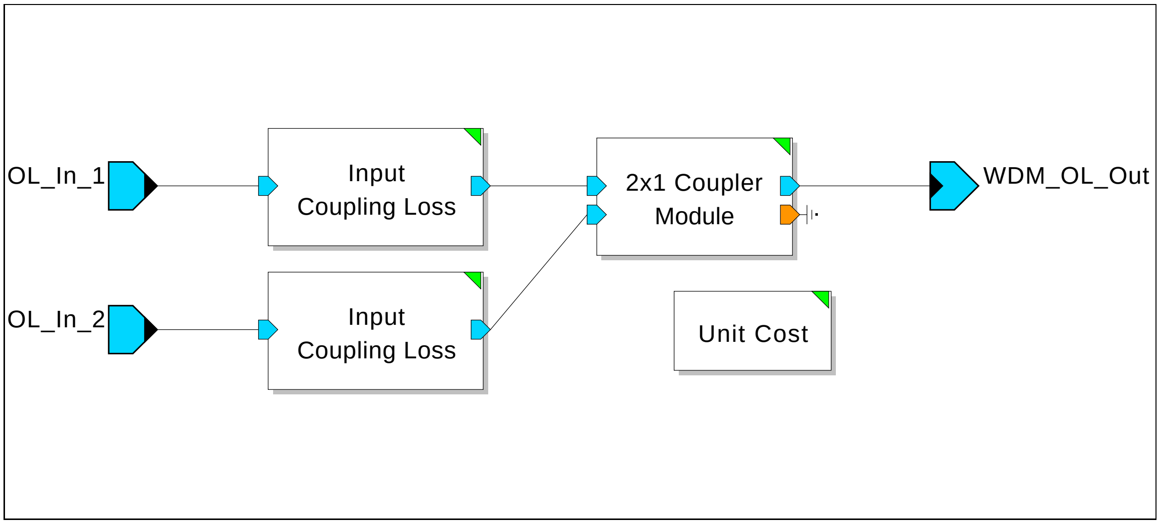

Figure 35: 2X1 Coupler Module

Outputs
WDM_OL_Out
Parameters:
Input_Coupling_Loss_CH1
Input_Coupling_Loss_CH2
Output_Coupling_Loss
Coupling_Ratio_CH1
Coupling_Ratio_CH2
InitTime
StopTime
InterPulseTime

2x4_Coupler module in DE domain

This module couples two input optical signals and splits coupled signal into four outputs corresponding to coupling and splitting ratio of each input port and output port. Input signal can be either single or WDM OL but output ports are always WDM OL. If only one signal at input port is received at particular time, this signal can pass through and split to outputs without coupling. Each channel has coupling ratio which can be defined individually. Each input port has input coupling loss which can be defined in dB as well as output coupling loss. The coupling module is exploited from internal library; refer to "2x1 Coupler Module" for more details. The splitting module is exploited from internal library; refer to "1x4 Splitting Module" for more details.

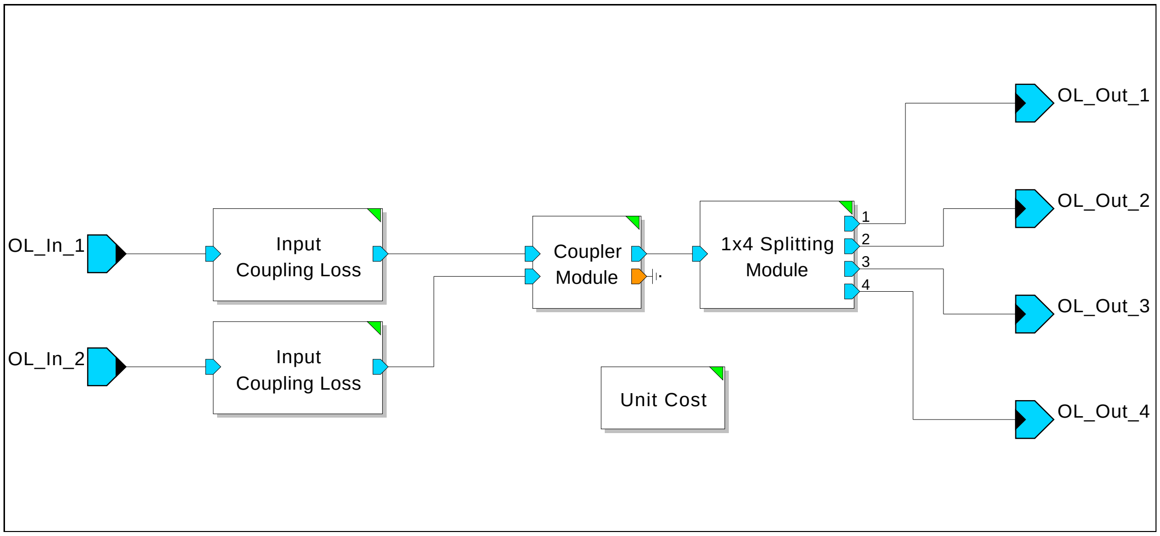

Figure 36: 2X4 Coupler Module

Inputs

OL_In_1
OL_In_2

Outputs

OL_Out_1
OL_Out_2
OL_Out_3
OL_Out_4

Parameters

Input_Coupling_Loss_CH1
Input_Coupling_Loss_CH2
Coupling_Ratio_CH1
Coupling_Ratio_CH2
Output_Coupling_Loss_CH1
Output_Coupling_Loss_CH2
Output_Coupling_Loss_CH3
Output_Coupling_Loss_CH4
Splitting_Ratio_CH2
Splitting_Ratio_CH3
Splitting_Ratio_CH4
InitTime
StopTime
InterPulseTime

2x8_Coupler module in DE domain

This module couples two input optical signals and splits coupled signal into eight outputs corresponding to coupling and splitting ratio of each input port and output port. Input signal can be either single or WDM OL but output ports are always WDM OL. If only one signal at input port is received at particular time, this signal can pass through and split to outputs without coupling. Each channel has coupling ratio which can be defined individually. Each input port has input coupling loss which can be defined in dB as well as output coupling loss. The coupling module is exploited from internal library; refer to “2x1 Coupler Module” for more details. The splitting module is exploited from internal library; refer to “1x8 Splitting Module” for more details.

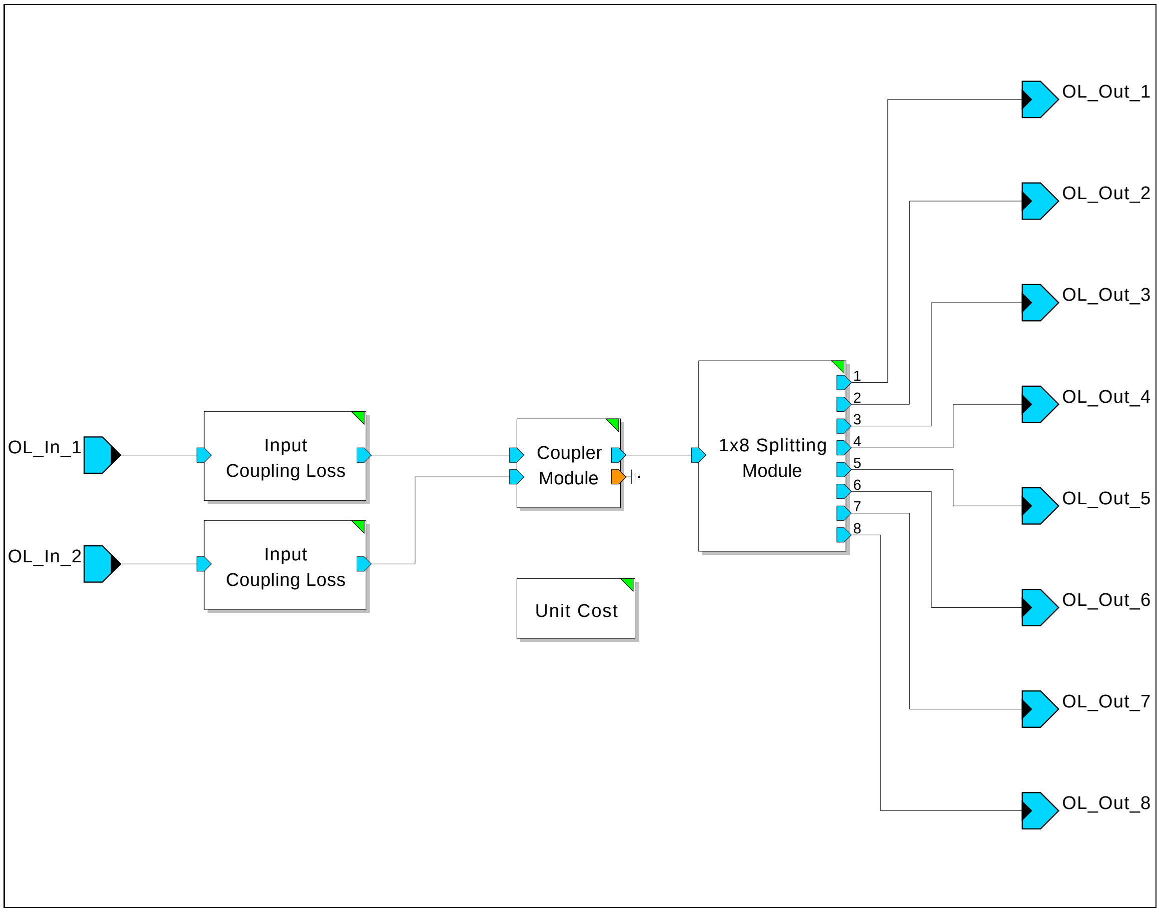

Figure 37: 2X8 Coupler Module

Inputs

OL_In_1
OL_In_2

Outputs

OL_Out_1
OL_Out_2

OL_Out_3
OL_Out_4
OL_Out_5
OL_Out_6
OL_Out_7
OL_Out_8

Parameters
Input_Coupling_Loss_CH1
Input_Coupling_Loss_CH2
Coupling_Ratio_CH1
Coupling_Ratio_CH2
Output_Coupling_Loss_CH1
Output_Coupling_Loss_CH2
Output_Coupling_Loss_CH3
Output_Coupling_Loss_CH4
Splitting_Ratio_CH2
Splitting_Ratio_CH3
Splitting_Ratio_CH4
Splitting_Ratio_CH5
Splitting_Ratio_CH6
Splitting_Ratio_CH7
Splitting_Ratio_CH8
Output_Coupling_Loss_CH5
Output_Coupling_Loss_CH6
Output_Coupling_Loss_CH7
Output_Coupling_Loss_CH8
InitTime
StopTime
InterPulseTime
float

4x1_Coupler module in DE domain

This module couples four input optical signals to WDM output signal. Input signal can be either single or WDM OL but output signal is always WDM OL. If only one signal at input port is received at particular time, this signal can pass through output without coupling. Each channel has coupling ratio which can be defined individually. Each input port has input coupling loss which can be defined in dB as well as output coupling loss. The coupling module is exploited from internal library; refer to “4x1 Coupler Module” for more details.

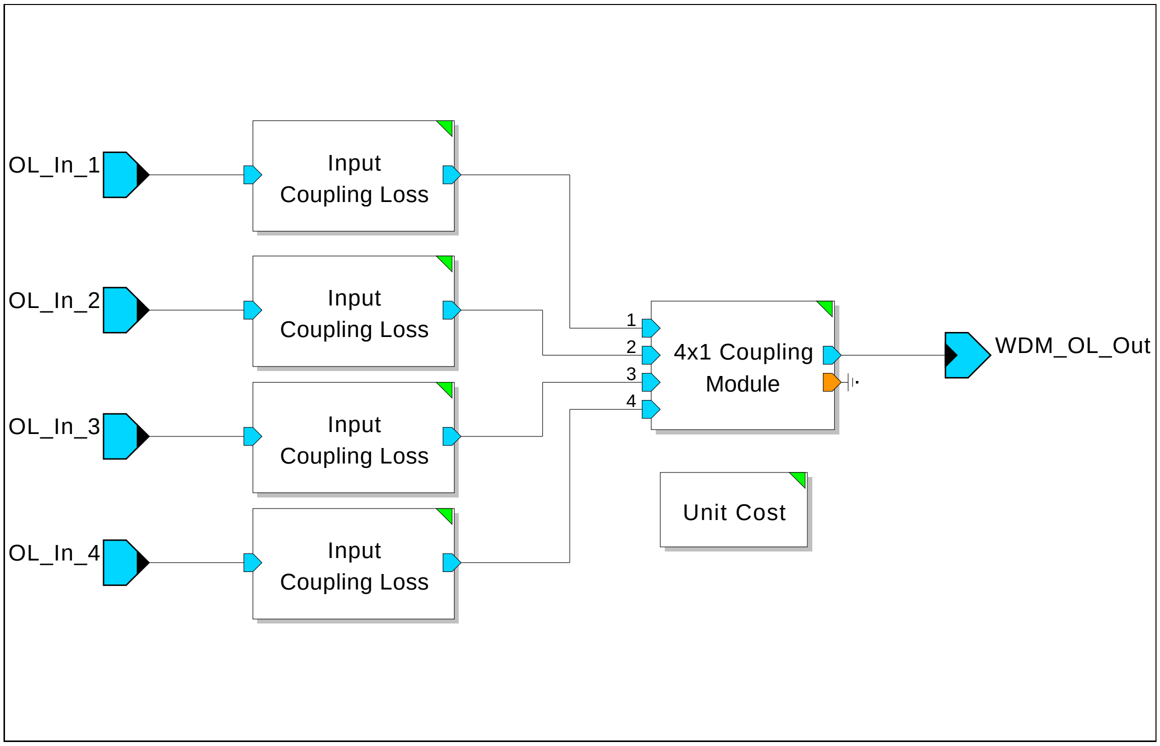

Figure 38: 1X4 Coupler Module

Inputs
OL_In_1
OL_In_2
OL_In_3
OL_In_4

Outputs
WDM_OL_Out

Parameters
Input_Coupling_Loss_CH1
Input_Coupling_Loss_CH2
Output_Coupling_Loss
Coupling_Ratio_CH1
Coupling_Ratio_CH2
Input_Coupling_Loss_CH3
Input_Coupling_Loss_CH4
Coupling_Ratio_CH3
Coupling_Ratio_CH4
InitTime
StopTime
InterPulseTime

4x4_Coupler module in DE domain

This module couples four input optical signals and splits coupled signal into four outputs corresponding to coupling and splitting ratio of each input port and output port. Input signal can be either single or WDM OL but output ports are always WDM OL. If only one signal at input port is received at particular time, this signal can pass through and split to outputs without coupling. Each channel has coupling ratio which can be defined individually. Each input port has input coupling loss which can be defined in dB as well as output coupling loss. The coupling module is exploited from internal library; refer to "4x1 Coupler Module" for more details. The splitting module is exploited from internal library; refer to "1x4 Splitting Module" for more details.

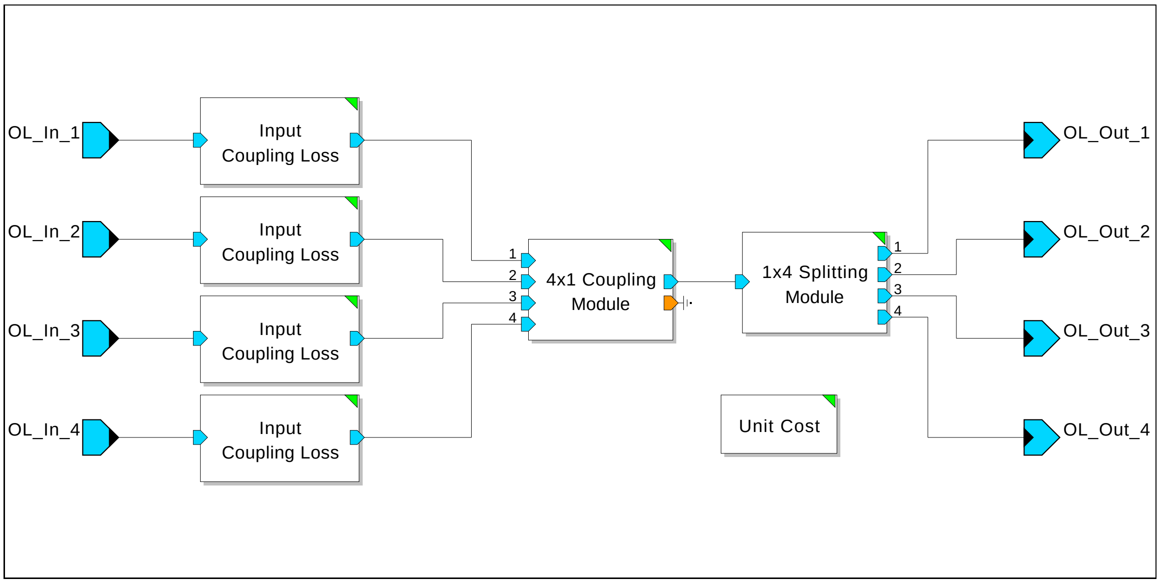

Figure 39: 4X4 Coupler Module

Inputs

OL_In_1
OL_In_2
OL_In_3
OL_In_4

Outputs

OL_Out_1
OL_Out_2
OL_Out_3
OL_Out_4

Parameters

Input_Coupling_Loss_CH1
Input_Coupling_Loss_CH2
Coupling_Ratio_CH1

Coupling_Ratio_CH2
Output_Coupling_Loss_CH1
Output_Coupling_Loss_CH2
Output_Coupling_Loss_CH3
Output_Coupling_Loss_CH4
Splitting_Ratio_CH2
Splitting_Ratio_CH3
Splitting_Ratio_CH4
Input_Coupling_Loss_CH3
Input_Coupling_Loss_CH4
Coupling_Ratio_CH3
Coupling_Ratio_CH4
InitTime
StopTime
InterPulseTime

8x1_Coupler module in DE domain

This module couples eight input optical signals to WDM output signal. Input signal can be either single or WDM OL but output signal is always WDM OL. If only one signal at input port is received at particular time, this signal can pass through output without coupling. Each channel has coupling ratio which can be defined individually. Each input port has input coupling loss which can be defined in dB as well as output coupling loss. The coupling module is exploited from internal library; refer to "8x1 Coupler Module" for more details.

Inputs

OL_In_1
OL_In_2
OL_In_3
OL_In_4
OL_In_5
OL_In_6
OL_In_7
OL_In_8

Outputs

OL_Out

Parameters

Input_Coupling_Loss_CH1
Input_Coupling_Loss_CH2
Output_Coupling_Loss
Coupling_Ratio_CH1
Coupling_Ratio_CH2

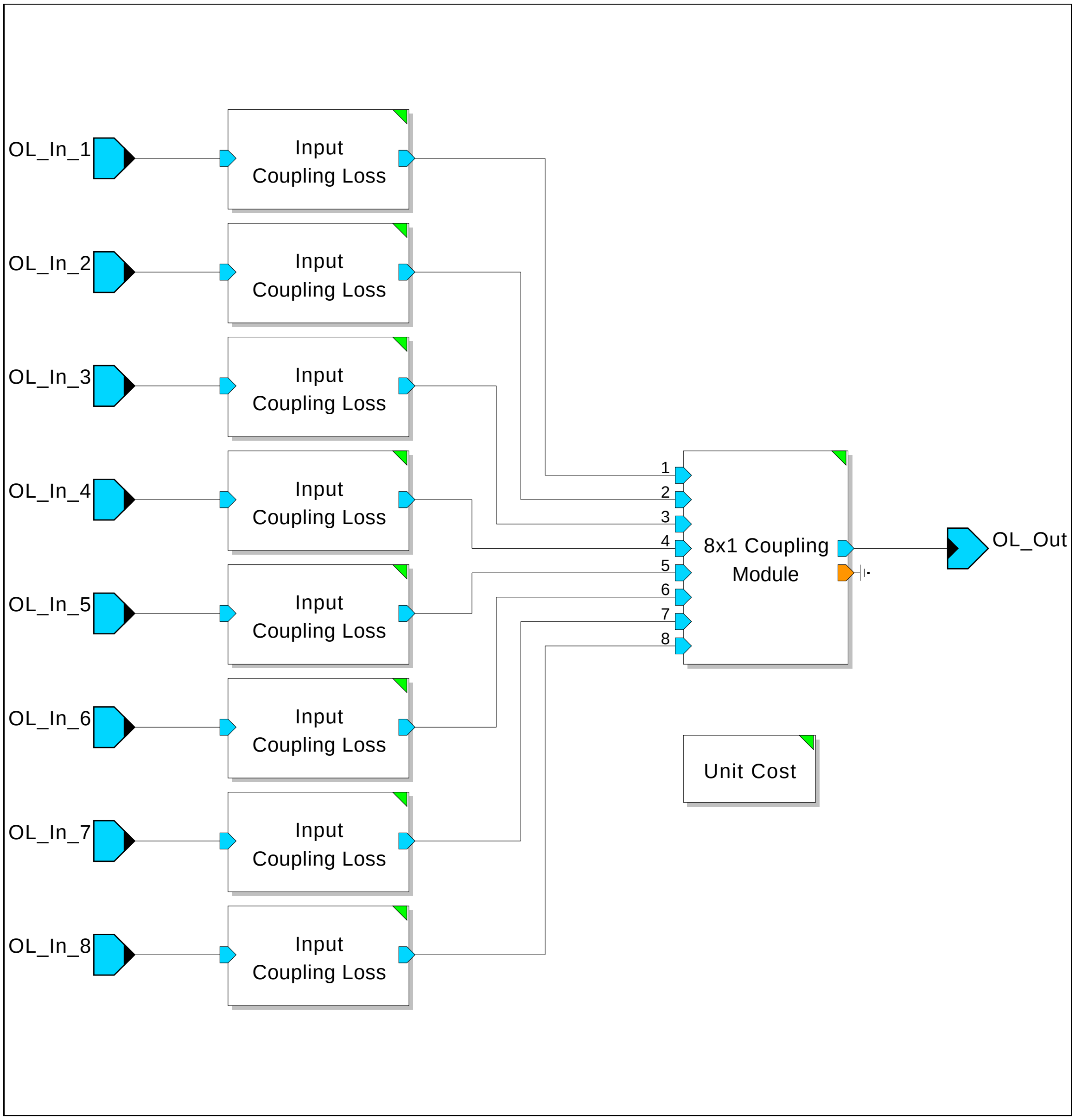

Figure 40: 1X8 Coupler Module

Input_Coupling_Loss_CH3
Input_Coupling_Loss_CH4
Coupling_Ratio_CH3
Coupling_Ratio_CH4
Input_Coupling_Loss_CH5
Input_Coupling_Loss_CH6
Input_Coupling_Loss_CH7
Input_Coupling_Loss_CH8
Coupling_Ratio_CH5
Coupling_Ratio_CH6
Coupling_Ratio_CH7

Coupling_Ratio_CH8
InitTime
StopTime
InterPulseTime

8x8 Coupler module in DE domain

This module couples eight input optical signals and splits coupled signal into four outputs corresponding to coupling and splitting ratio of each input port and output port. Input signal can be either single or WDM OL but output ports are always WDM OL. If only one signal at input port is received at particular time, this signal can pass through and split to outputs without coupling. Each channel has coupling ratio which can be defined individually. Each input port has input coupling loss which can be defined in dB as well as output coupling loss. The coupling module is exploited from internal library; refer to “8x1 Coupler Module” for more details. The splitting module is exploited from internal library; refer to “1x8 Splitting Module” for more details.

Inputs
OL_In_1
OL_In_2
OL_In_3
OL_In_4
OL_In_5
OL_In_6
OL_In_7
OL_In_8

Outputs
OL_Out_1
OL_Out_2
OL_Out_3
OL_Out_4
OL_Out_5
OL_Out_6
OL_Out_7
OL_Out_8

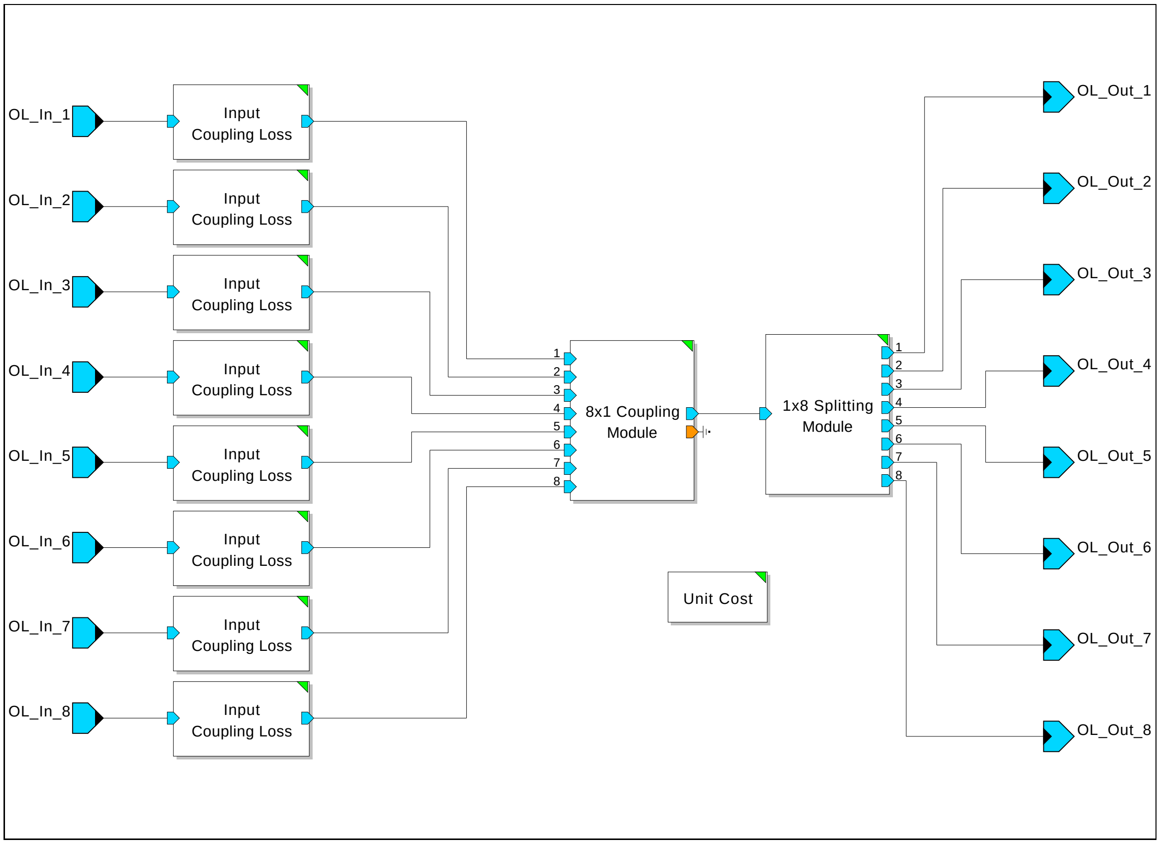

Figure 41: 8X8 Coupler Module

Parameters

Input_Coupling_Loss_CH1
Input_Coupling_Loss_CH2
Coupling_Ratio_CH1
Coupling_Ratio_CH2
Input_Coupling_Loss_CH3
Input_Coupling_Loss_CH4
Coupling_Ratio_CH3
Coupling_Ratio_CH4
Input_Coupling_Loss_CH5
Input_Coupling_Loss_CH6
Input_Coupling_Loss_CH7
Input_Coupling_Loss_CH8
Coupling_Ratio_CH5
Coupling_Ratio_CH6
Coupling_Ratio_CH7
Coupling_Ratio_CH8
Splitting_Ratio_CH1
Splitting_Ratio_CH2t
Splitting_Ratio_CH3

Splitting_Ratio_CH4
Splitting_Ratio_CH5
Splitting_Ratio_CH6
Splitting_Ratio_CH7
Splitting_Ratio_CH8
Output_Coupling_Loss_CH1
Output_Coupling_Loss_CH2
Output_Coupling_Loss_CH3
Output_Coupling_Loss_CH4
Output_Coupling_Loss_CH5
Output_Coupling_Loss_CH6
Output_Coupling_Loss_CH7
Output_Coupling_Loss_CH8
InitTime
StopTime
InterPulseTime

4.3.3 DEMUX

1x2_DEMUX module in DE domain

This module performs optical DEMUX by specified wavelengths to be demultiplexed. Optical input signal is split to number of output ports of DEMUX. Each split signal is passed through optical filter to select desired wavelength. By giving tuning wavelength, each output signal can be demultiplexed corresponding to those wavelengths. Refer to "Optical Filter" for more details how to use this module. Filtered signals from optical filter are WDM OL structure with only one wavelength. "WDM to Single_ OL" converts WDM OL to Single OL structure for outputs.

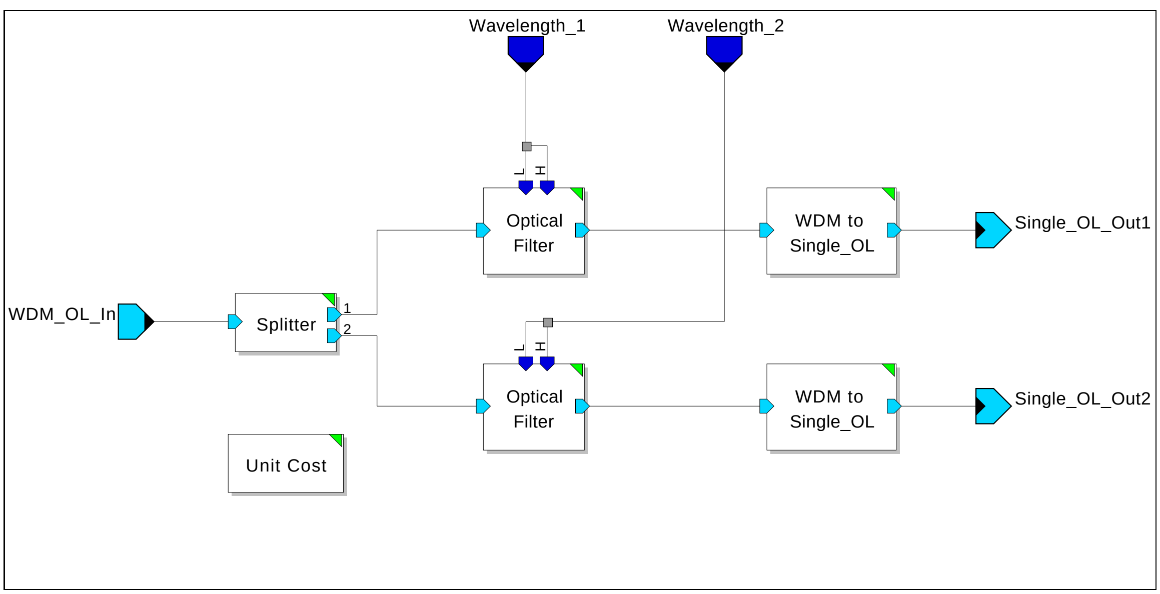

Figure 42: 1X2 DEMUX Module

Inputs
WDM_OL_In
Wavelength_1
Wavelength_2

Outputs
Single_OL_Out1
Single_OL_Out2

Parameters
Input_Coupling_Loss
Tune_Delay
Amount of time delay.
Output_Coupling_Loss_CH1
Output_Coupling_Loss_CH2

1x4 DEMUX module in DE domain

This module performs optical DEMUX by specified wavelengths to be demultiplexed. Optical input signal is split to number of output ports of DEMUX. Each split signal is passed through optical filter to select desired wavelength. By giving tuning wavelength, each output signal can be demultiplexed corresponding to those wavelengths. Refer to "Optical Filter" for more details how to use this module. Filtered signals from optical filter are WDM OL structure with only one wavelength. "WDM to Single_ OL" converts WDM OL to Single OL structure for outputs.

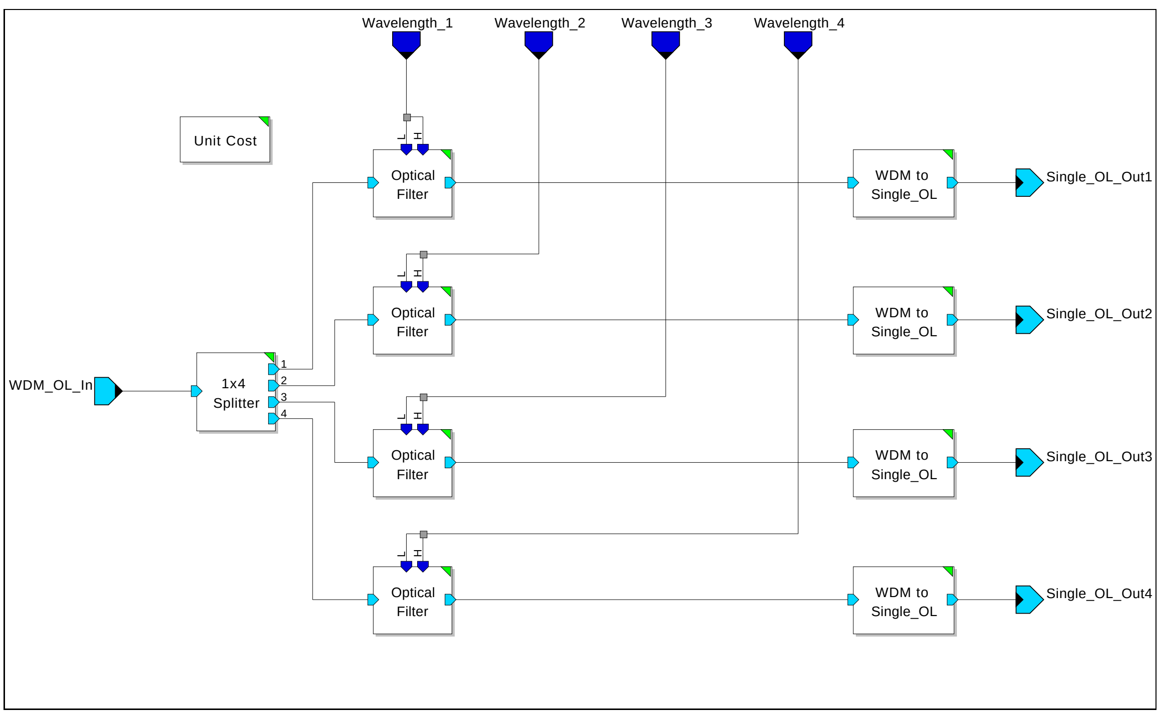

Figure 43: 1X4 DEMUX Module

Inputs
WDM_OL_In
Wavelength_1
Wavelength_2
Wavelength_3
Wavelength_4

Outputs
Single_OL_Out1
Single_OL_Out2
Single_OL_Out3
Single_OL_Out4

Parameters
Input_Coupling_Loss
Tune_Delay

1x8_DEMUX module in DE domain

This module performs optical DEMUX by specified wavelengths to be demultiplexed. Optical input signal is split to number of output ports of DEMUX. Each split signal is passed through optical filter to select desired wavelength. By giving tuning wavelength, each output signal can be demultiplexed corresponding to those wavelengths. Refer to "Optical Filter" for more details how to use this module. Filtered signals from optical filter are WDM OL structure with only one wavelength. "WDM to Single_ OL" converts WDM OL to Single OL structure for outputs.

Inputs
WDM_OL_In
Wavelength_1
Wavelength_2
Wavelength_3
Wavelength_4
Wavelength_5
Wavelength_6
Wavelength_7
Wavelength_8

Outputs
Single_OL_Out1
Single_OL_Out2
Single_OL_Out3
Single_OL_Out4
Single_OL_Out5
Single_OL_Out6

Single_OL_Out7
Single_OL_Out8

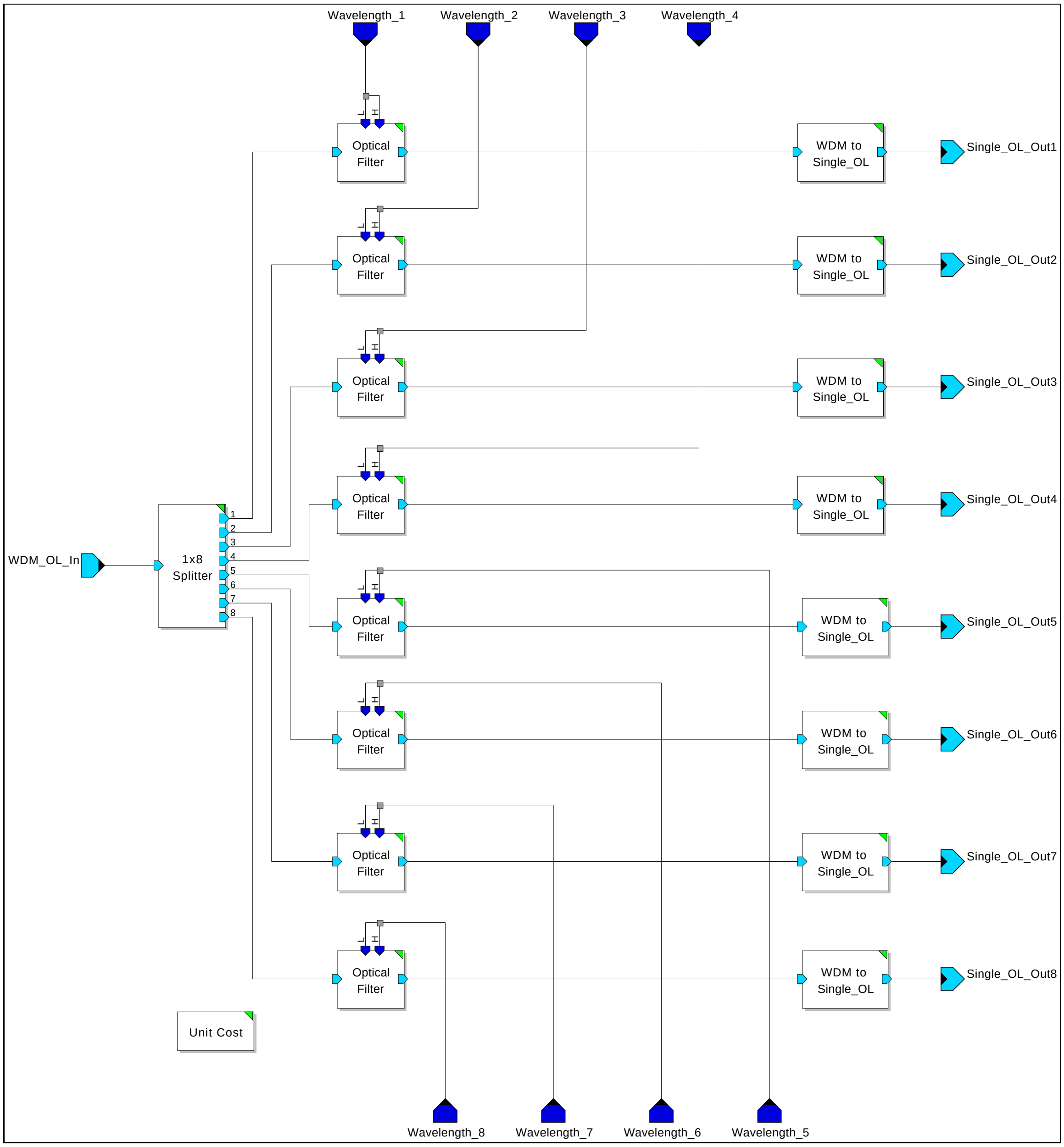

Figure 441X8 DEMUX Module

Parameters

Input_Coupling_Loss
Tune_Delay
Output_Coupling_Loss_CH1
Output_Coupling_Loss_CH2
Output_Coupling_Loss_CH3
Output_Coupling_Loss_CH4

Output_Coupling_Loss_CH5
Output_Coupling_Loss_CH6
Output_Coupling_Loss_CH7
Output_Coupling_Loss_CH8

4.3.4 MUX

2x1_MUX module in DE domain

This module performs optical MUX by specified wavelengths to be multiplexed. Each optical signal at input port (OL_ In1: OL_ In2) passes through optical filter to select wavelength to be multiplexed. Refer to “Optical Filter” for more details how to use this module. The filtered signals are multiplexed by using “2x1 Coupler” and place coupled signal at output port (WDM_ OL_ Out).

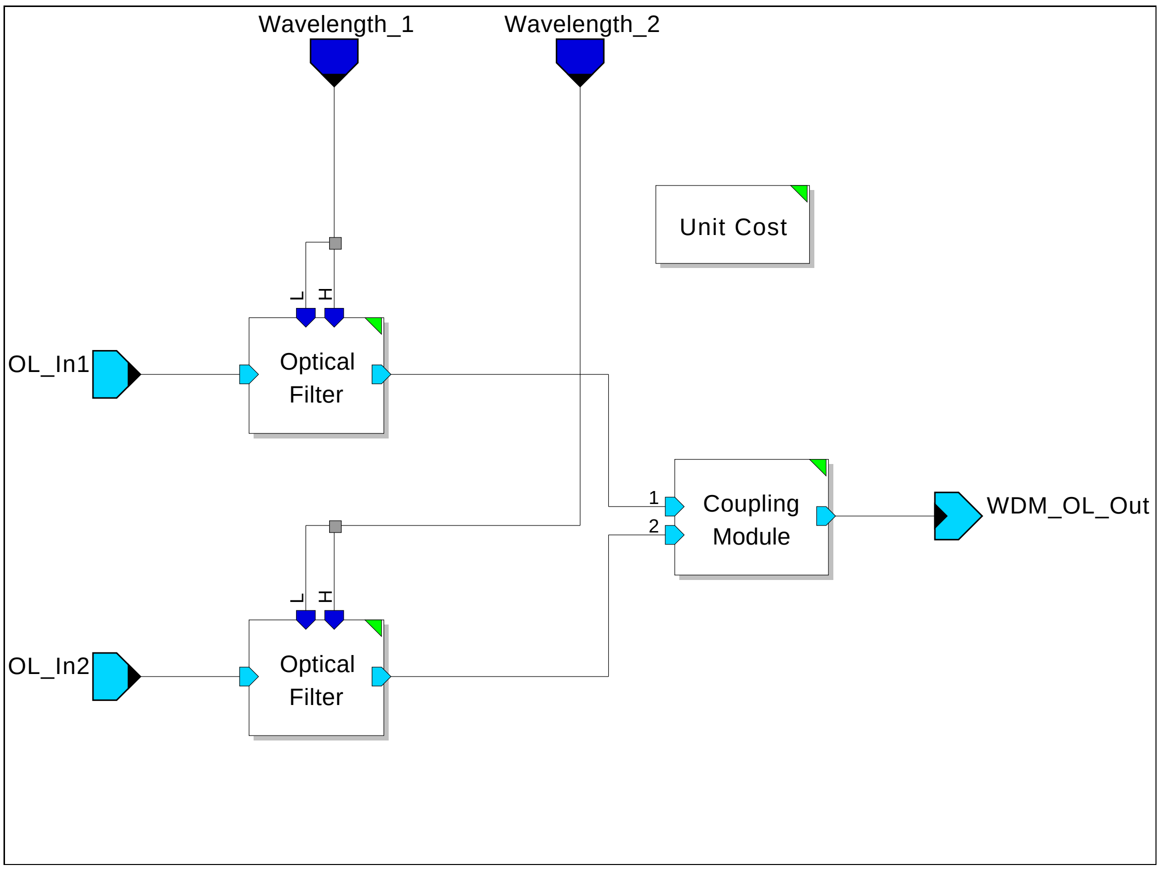

Figure 45: 2X1 MUX Module

Inputs

OL_In1
OL_In2
Wavelength_1
Wavelength_2

Outputs
WDM_OL_Out (datastruct:SystemDS:Root.Packet)

Parameters
Input_Coupling_Loss_CH2
Input_Coupling_Loss_CH1
Output_Coupling_Loss
Tune_Delay
Amount of time delay.

4x1_MUX module in DE domain

This module performs optical MUX by specified wavelengths to be multiplexed. Each optical signal at input port (OL_ In1: OL_ In4) passes through optical filter to select wavelength to be multiplexed. Refer to "Optical Filter" for more details how to use this module. The filtered signals are multiplexed by using "4x1 Coupler" and place coupled signal at output port (WDM_ OL_ Out).

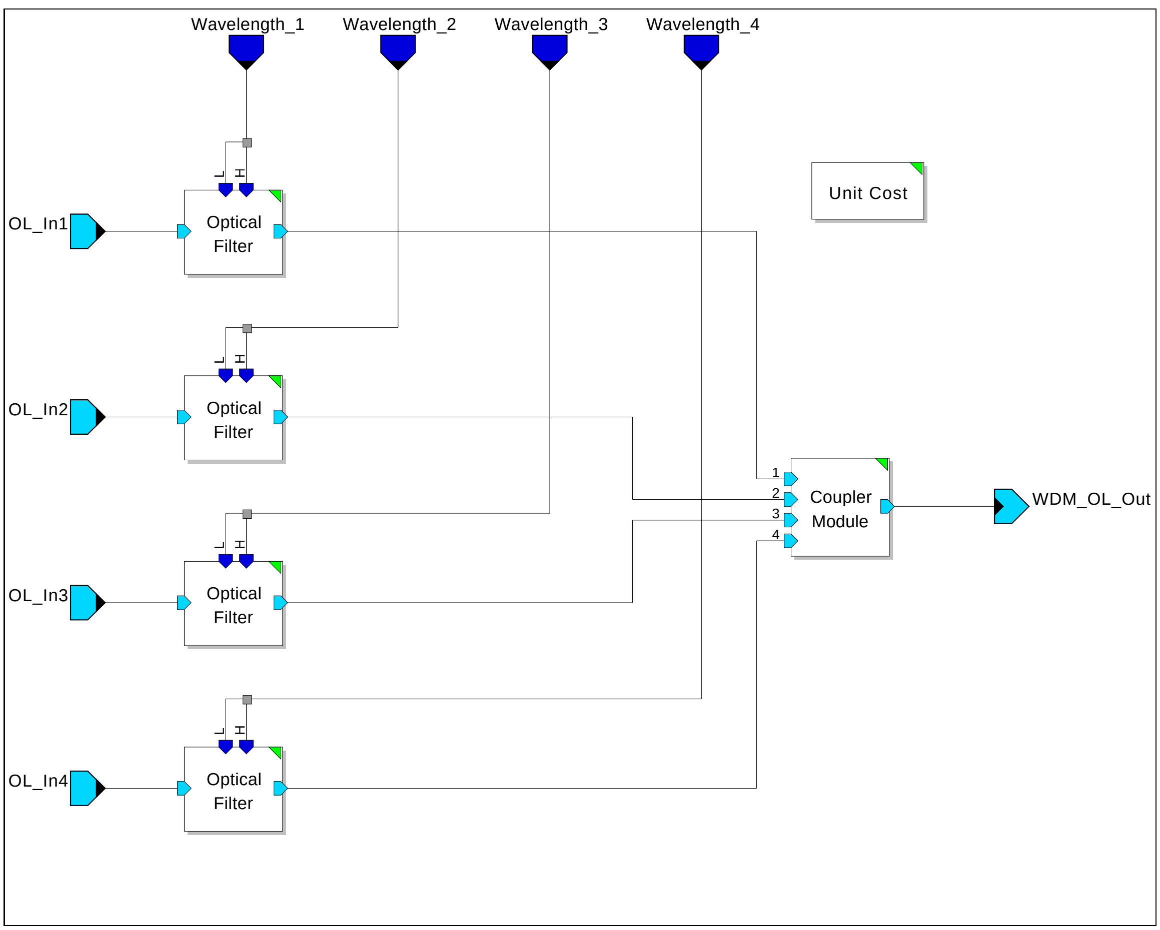

Figure 46: 4X1 MUX Module

Inputs
OL_In1
OL_In2
Wavelength_1
Wavelength_2
OL_In3
OL_In4
Wavelength_3
Wavelength_4

Outputs
WDM_OL_Out (datastruct:SystemDS:Root.Packet)

Parameters
Input_Coupling_Loss_CH2
Input_Coupling_Loss_CH1
Output_Coupling_Loss
Tune_Delay
Input_Coupling_Loss_CH3
Input_Coupling_Loss_CH4

8x1_MUX module in DE domain

This module performs optical MUX by specified wavelengths to be multiplexed. Each optical signal at input port (OL_ In1: OL_ In8) passes through optical filter to select wavelength to be multiplexed. Refer to “Optical Filter” for more details how to use this module. The filtered signals are multiplexed by using “8x1 Coupler” and place coupled signal at output port (WDM_ OL_ Out).

Inputs
OL_In1
OL_In2
OL_In3
OL_In4
OL_In5
OL_In6
OL_In7
OL_In8
Wavelength_1
Wavelength_2
Wavelength_3
Wavelength_4
Wavelength_5
Wavelength_6
Wavelength_7

Wavelength_8

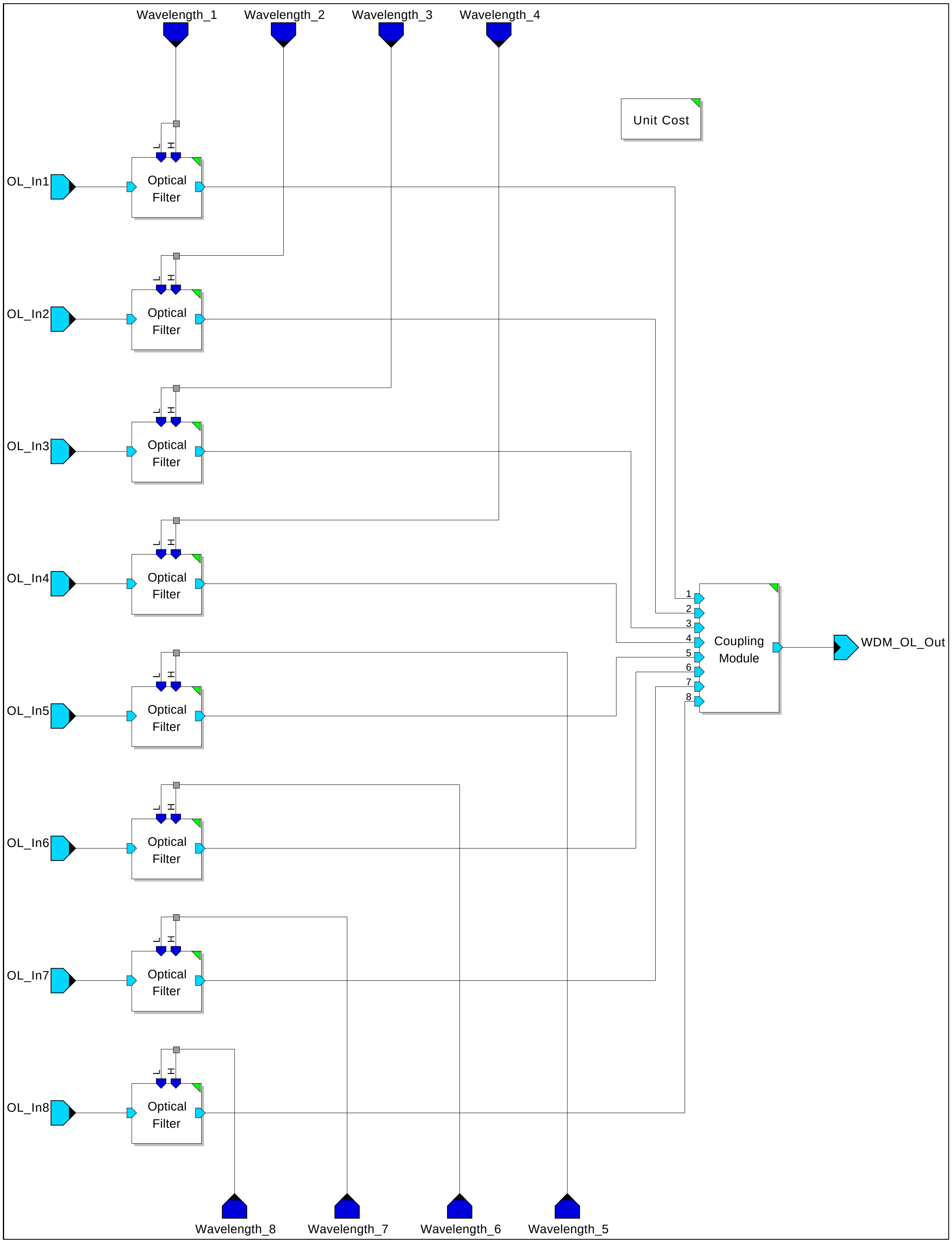

Figure 47: 1X8 MUX Module

Outputs
WDM_OL_Out (datastruct:SystemDS:Root.Packet)

Parameters
Input_Coupling_Loss_CH1
Input_Coupling_Loss_CH2
Input_Coupling_Loss_CH3
Input_Coupling_Loss_CH4
Input_Coupling_Loss_CH5
Input_Coupling_Loss_CH6
Input_Coupling_Loss_CH7
Input_Coupling_Loss_CH8
Tune_Delay
Output_Coupling_Loss

4.3.5 Optical Switch

1x2_Optical_Switch module in DE domain

This module routes incoming Optical Layer Data Structures (Single OL DS or WDM OL DS) to one of the two output ports. Routing decisions are made by an external control device and transmitted to the switch using Switch Control Data Structures. When a Switch Control DS is received it is checked to ensure that it is the correct type, if not intended for a 1x2 switch it is placed on the Incorrect Ctrl port and no action is taken. If the switch type is correct, the Control Vector is written into memory and used to configure the 1x2 Logical Switch primitive to route the Optical Layer Data Structures as desired. The routing configuration of a switch will remain the same until a Switch Control DS is received. The initial setup of the switch routes the input to the output 1 port. The switch does not support multicasting (the input can be switched to only one output).

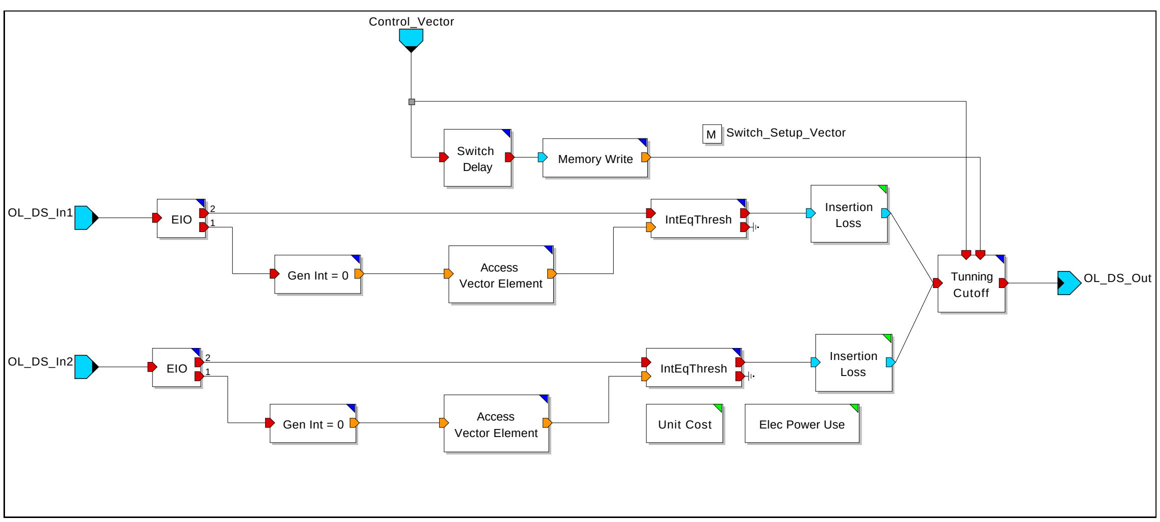

Figure 48: 1X2 Optical Switch Module

Inputs
OL_DS_In
Switch_Ctrl_In (datastruct:OPN_Version_1_0:Root.Switch_Control)

Outputs
OL_DS_Out1
OL_DS_Out2
Incorrect_Ctrl
datastruct:OPN_Version_1_0:Root.Switch_Control

Parameters
Insertion_Loss_dB
Switch_Delay

Memories
Switch_Setup_Vector

1x4_Optical_Switch module in DE domain

This module routes incoming Optical Layer Data Structures (Single OL DS or WDM OL DS) to one of the four output ports. Routing decisions are made by an external control device and transmitted to the switch using Switch Control Data Structures. When a Switch Control DS is received it is checked to ensure that it is the correct type, if not intended for a 1x4 switch it is placed on the Incorrect Ctrl port and no action is taken. If the switch type is correct, the Control Vector is written into memory and used to configure the 1x2 Logical Switch primitive to route the Optical Layer Data Structures as desired. The routing configuration of a switch will remain the same until a Switch Control DS is received. The initial setup of the switch routes the input to the output 1 port. The switch does not support multicasting (the input can be switched to only one output).

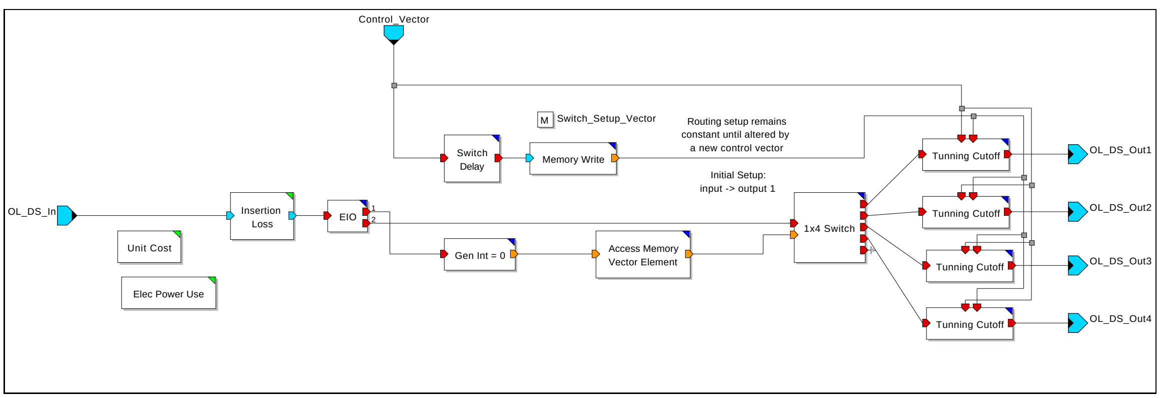

Figure 49: 1X4 Optical Switch Module

Inputs
OL_DS_In
Switch_Ctrl_In (datastruct:OPN_Version_1_0:Root.Switch_Control)

Outputs
OL_DS_Out1
OL_DS_Out2
OL_DS_Out3
OL_DS_Out4
Incorrect_Ctrl (datastruct:OPN_Version_1_0:Root.Switch_Control)

Parameters
Switch_Delay
Insertion_Loss_dB

Memories
Switch_Setup_Vector

2x2_Optical_Switch module in DE domain

This module routes Optical Layer Data Structures on the input ports (Single OL DS or WDM OL DS) to one of the two output ports. Routing decisions are made by an external control device and transmitted to the switch using Switch Control Data Structures. When a Switch Control DS is received it is checked to ensure that it is the correct type, if not intended for a 2x2 switch it is placed on the Incorrect Ctrl port and no action is taken. If the switch type is correct, the Control Vector is written into memory and used to configure the two 1x2 Logical Switch primitives to route the Optical Layer Data Structures as desired. The routing configuration of a switch will remain the same until a Switch Control DS is received. The initial setup of the switch routes input 1 to the output 1 port and input 2 to the output 2 port. The switch does not support multicasting (each input must be switched to one and only one output).

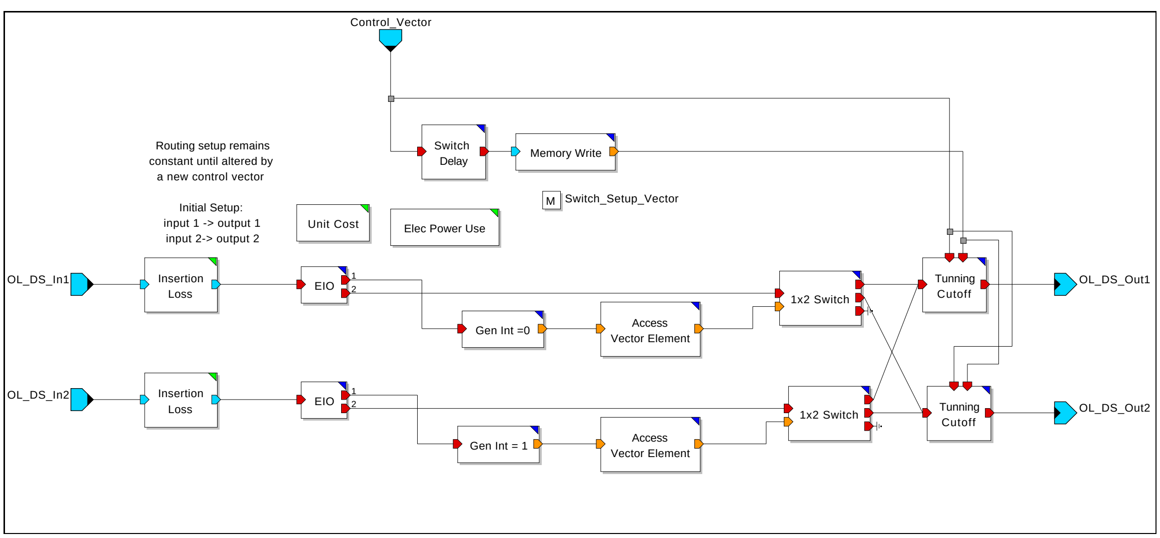

Figure 50: 2X2 Optical Switch Module

Inputs
OL_DS_In1
OL_DS_In2
Switch_Ctrl_DS_In (datastruct:OPN_Version_1_0:Root.Switch_Control)

Outputs
OL_DS_Out1
OL_DS_Out2
Incorrect_Ctrl (datastruct:OPN_Version_1_0:Root.Switch_Control)

Parameters
Switch_Delay
Insertion_Loss_dB

Memories
Switch_Setup_Vector (Root.IntVector)

2x4_Optical_Switch module in DE domain

This module routes Optical Layer Data Structures on the input ports (Single OL DS or WDM OL DS) to one of the four output ports. Routing decisions are made by an external control device and transmitted to the switch using Switch Control Data Structures. When a Switch Control DS is received it is checked to ensure that it is the correct type, if not intended for a 2x4 switch it is placed on the Incorrect Ctrl port and no action is taken. If the switch type is correct, the Control Vector is written into memory and used to configure the two 1x4 Logical Switch primitives to route the Optical Layer Data Structures as desired. The routing configuration of a switch will remain the same until a Switch Control DS is received. The initial setup of the switch routes input 1 to the output 1 port and input 2 to the output 2 port. The switch does not support multicasting (each input must be switched to one and only one output).

Inputs
OL_DS_In1
OL_DS_In2
Switch_Ctrl_In (datastruct:OPN_Version_1_0:Root.Switch_Control)

Outputs
OL_DS_Out1
OL_DS_Out2
OL_DS_Out3
OL_DS_Out4
Incorrect_Ctrl (datastruct:OPN_Version_1_0:Root.Switch_Control)

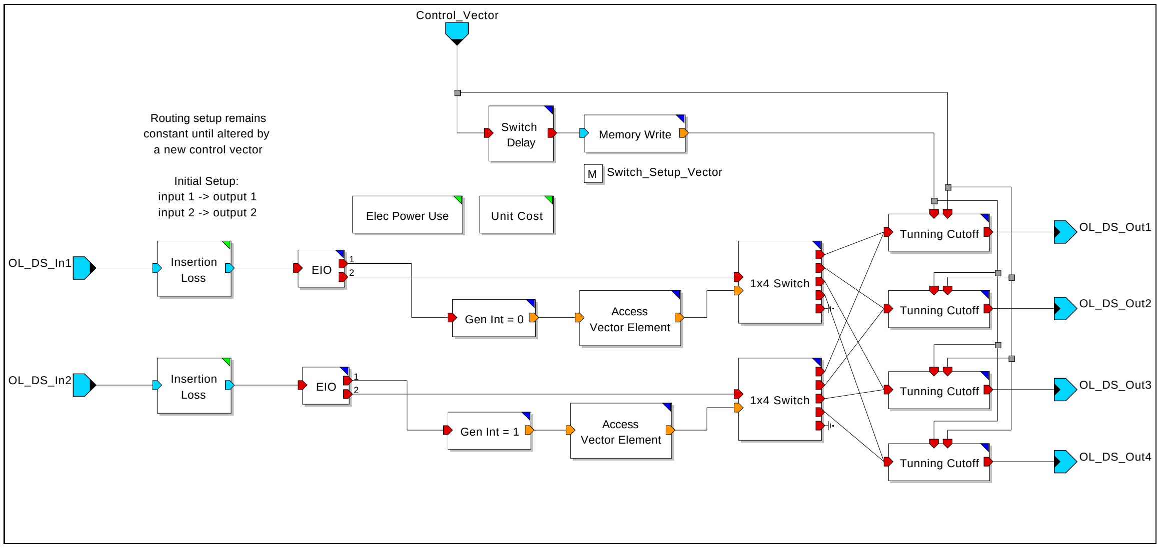

Figure 51: 2X4 Optical Switch Module

Parameters
Switch_Delay
Amount of time delay.
Insertion_Loss
Value in dB

Memories
Switch_Setup_Vector

<u>4x4_Optical_Switch module in DE domain</u>

This module routes Optical Layer Data Structures on the input ports (Single OL DS or WDM OL DS) to one of the four output ports. Routing decisions are made by an external control device and transmitted to the switch using Switch Control Data Structures. When a Switch Control DS is received it is checked to ensure that it is the correct type, if not intended for a 4x4 switch it is placed on the Incorrect Ctrl port and no action is taken. If the switch type is correct, the Control Vector is written into memory and used to configure the four 1x4 Logical Switch primitives to route the Optical Layer Data Structures as desired. The routing configuration of a switch will remain the same until a Switch Control DS is received. The initial setup of the switch routes input 1 to the output 1 port, input 2 to the output 2 port, input 3 to the output 3 port and input 4 to the output 4 port. The switch does not support multicasting (each input must be switched to one and only one output).

Inputs
OL_DS_In1
OL_DS_In2
OL_DS_In3
OL_DS_In4

Switch_Ctrl_In (datastruct:OPN_Version_1_0:Root.Switch_Control)

Outputs
OL_DS_Out1
OL_DS_Out2
OL_DS_Out3
OL_DS_Out4
Incorrect_Ctrl (datastruct:OPN_Version_1_0:Root.Switch_Control)

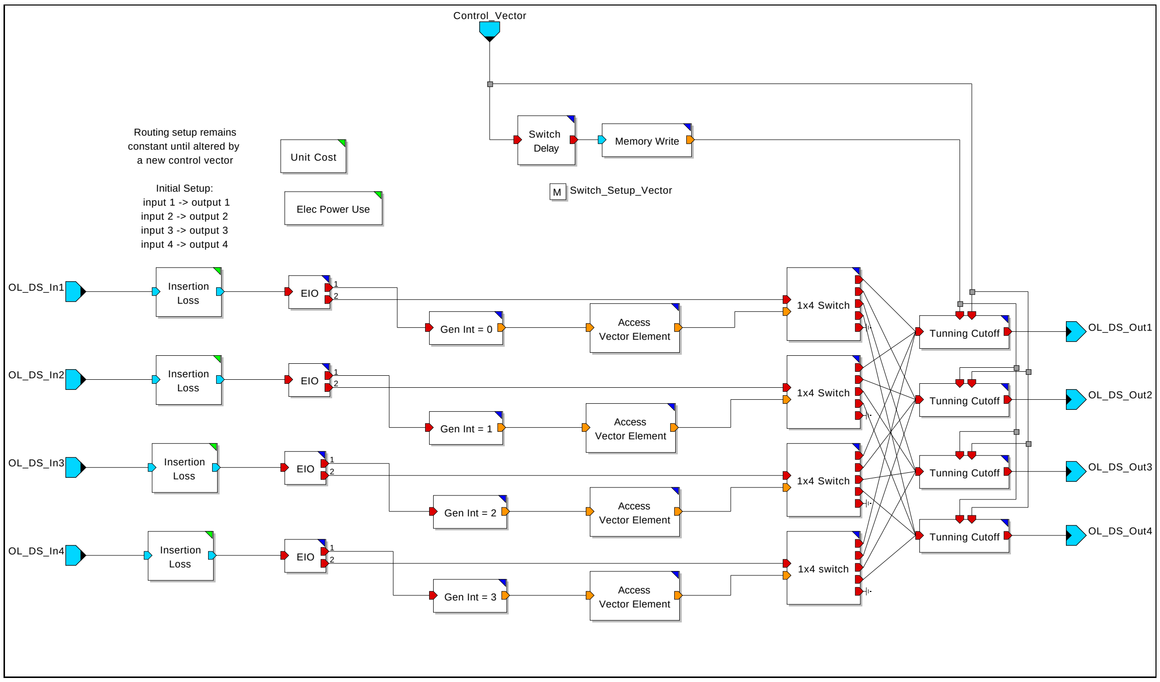

Figure 52: 4X4 Optical Switch Module

Parameters
Switch_Delay
Insertion_Loss_dB

Memories
Switch_Setup_Vector

4.3.6 Splitters

<u>1x2_Splitter module in DE domain</u>

This module splits optical input signal to two output signals. Input coupling loss and output coupling loss are included with splitter. These losses are specified in dB of each port. The splitting ratio can be defined individually for each channel. The effects of splitting can be included by adding function blocks between input and output. For this version, “CrossWaveInterference”, “OpticalBitErrorInjection” and “SplittingDelay” present the effects of splitting.

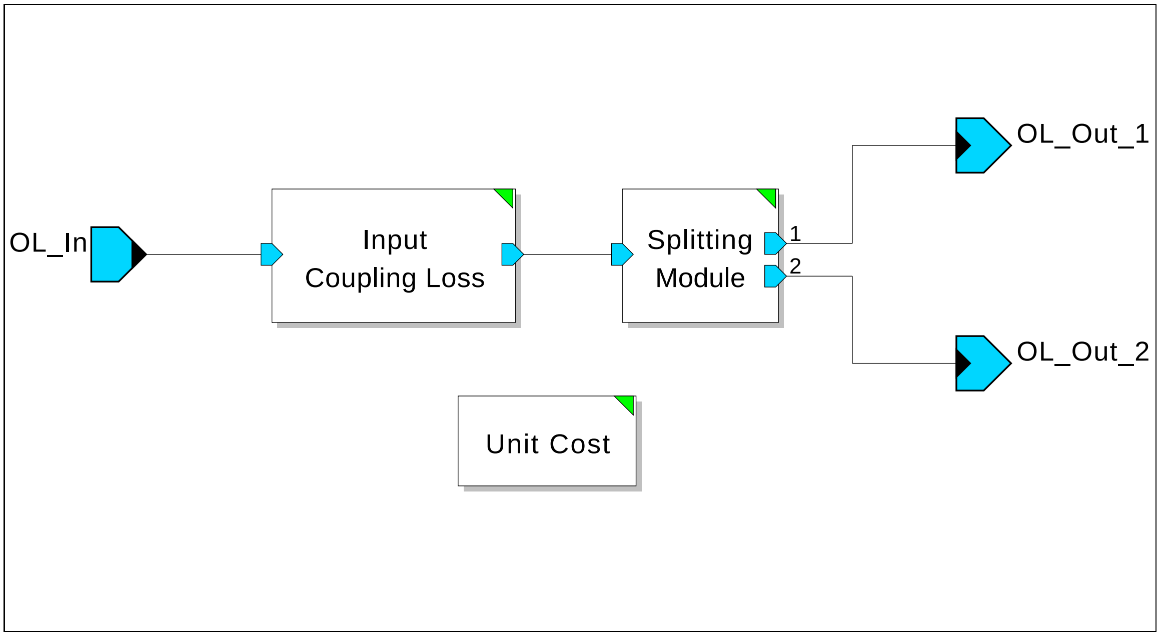

Figure 53: 1X2 Splitter Module

Inputs
OL_In
datastruct:OPN_Version_1_0:Root.OpticalLayer

Outputs
OL_Out_1
datastruct:OPN_Version_1_0:Root.OpticalLayer
OL_Out_2
datastruct:OPN_Version_1_0:Root.OpticalLayer

Parameters
Input_Coupling_Loss
Output_Coupling_Loss_1
Output_Coupling_Loss_2
CH1_Ratio
Splitter_Delay
CH2_Ratio

1x4_Splitter module in DE domain

This module splits optical input signal to four output signals. Input coupling loss and output coupling loss are included with splitter. These losses are specified in dB of each port. The splitting ratio can be defined individually for each channel. The effects of splitting can be included by adding function blocks between input and output. For this version, “CrossWaveInterference”, “OpticalBitErrorInjection” and “SplittingDelay” 9 present the effects of splitting.

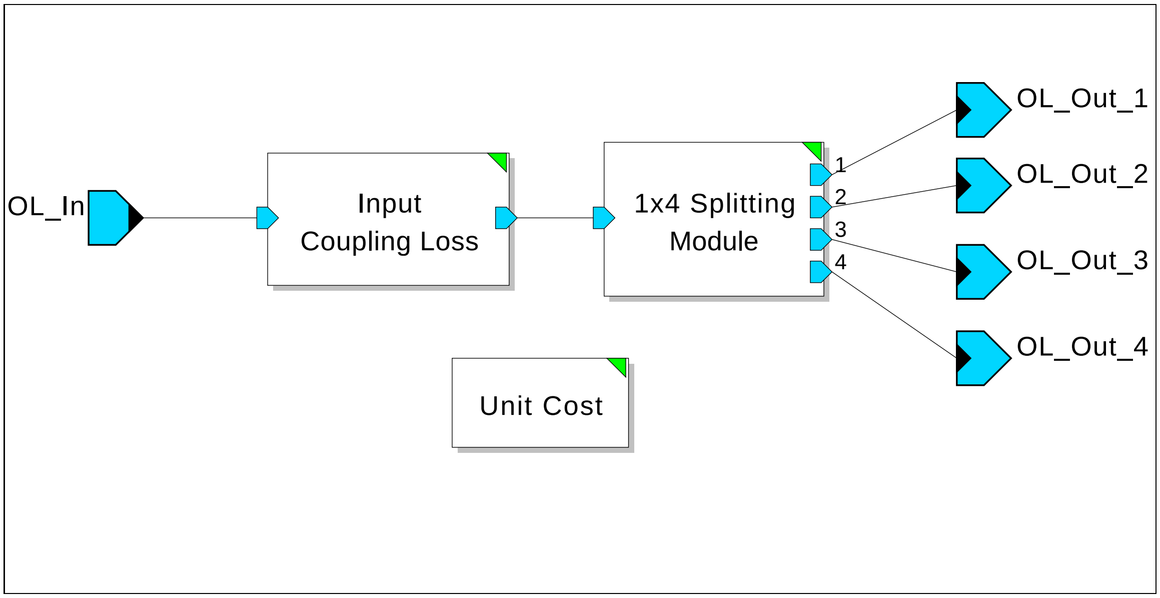

Figure 54: 1X4 Splitter Module

Inputs
OL_In

Outputs
OL_Out_1
OL_Out_2
OL_Out_3
OL_Out_4

Parameters
Input_Coupling_Loss
CH1_Ratio
CH2_Ratio
CH3_Ratio
CH4_Ratio
Output_Coupling_Loss_CH3
Output_Coupling_Loss_CH1
Output_Coupling_Loss_CH2
Output_Coupling_Loss_CH4
Splitter_Delay
Amount of time delay
Output_Coupling_Loss_CH1
Output_Coupling_Loss_CH2
Output_Coupling_Loss_CH3
Output_Coupling_Loss_CH4

1x8_Splitter_Module module in DE domain

This module splits optical signal either single or WDM OL from one input port (OL_ In) to eight output ports (OL_ Out_ 1: OL_ Out_ 8). The power level of each output channel is defined by individual splitting ratio (CH1_ Ratio: CH8_ Ratio).

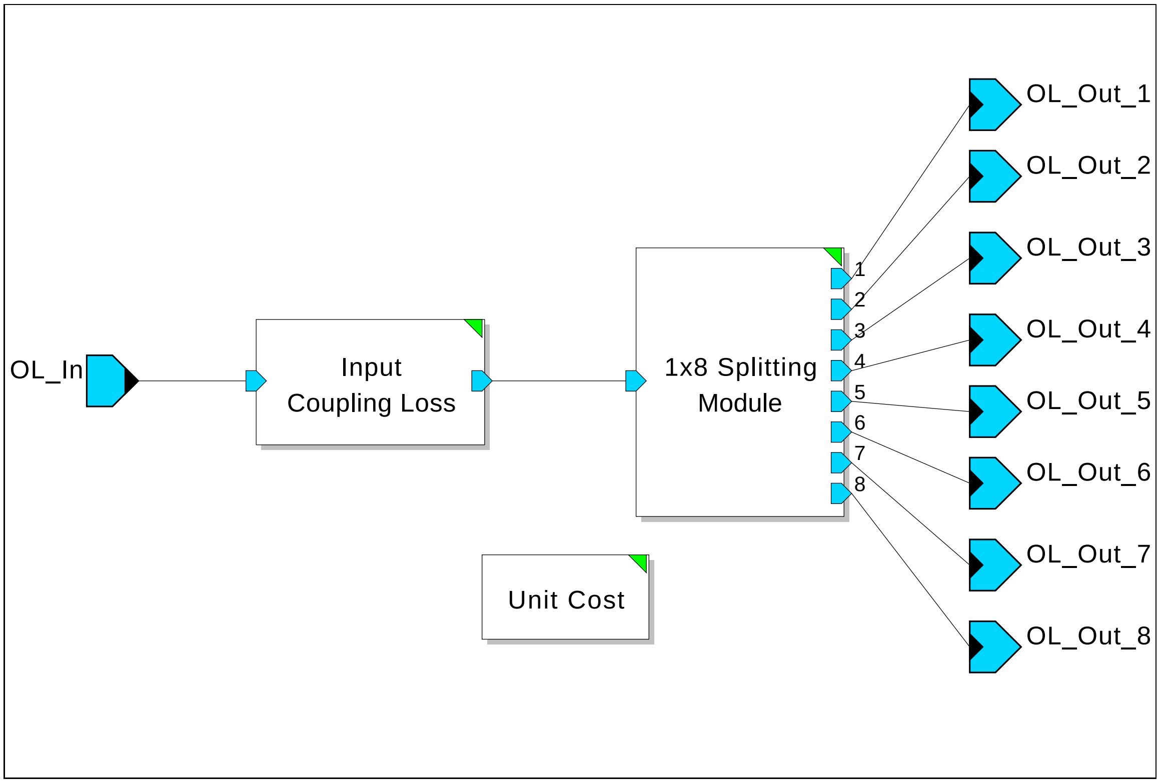

Figure 55: 1X8 Splitter Module

Inputs
OL_In

Outputs
OL_Out_1
OL_Out_2
OL_Out_3
OL_Out_4
OL_Out_5
OL_Out_6
OL_Out_7
OL_Out_8

Parameters
CH1_Ratio
CH2_Ratio
CH3_Ratio
CH4_Ratio

CH5_Ratio
CH6 Ratio
CH7Ratio
CH8_Ratio

4.3.7 Other Modules

Discrete_Tunable_Opt_Transmitter module in DE domain

This module emulates the functionality of many real- world tunable optical transmitters in that it has a set number of discrete wavelengths to choose from. In its default configuration this module has four wavelengths to choose from (1528.77nm, 1529.55nm, 1530.33nm, 1531.12nm). The output wavelength is selected by placing a 0, 1, 2 or 3 (corresponding to the index of the wavelength in the vector of values) on the Change_ Transmit_ Wavelength port. The initial output wavelength is in the 0 position (1528.77nm). The number and value of available wavelengths can be altered (See Discrete Tunable Laser Source). All other aspects of this modules operation are identical to the Optical Transmitter. Data and Number of Bits are input and a Single Optical Layer Data Structure is created.

Inputs

Data_In
Change_Transmit_Wavelength
Number_of_Bits

Outputs

Single_OL_DS_Out
datastruct:OPN_Version_1_0:Root.OpticalLayer.Single_OL

Parameters

DS_ID_Num_Start
Switching_Time
Opt_Power_Level
Opt_Coupling_Loss
Laser_Power_Fluctuation_Vector (Array of dB values which will be subtracted from Opt_Power_Level. The values represent one period of the waveform)

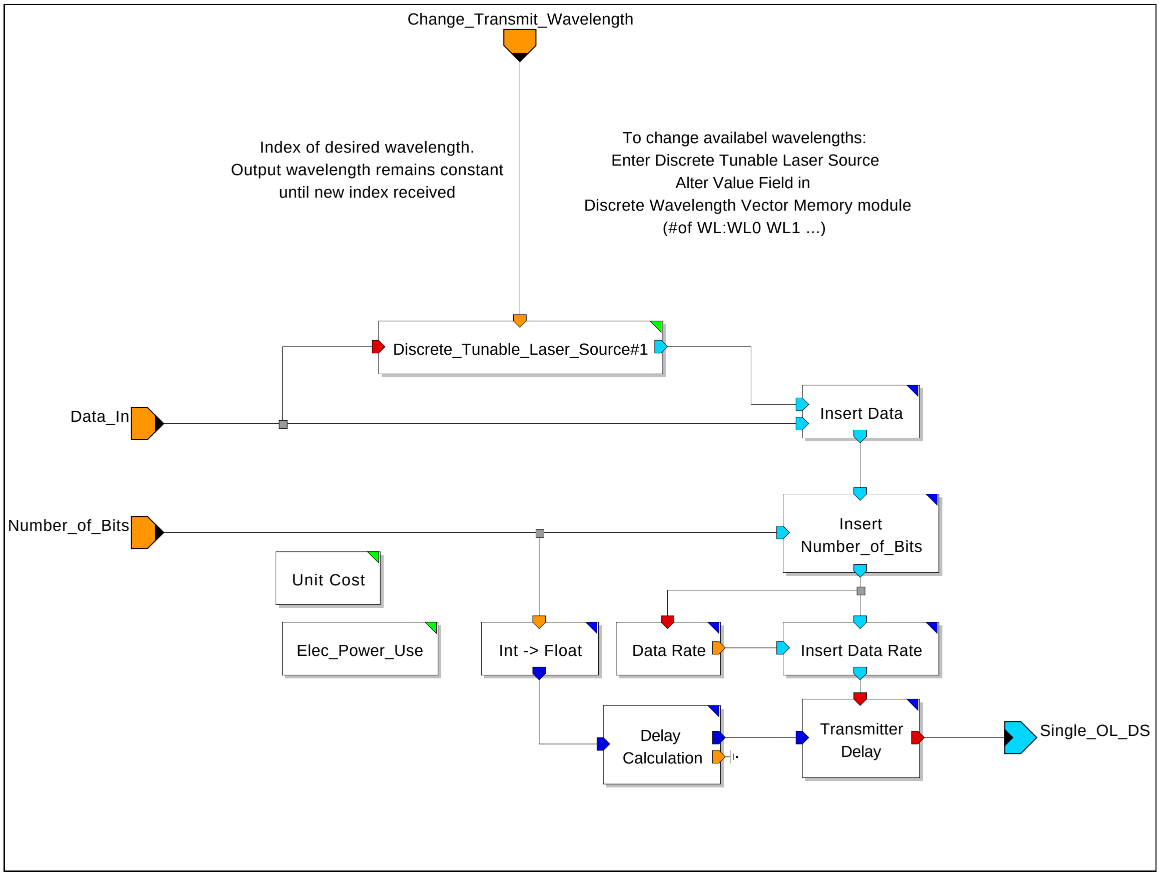

Figure 56: Discrete Tunable Optical Transmitter Module

Optical_Connector module in DE domain

This module implements simple optical connector. Only one characteristic is included in this module 18 ; optical power loss or insertion loss which is defined in dB. Input port can be either Single OL or WDM OL structure.

Inputs

In

Outputs

Out

Parameters

Insertion_Loss

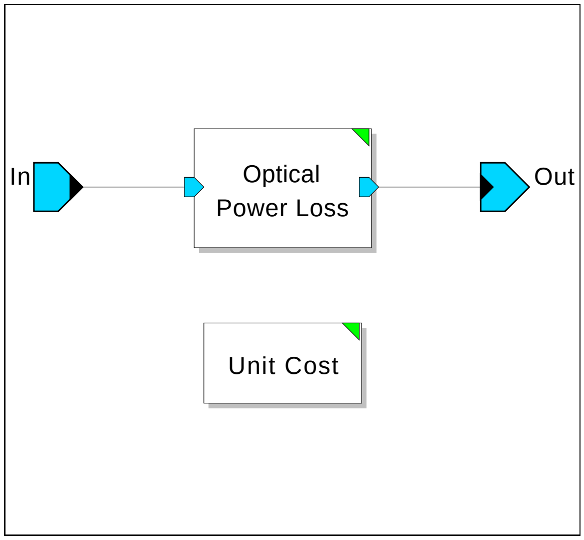

Figure 57: Optical Connector Module

Optical_Fiber module in DE domain

This module performs an optical fiber link with specified length of fiber. This fiber includes some characteristics; power loss, propagation delay, cross wave interference. There is a case that propagation delay is greater than transmission delay so that more than one packet is traveling in the link. FIFO block is used to store packets which are traveling on the link. Refer to parameters setting for details. FIFO will overflow if number of packet on the link is greater than desired packets. The error message is displayed and simulation is aborted in case of FIFO overflow.

Inputs

OL_In

Outputs

OL_Out

Parameters

Power_Loss_Per_Length
Length
Delay_Per_Length
Maximum_Flits_in_Fiber

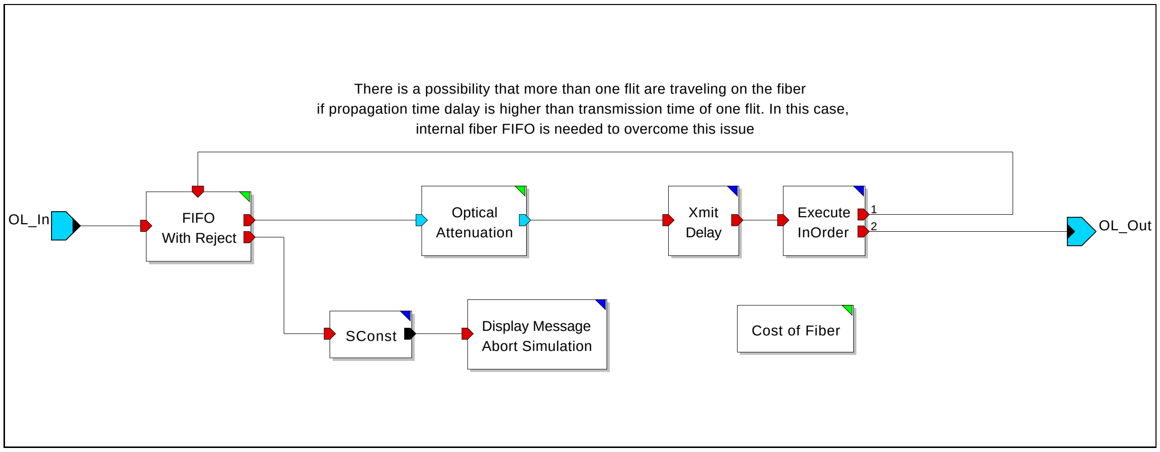

Figure 58: Optical Fiber Module

Optical_Filter module in DE domain

This module performs optical filter by specified low and high boundary of filter. This filter is band- pass filter.Input signal can be either Single OL or WDM OL. The boundaries are tunable and there is a delay associated with wavelength tuning. Any wavelength which falls in boundaries or equal to boundaries is passed through output port. If any wavelength has power lower than power threshold, it will be dropped by “Low Power Drop” module.

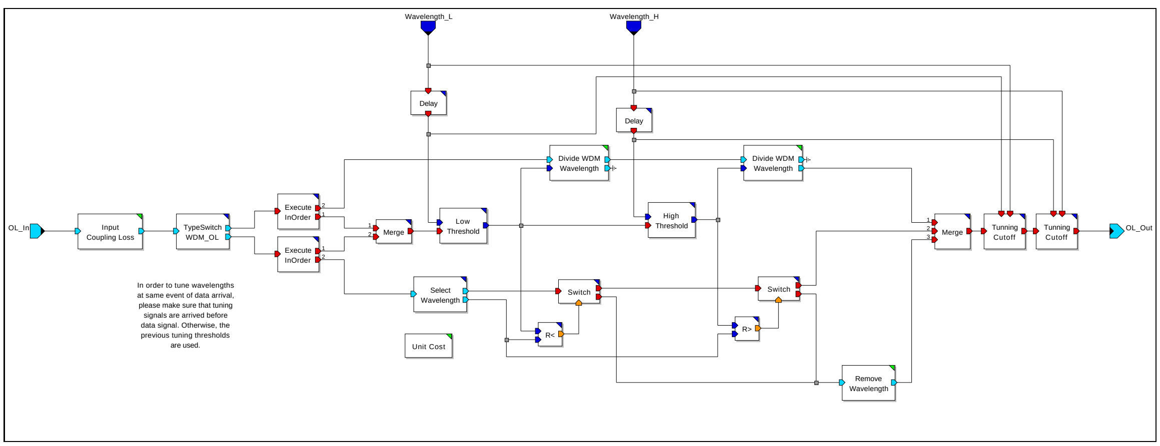

Figure 59: Optical Filter Module

Inputs

OL_In
Wavelength_L
Wavelength_H

Outputs

OL_Out
Low_Power

Parameters
Input_Coupling_Loss
Tune_Delay
Init_Tune_Low
Init_Tune_High
Power_Threshold
Output_Coupling_Loss

Optical_Invert_Filter module in DE domain

This module performs optical invert filter by specified low and high boundary of filter. This filter is band-reject filter. Input signal can be either Single OL or WDM OL. The boundaries are tunable and there is a delay associated with wavelength tuning. Any wavelength which locates outside boundaries is passed through output port. If any wavelength has power lower than power threshold, it will be dropped by "Low Power Drop" module.

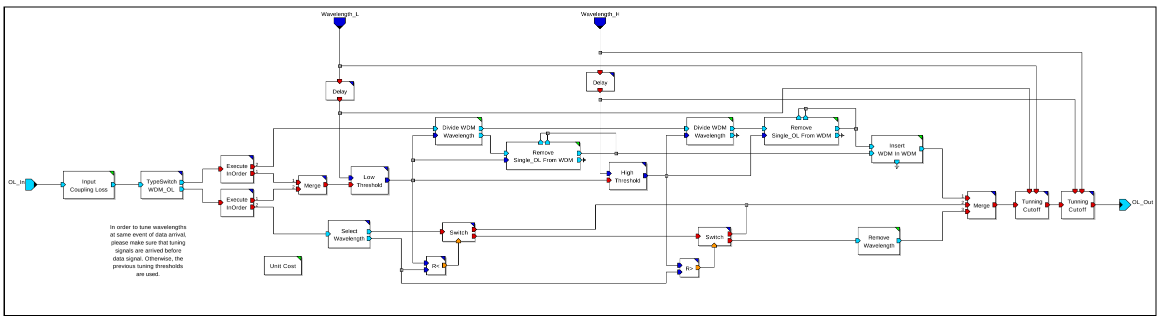

Figure 60: Optical Invert Filter Module

Inputs
OL_In
Wavelengt h_L
Wavelength_H

Outputs
OL_Out
Low_Power

Parameters
Input_Coupling_Loss
Tune_Delay
Init_Tune_Low
Init_Tune_High
Power_Threshold
Output_Coupling_Loss

Optical_Power_Amplifier module in DE domain

This module performs optical power amplifier. Input signal can be either Single OL or WDM OL. Input coupling loss or insertion loss and output coupling loss are included in this module. Power gain is given in unit of dB. The effects of amplifier can be represented by adding functional blocks between input and output port. For this version, “CrossWaveInterference” and “OpticalBitErrorInjection” are included for example.

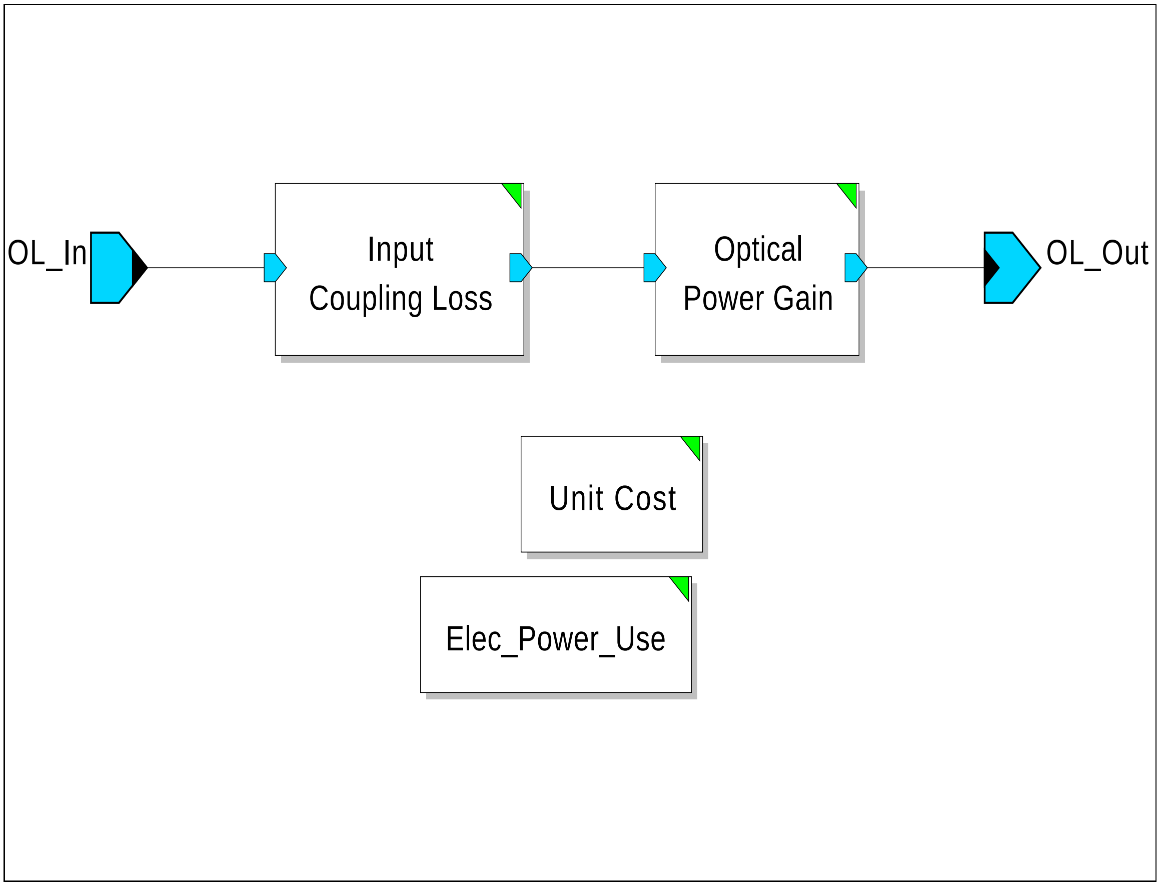

Figure 61: Optical Power Amplifier Module

Inputs
OL_In

Outputs
OL_Out

Parameters
Input_Coupling_Loss
Output_Coupling_Loss
Power_Gain

Optical_Transmitter module in DE domain

This module creates a single optical wavelength with specified wavelength. Single OL is created corresponding to “Data” and “Bits” at input ports. Wavelength of signal is given by module parameter at “Laser Source”. Refer to “Laser Source” for more details of each parameter.

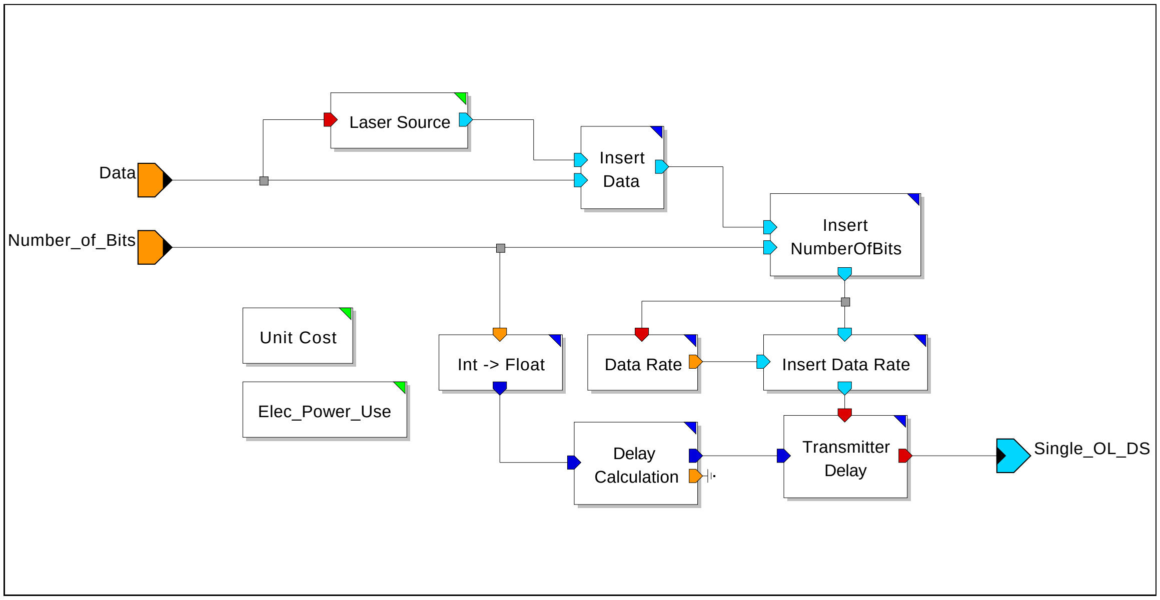

Figure 62: Optical Transmitter Module

Inputs
Data
Bits

Outputs
Single_OL

Parameters
Starting_ID_Number
Laser_Wavelength
Output_Coupling_Loss
Laser_Power_Fluctuation_Vector
Opt_Power_Level

Optical_Tunable_Receiver module in DE domain

This module performs optical receiver with tunable wavelength. Either Single OL or WDM OL can place at input port. To make receiver tunable, optical filter is exploited at front end. If selected wavelength has power less than power threshold in “Optical Filter” module, the warning message is given at console and this wavelength will be dropped. If input signal is Single OL structure, it will by pass through and extract field for data. Otherwise, only first index of WDM vector is extracted for data. If selected wavelength is undefined value of wavelength (default), it will be dropped before extracting data.

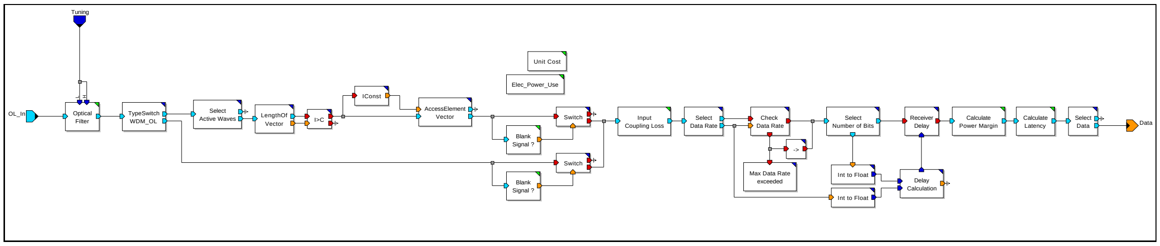

Figure 63: Optical Tunable Receiver Module

Inputs
OL_In
Tuning

Outputs
Data

Parameters
Input_Couping_Loss
Tune_Delay
Init_Tune_Wavelength
Power_Threshold

Tunable_Optical_Transmitter module in DE domain

This module has the same basic functionality as the Optical Transmitter. It inputs Data and the Number of Bits and produces a Single Optical Layer Data Structure. The added functionality of this module is the ability to change the transmitting wavelength during simulation run time. An initial wavelength is set and remains until a new wavelength value in nanometers is placed on the Change_ Transmit_ Wavelength port. The actual wavelength switch over is delayed by the amount of time specified by the “Switching_ Time” parameter.

Inputs
Data_In
Change_Transmit_Wavelength
Number_of_Bits

Outputs
Single_OL_DS_Out

Parameters
Switching_Time
Output_Coupling_Loss

Initial_Wavelength
Laser_Power_Fluctuation_Vector
Opt_Power_Level

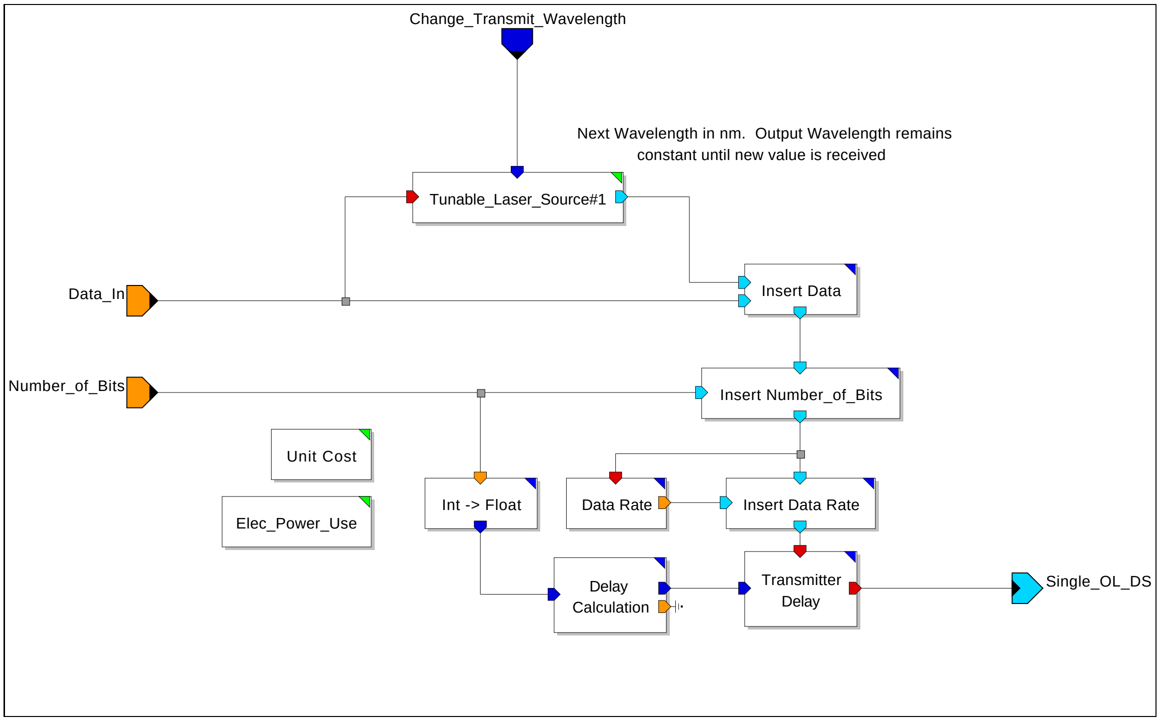

Figure 64: Tunable Optical Transmitter Module

4.3.8 TDM Components

TDM_DEMUX emux module in DE domain

This module receives TDM data structures and places the data payload on the output which corresponds to the TDM data structures time slot (Time Slot 1 placed on Data1 output port,...)

Inputs

TDM_DS

Outputs

Data1
Data2
Data3
Data4

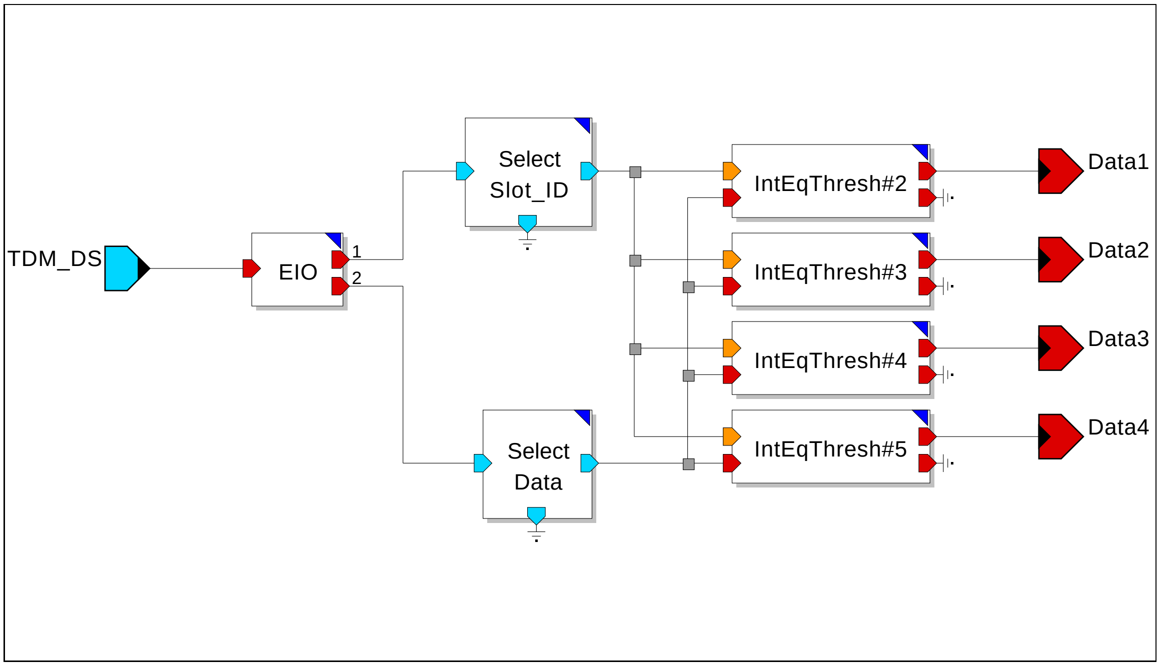

Figure 65: TDM DEMUX Module

TDM_MUX module in DE domain

This module represents a Synchronous Time-Division Multiplexer. A single input port is sampled each time the Clock input is triggered. The module samples the ports in order (1,2,3,4,1...) and places the sampled input in a TDM data structure with a time slot identifier equal to the input port number (Data1->Slot 1). If a port does not have data to be sampled during its time slot no data structure is produced (the slot remains empty). Data can arrive at the ports asynchronously, but must wait for its slot to come around before being transmitted. Data can arrive no faster than 1/4 of the clock frequency. Faster data frequencies will experience loss (data will be overwritten before it can be transmitted).

Inputs

Clock
Data1
Data2
Data3
Data4

Outputs

Bits_Per_Slot

Parameters

Bits_per_Slot

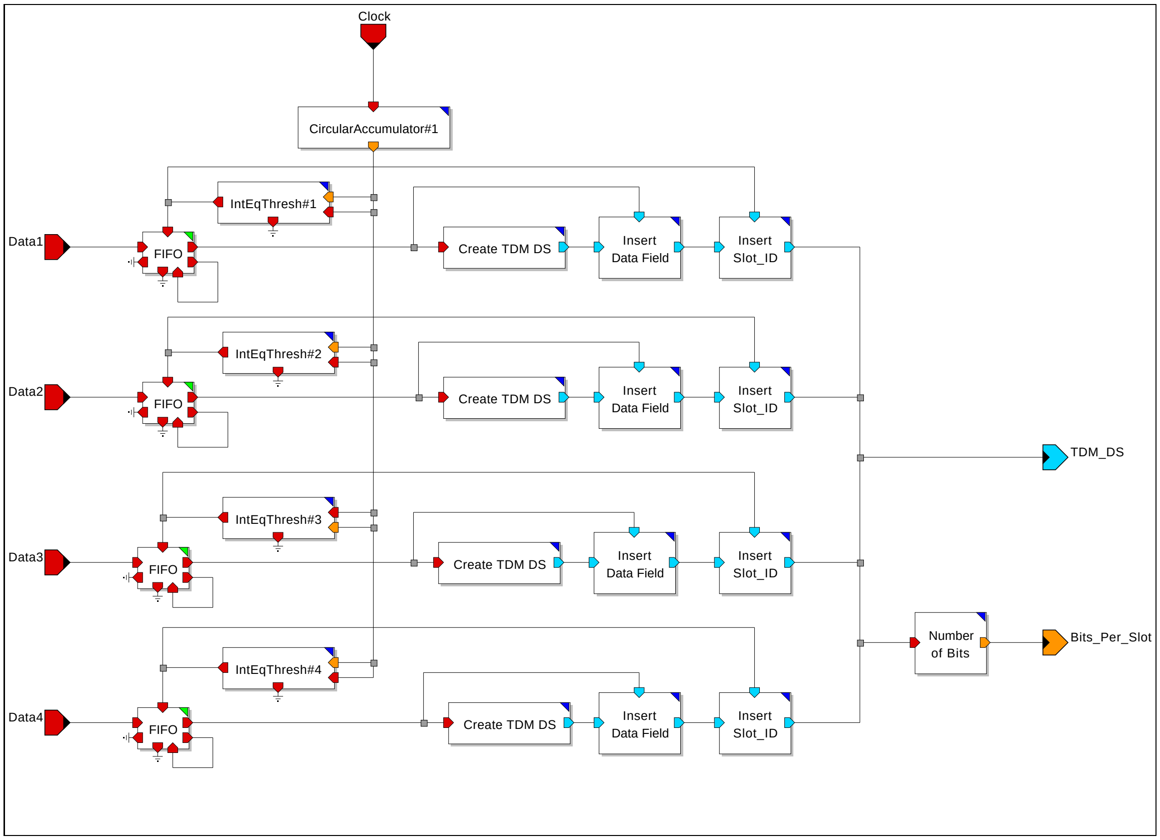

Figure 66: TDM MUX Module

TDM_Receiver module in DE domain

This module searches for a TDM DS with a given time slot. The data payload is extracted and placed on the Data output port. The module is tunable, the received time slot can be changed from the Initial_Slot_ID by placing the integer of the new time slot on the Change_Slot_ID port. This device accepts Integer type Control Data Structures.

Inputs

Change_Slot_ID
TDM_DS

Outputs

Data

Parameters

Elec_Power_Use
Initial_Slot_ID
Unit_Cost

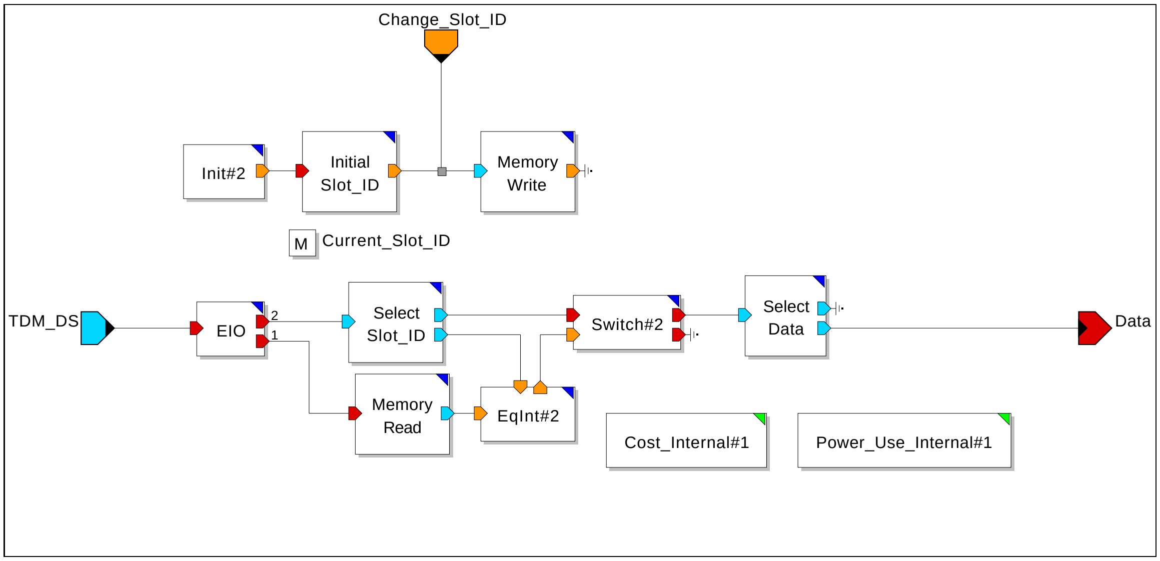

Figure 67: TDM Receiver Module

Memories

Current_Slot_ID

4.3.9 Simulation Components

Control interface float

This module will check the ID field of a Control DS to see if it is intended for its component. If it is it will pass the float value to the component port. If not it will allow the Control DS to pass through. See the online documentation for individual components to see if they accept Control DS with float values.

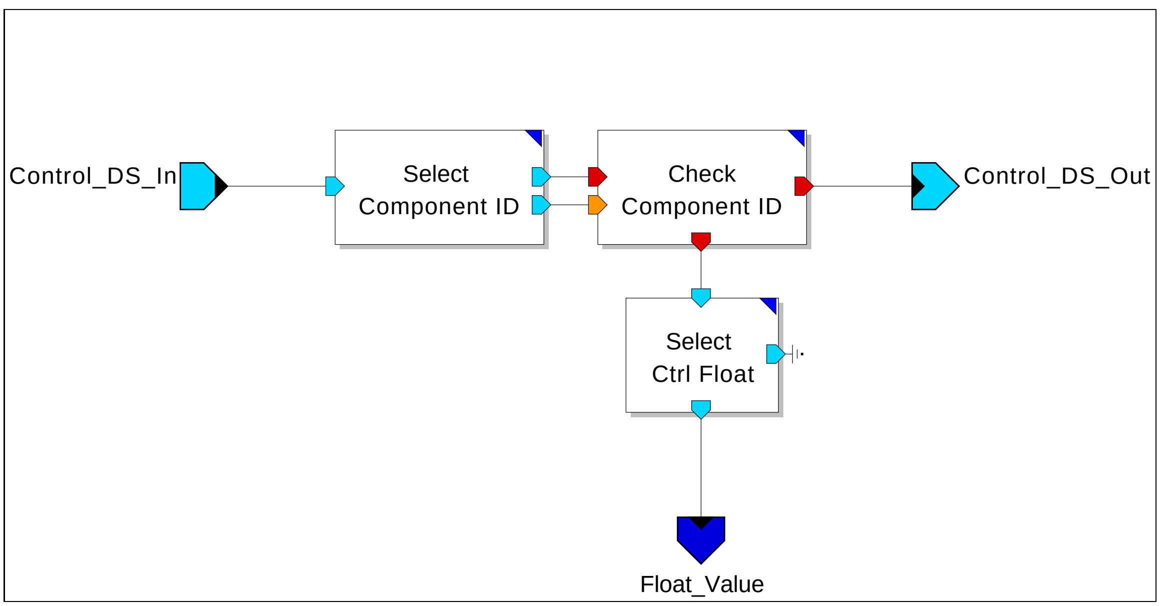

Figure 68: Control Interface Float Module

Inputs
Control_DS_In

Outputs
Control_DS_Out
Float_value

Parameters
Component_ID

Control Interface Integer

This module will check the ID field of a Control DS to see if it is intended for its component. If it is it will pass the integer value to the component port. If not it will allow the Control DS to pass through. See the online documentation for individual components to see if they accept Control DS with integer values.

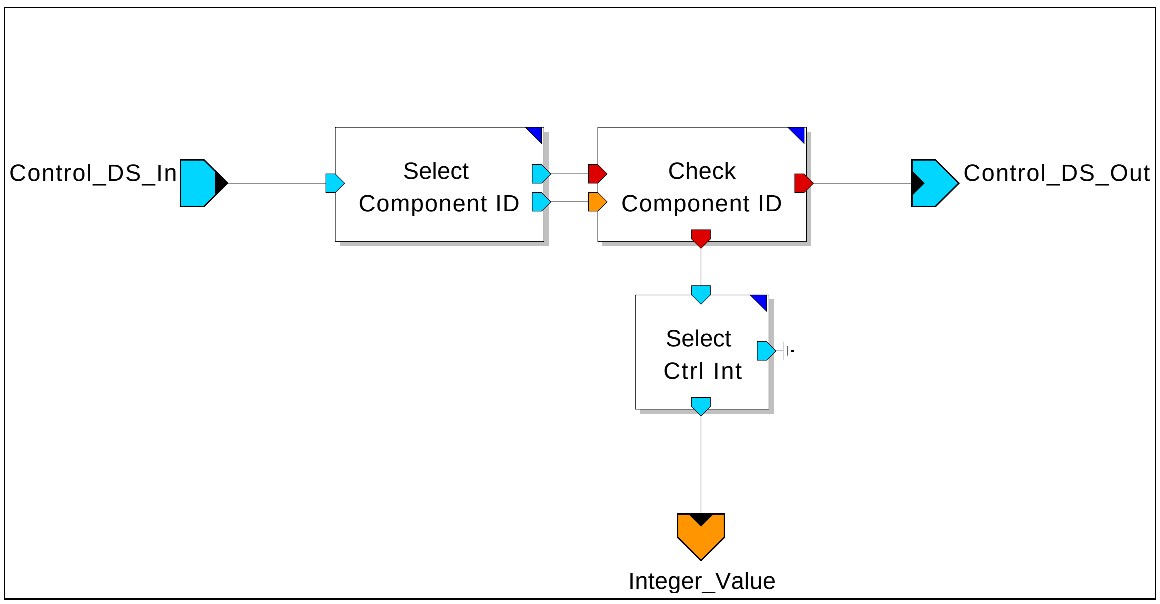

Figure 69: Control Interface Integer Module

Inputs
Control_DS_In

Outputs
Control_DS_Out
Integer_Value

Parameters
Component_ID

<u>Control Interface Vector</u>

This module will check the ID field of a Control DS to see if it is intended for its component. If it is it will pass the vector value to the component port. If not it will allow the Control DS to pass through. See the online documentation for individual components to see if they accept Control DS with vector values.

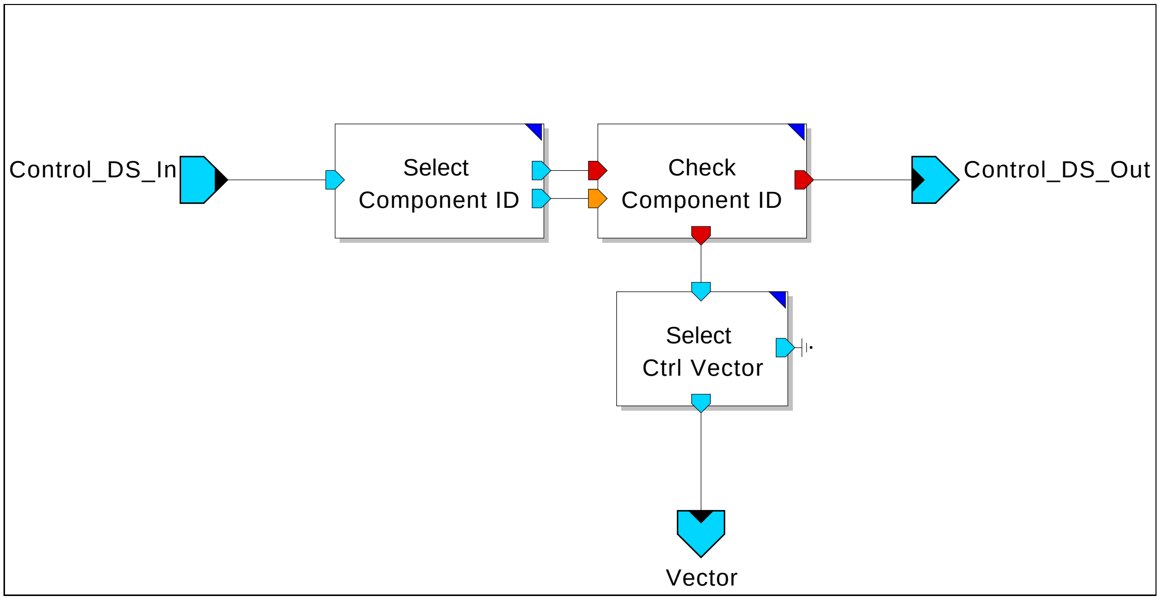

Figure 70: Control Interface Vector Module

Inputs

Control_DS_In

Outputs

Control_DS_Out
Vector

Parameters

Component_ID

<u>Control Unit</u>

The Control Unit allows tunable components such as tunable receivers and optical switches to change there settings during a simulation. This is accomplished through the use of the Control Data Structure and a user defined ASCII file. The Control Unit works in conjunction with Control Interface Modules (see online documentation) which are connected to the component which is to be controlled and are given a unique ID that will allow them to search for Control Data Structures intended for their component. The type of Control Interface used (float, integer, or vector) must be compatible with the control data type of the device it is to control. For example an optical switch requires vector control information while an OADM requires float type. See the online documentation for individual components to see if they can accept control information and of what type.

The ASCII file contains string representations of the Control Data Structures (See MLD manual for explanation of string representation). The first field of the DS contains the ID of the Control Interface Module that the control is intended for. The following three fields can contain vector, float and integer data respectively. (see Control Data Structure online documentation. It is essential that the proper type of control information is passed to the proper ID or errors will occur. The format of the ASCII file is as follows:

LION_ver_1_0_r01:Root.Control{ID,vector,float,int}

Sample File:
LION_ver_1_0_r01:Root.Control{1,2:2 1,0.0,0}
LION_ver_1_0_r01:Root.Control{5,0:0,1555.2,0}
LION_ver_1_0_r01:Root.Control{9,0:0,0.0,4}

The Control Unit will read one line each time it is triggered and produce a Control Data Structure which can be output into the system. In the above example the first trigger will read the first line and produce a DS intended for Control Interface Module #1 and will deliver a two element vector with elements 2 and 1. The second trigger will read the second line and produce a DS intended for Control Interface Module ID#5. It will pass the float value 1555.2. The third line will pass the integer value 4 to ID number 9. The number of total lines is limited only by simulation time. When the end of the file is reached DS will not be produced even if the Control Unit Module is triggered.

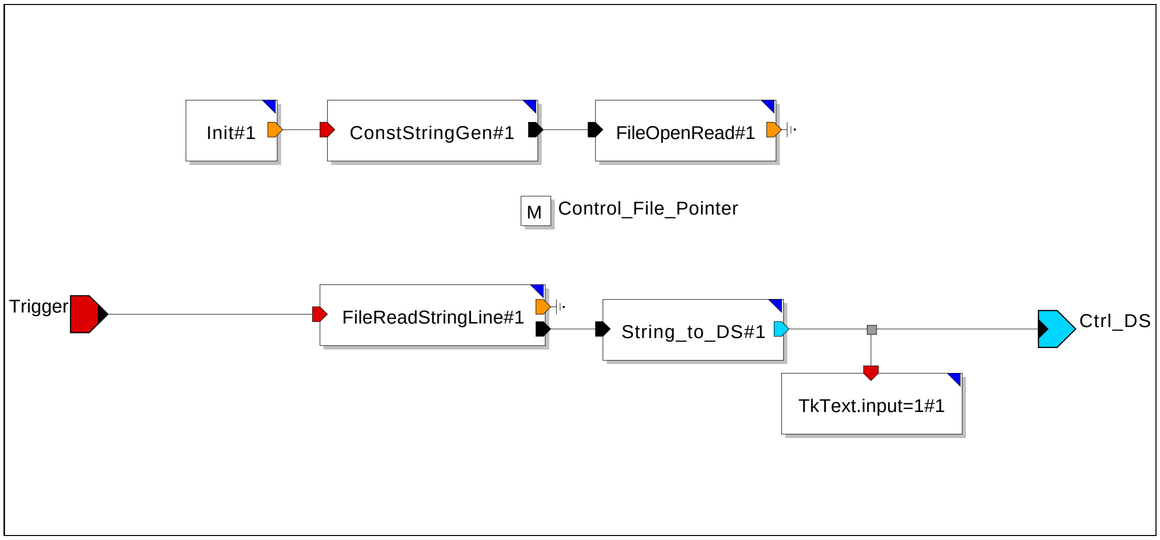

Figure 71: Control Unit Module

Note: Some ASCII editors add hidden characters which can disrupt the translation from string representation to actual data structure. The author suggests using the vi editor in the Linux environment or a similar editor.

Inputs
Trigger

Outputs
Ctrl_DS

Parameters
File_Path

Memories
Control_File_Pointer

Cost Display

Module displays the total cost of the system which was tallied in the Cost global memory. This block must be present in all systems or an error will be generated during simulation.

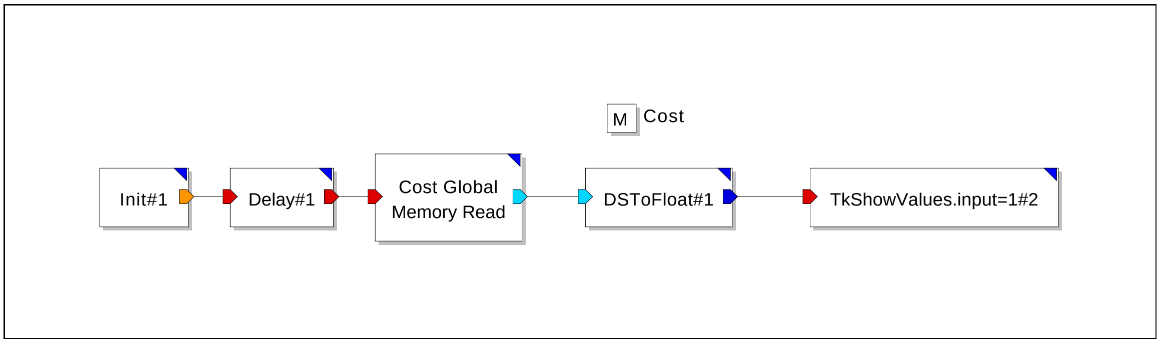

Figure 72: Cost Display Module

Memories
Cost

Electric Power Use Displayed

Module displays the total electrical power use (W) of the system which was tallied in the Power global memory. This block must be present in all systems or an error will be generated during simulation.

Memories
Power Description

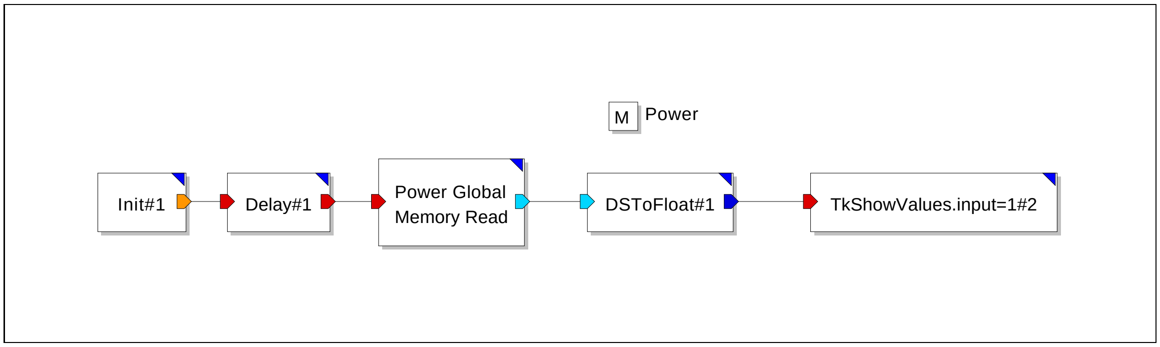

Figure 73: Electric Power Use Display Module

Graph Single OL

Graphs all Fields of Single OL DS with respect to the time they arrive.

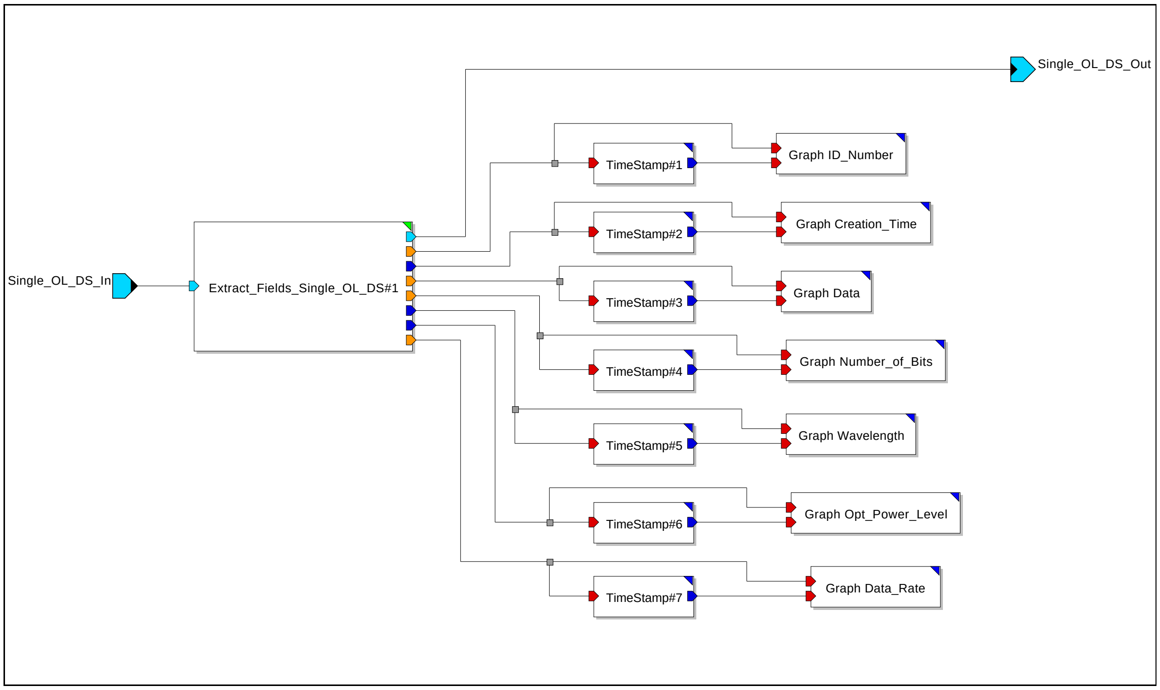

Figure 74: Graph Single OL Module

Inputs

Single_OL_DS_In

Outputs

Single_OL_DS_Out (Graphed)

Graph WDM OL

Graphs the Fields of a WDM OL D**S.**

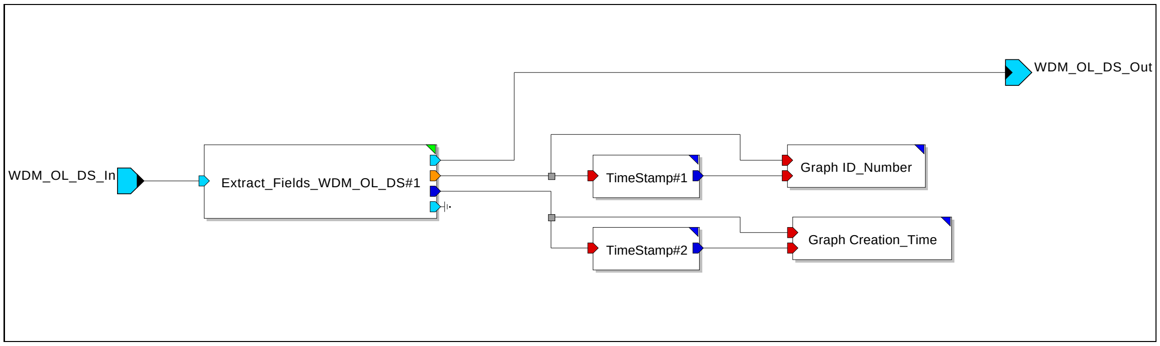

Figure 75: Graph WDM OL Module

Inputs
WDM_OL_DS_In

Outputs
WDM_OL_DS_Out (Graph)

Optical Power Probe

By placing instances of this module along a network path the attenuation experienced by a signal (Single_OL DS) moving along that path can be measured. The measurements are written to a user defined text file and can be imported into a spreadsheet program to produce a graph that represents the Power Budget of the link.

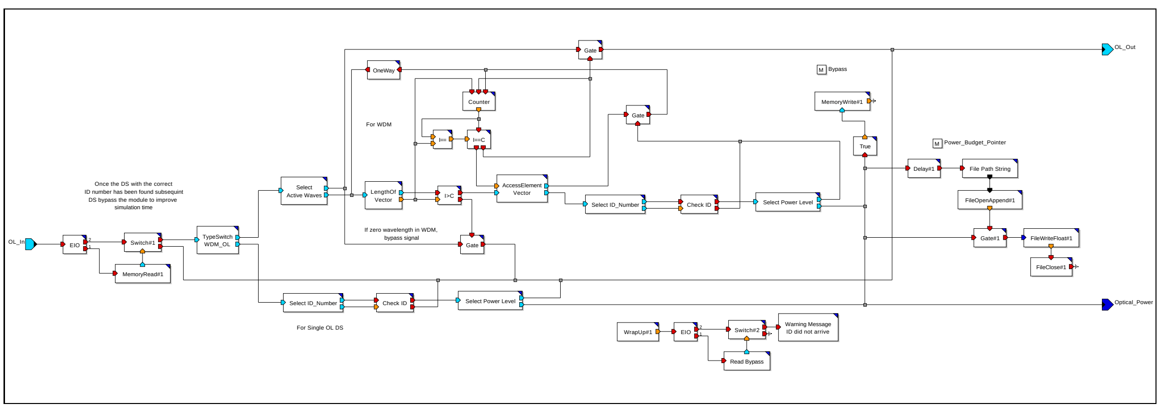

Figure 76: Optical Power Probe Module

The module scans incoming Data Structures for a certain Single Optical ID number. Upon finding the intended DS its optical power level is appended into a specified file. The power level is also placed on the Optical Power output port. Once the desired ID has been found all future data structures will be allowed to flow through the device without being processed. If the expected ID does not arrive by the end of the simulation a warning is given. Probes searching for different ID numbers can coexist in the same system as long as they write to different text files.

To ensure that each probe produces an output the choice of ID number and the system configuration should be set to ensure that the data structure with that ID travels along the path to be studied.

If successive simulations are run, the new data will be appended at the end of the old data in the text file.

Inputs
OL_In

Outputs
OL_Out

Parameters
File_Path
Probe_Number

Memories
Bypass
Power_Budget_Pointer

Triggered Waveform

Waveform element output after n triggers used for Module testing.

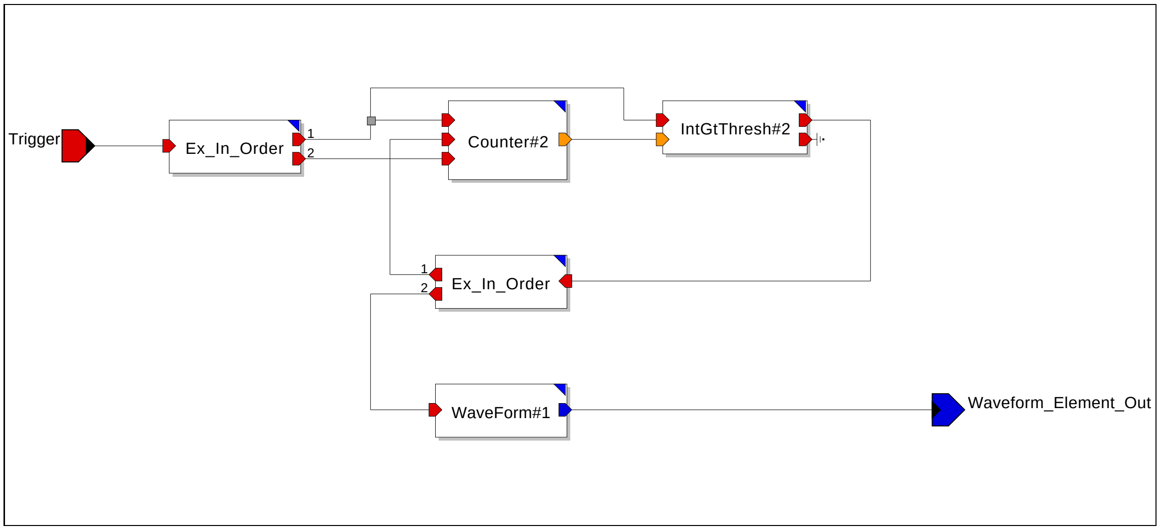

Figure 77: Triggered Waveform Module

Input
Trigger

Output
Waveform_Element_out

Parameters
n_triggers

<u>WDM DS Text 4channel</u>
This module can be used to view the text representation of up to four Single_OL data structure channels within a WDM data structure .

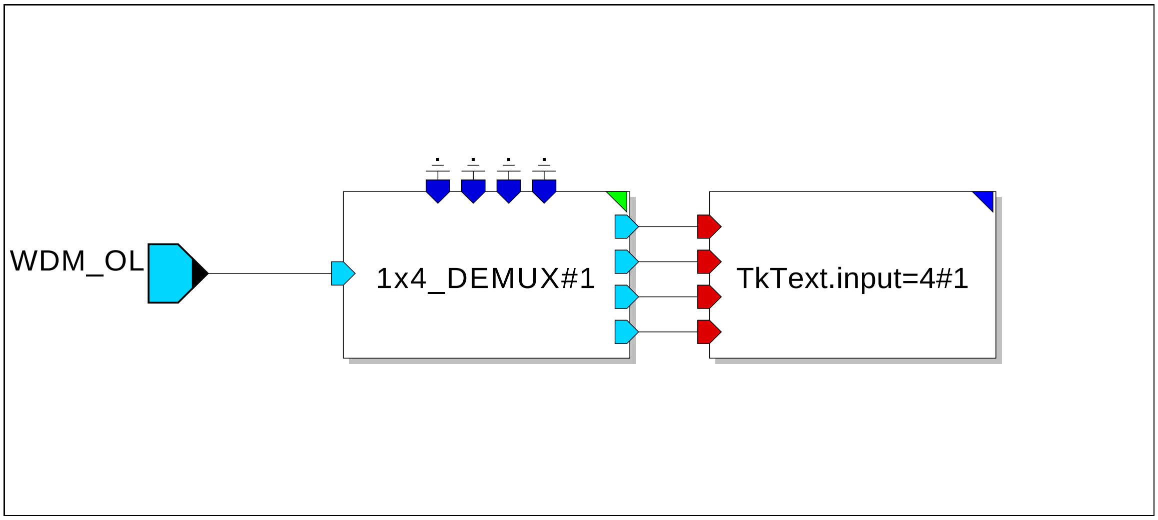

Figure 78: WDM DS Text 4 Channel Module

Inputs
WDM_OL

Outputs
Text for data structures

Parameters
WL_1
WL_2
WL_3
WL_4

5. Sample System Models

Pixel_Bus_Broadcast_and_Select system in DE domain

This system represents a four channel broadcast and select network operating at 2.5Gbps. Each of the four channels is transmitted on a separate DWDM wavelength and then coupled onto a single fiber. The WDM signal is split, and the receivers can be tuned to receive any of the four channels.

The data input into the system is a stream of integers indicating the channel. For instance, Transmitter 1 (channel 1) is given a stream of integer 1's as data, Transmitter 2 is given a stream if 2's, and so on. This is for demonstration purposes to make channel switching evident on the output graphs. In future the integer data will be replaced with more realistic data streams.

The Control Unit determines which wavelength will be received by each optical receiver. It represents a centralized electrical control of the optical network. The Control Unit will read from a file named LION_ver_1_0_r01/wdm_broadcast_control_file which was imported along with the library. A new instruction will be sent to the system every 5 clocks. Please see the Control_Unit online documentation for more information on its use.

The Power Probe modules will read the optical power level of the Single_OL data structure with ID#2 as it travels through the system. Its optical power values will be written into a text file called LION_ver_1_0_r01/wdm_broadcast_power_budget which was imported along with the library. After a simulation this text file can be imported into a spreadsheet program to produce a graphical representation of the optical power loss through a link. Please see the Optical_Power_Probe online documentation for more information.

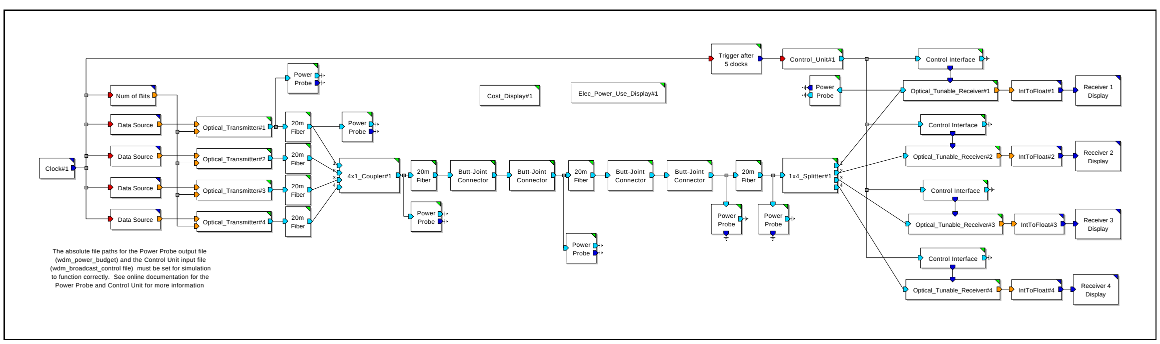

Figure 79: Pixel Bus Broadcast and Selection System Model

System Outputs

Receiver Information:

Received Data (graph)
Latency for each Single_OL received (text)
Average Latency (text)

Power Margin for each Single_OL received (graph)
Average Power Margin (text)

System Information:

Electrical Power Use (text)
System Cost (text)
Text File with Power Probe Values (file)

Note: The Control Unit and Power Probe modules require the full pathnames of the LION_ver_1_0_r01/wdm_broadcast_control_file and LION_ver_1_0_r01/wdm_broadcast_power_budget respectively. A simulation error will be produced if the pathnames are not correct.

Pixel_Bus_WDM_Switched system in DE domain

This system is similar to the WDM Broadcast and Select Pixel Bus. In place of the tunable optical receivers are fixed wavelength receivers. The channels are selected by the bank of demultiplexers and optical switches. This setup while requiring more equipment may be more economical due to the high cost of tunable optical receivers. The system operates at 2.5Gbps.

The data input into the system is a stream of integers indicating the channel. For instance, Transmitter 1 (channel 1) is given a stream of integer 1's as data, Transmitter 2 is given a stream if 2's, and so on. This is for demonstration purposes to make channel switching evident on the output graphs. In future the integer data will be replaced with more realistic data streams.

The Control Unit determines which channel will be received by each optical receiver by controlling to routing characteristics of the optical switches. It represents a centralized electrical control of the optical network. The Control Unit will read from a file named LION_ver_1_0_r01/single_wl_control_file which was imported along with the library. A new instruction will be sent to the system every 5 clocks. Please see the Control_Unit online documentation for more information on its use.

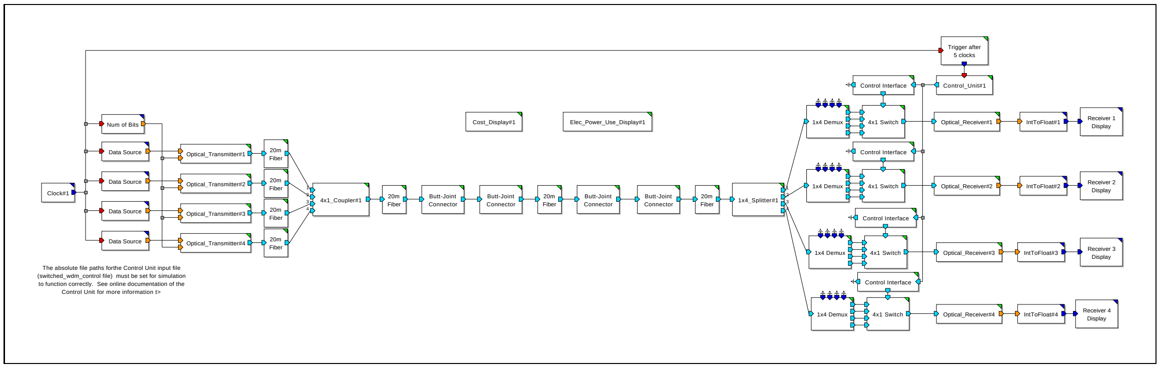

Figure 80: Pixel Bus WDM Switched System Model

System Outputs

Receiver Information:

Received Data (graph)
Latency for each Single_OL received (text)
Average Latency (text)
Power Margin for each Single_OL received (graph)
Average Power Margin (text)

System Information:

Electrical Power Use (text)
System Cost (text)

Note: The Control Unit module requires the full pathname of the LION_ver_1_0_r01/single_wl_control_file. A simulation error will be produced if the pathname is not correct. To correct the pathname simply highlight the Control unit block and alter the File_Path parameter with the location of the LION library in your system.

Pixel Bus Switched Network

In this system, a four channel pixel bus is created using a separate fiber for each channel (no WDM). This is done to explore the possibility of using lower cost multimode components. The signaling rate of the system is limited to 1.0 Gbps. A bank of splitters and optical switches are used to provide each receiver access to all of the channels. The data input into the system is a stream of integers indicating the channel. For instance, Transmitter 1 (channel 1) is given a stream of integer 1's as data, Transmitter 2 is given a strewn if 2's, and so on. This is for demonstration purposes to make channel switching evident on the output graphs. In future the integer data will be replaced with more realistic data streams. The Control Unit determines which channel will be received by each optical receiver by controlling the routing characteristics of the optical switches. It represents a centralized electrical control of the optical network. The Control Unit will read from a file named LION_ver_1_0_r01/single_wl_control_file which was imported along with the library. A new instruction will be sent to the system every 5 clocks.

Please see the Control_Unit online documentation for more information on its use. System Outputs: Receiver Information: Received Data (graph) Latency for each Single_OL received (text) Average Latency (text) Power Margin for each Single_OL received (graph) Average Power Margin (text) System Information: Electrical Power Use (text) System Cost (text)

Note: The Control Unit module requires the full pathname of the LION_ver_1_0_r01/single_wl_control_file. A simulation error will be produced if the pathname is not correct.

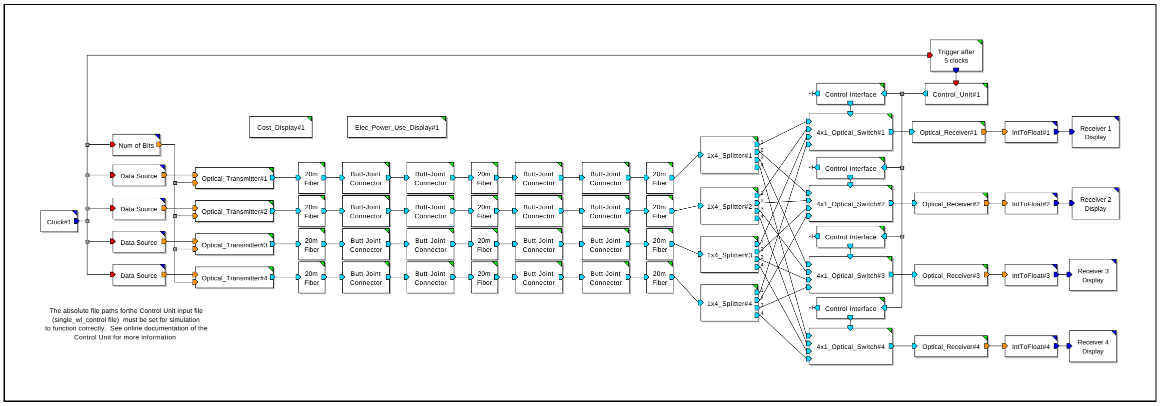

Figure 81: Pixel Bus Switched System Model

The Development Team

This library was developed by members of the OPN Group at the High Performance Computing and Simu;lation Lab (HCS) at the University of Florida, Gainesville. Members of the team included Ian Troxel, group leader, Vikas Aggarwal, Ramesh Balasubramanian, Chris Catoe, Nang Dilakanont, Mythili Muruganathan, John Wemicke and Jeremy Wills.

Dr. Alan George was principal investigator for the project.