D4.3 "Integration of controlled OBST Metro Mesh interconnected with OPST Metro Ring network"

Status and Version:	Issue 1.1	
Date of issue:	10.06.2012	
Distribution:	Public	
Author(s):	Name	Partner
Basham	Mark	INT
O'Connor	Peter	INT
Fletcher	Nigel	INT
Santhosh	Kumar	INT
Zervas	Georgios	UESSEX
Yan	Yan	UESSEX
Rahimzadeh Rofoee	Bijan	UESSEX

Table of Contents

1 Executive Summary 4

2 Introduction 5

3 TSON and OPST (Beta) networks 7

3.1 OPST (Beta) network 7
3.1.1 OPST (Beta) Topology 7
3.1.2 System Overview 8
3.1.3 Introduction & Terminology 8

3.2 TSON network test bed 10
3.2.1 Introduction 10
3.2.2 TSON Network System/Components/Operation: 10
3.2.3 TSON Test bed Topology 11
3.2.4 Optical layer (Layer 1) of TSON test bed 13
3.2.5 Electronic layer (Layer 2) of TSON test bed 14
3.2.5.1 Global Frame synchronization on multiple FPGA-based TSON metro nodes 14

3.2.6 TSON network control 16
3.2.6.1 Server to FPGA interface 16

3.2.6.2 FPGA to PLZT interface 18

3.3 TSON-Beta Integrated test bed 19
3.3.1 System overview: 19

4 TSON/ OPST (Beta) Test Cases and results: Multi-Technology point-to-point sub-wavelength transport service 21

4.1 Overview & Terminology 21

4.2 Tests conditions and characteristics: 21
4.2.1 Scenario representation: 21
4.2.1 Bitrate selection: 22
4.2.1 TSON time slice allocation patterns: 22

4.3 Test Approach 23

5 Point-to-point sub-wavelength services 24

5.1 Case 1: Single Unidirectional Flow 24
5.1.1 Latency result 25
5.1.2 Jitter Result 25

5.2 Case 2: Single Unidirectional Flow 27
5.2.1 Latency Result 28
5.2.2 Jitter Result 28

5.3 Case 3: Single bidirectional Flow 29
5.3.1 Latency Result 31
5.3.2 Jitter Result 32

6 TSON/ OPST (Beta) Test Cases and analysis: Multi-Technology multipoint-to-point sub-wavelength transport service 34

6.1 Case 4: Flow aggregation Unidirectional 34
6.1.1 Latency Result 35
6.1.2 Jitter Result 36

6.2 Case 5: Flow aggregation Unidirectional 37
6.2.1 Latency Result 39
6.2.2 Jitter Result 40

7 TSON/ OPST (Beta) Test Cases and analysis: Multi-Technology multipoint-to-multipoint sub-wavelength transport service 42

7.1 Case 6: Flow aggregation/segregation Unidirectional 42
7.1.1 Latency Result 43
7.1.2 Jitter Result 44

7.2 Case 7: Flow aggregation/segregation Unidirectional 46
7.2.1 Latency Result 47
7.2.2 Jitter Result 48

7.3 Case 8: Flow aggregation and flow segregation partial bidirectional 50
7.3.1 Latency Result 51
7.3.2 Jitter Result 51

7.4 Case 9: Flow aggregation and flow segregation bidirectional 52
7.4.1 Latency Result 53
7.4.2 Jitter Result 54

8 TSON/ OPST (Beta) Test Cases and analysis: Multi-Path interconnected technology domains and transport services 55

8.1 Case 10: Flow aggregation and flow segregation partial bidirectional 55
8.1.1 Latency Result 56
8.1.2 Jitter Result 57

8.2 Case 11: Flow aggregation and flow segregation partial bidirectional 58
8.2.1 Latency Result 60
8.2.2 Jitter Result 61

9 Observations, Analysis and Conclusions 63

10 References 65

11 Acronyms 66

1 Executive Summary

This report covers the test activities related to the MAINS integration of controlled OBST Metro interconnected with the OPST Metro Ring network. The integration and test activities were carried out during May 2012 at the Essex University campus laboratory, located in the United Kingdom.

Only data plane tests were carried out during this first phase of integration test activities. Control plan tests and further data plane tests will be undertaken and reported in Task 4.4 and Deliverable 4.4 respectively.

Four groups of tests were identified that could be achieved within the limitations of the available equipment, in terms of numbers ports and test equipment, as follows,

- End-to-end point-to-point sub-wavelength services - Section 5
- Multipoint-to-point sub-wavelength services - Section 6
- Multipoint-to-Multipoint sub-wavelength services - Section 7
- Multi-Path interconnected technology domains and transport services - Section 8

The TSON network domain consisted of 3 edge nodes and 1 bypass node configured in a star topology. The OPST (Beta) network domain consisted of 3 nodes configured in a ring topology. The two sub wavelength domains were interconnected utilising 10GE ports.

The integration tests validated that the two disparate sub wavelength network domains, TSON and OPST (Beta) can interoperate and pass simulated traffic flows successfully.

As expected the latency and jitter characteristics recorded were in line with each sub wavelength systems technical implementation.

The OPST (Beta) and TSON systems differ in that OPST (Beta) utilises a wavelength routed solution where the receiver is allocated a fixed wavelength, and the transmitter tunes to the receivers wavelength allocation in order to burst packets to the receive destination. Latency is dependent on traffic queue boundaries as traffic is processed ready for burst transmission, and the fibre distances between nodes. The fibre distance provides a constant period of latency whilst the traffic queue latency is dependent on buffer depth and system loading. This aspect of the system was not stressed during the integration tests. TSON operates a different approach to sub wavelength utilisation where time sliced allocation of the wavelength has been implemented. Although the latency and jitter recorded for TSON was higher values than for OPST (Beta) the values remain constant regardless of loading.

2 Introduction

This report covers the test activities related to the interconnection of the TSON Metro network with the OPST (Beta) Metro Ring network.

During May 2012 a team of Essex University and Intune Networks engineers carried out a range of introductory data plane interworking tests utilising the test bed facility housed at the Essex University campus laboratory. The test bed consisted of the Essex University provided TSON sub wavelength system and the Intune Networks OPST (Beta) sub wavelength systems. Both these systems are described in more detail in Section 3 of this document.

The test activities were restricted to the data plane interworking. The control plane interworking utilising the GMPLS controller to execute an end to end path across the TSON and OPST (Beta) network domains will be carried out in T4.4 and reported on in D4.4.

During the first phase of testing (first time the TSON and OPST (Beta) domains have been interconnected) the tests have been focussed on assessing system characteristics when passing test traffic through two OBS domains. In order to provide a set of baseline test data to enable an assessment for the interconnected data path interworking of the two sub wavelength systems to be made the TSON and OPST (Beta) domains were tested independently using the same test case scenarios.

This report includes a description of each test performed together with a corresponding set of results, covering through put, latency and jitter.

A total of 30 test cases were originally identified which covered tests to be conducted on the TSON and OPST (Beta) network domains separately to provide the base line test data, and a further set of tests to be conducted for the TSON plus OPST (Beta) interconnected network. However, a number of the test cases required more equipment in terms of sub wavelength system ports and test equipment than was available. In addition the test cases were consolidated to 11 primary tests cases, where within each tests were operated for TSON, OPST (Beta) and TSON interconnected to OPST (Beta). The 11 primary tests case are summarized below.

Case 1: Single Unidirectional Flow (Forward)

Case 2: Single Unidirectional Flow (Reverse)

Case 3: Single bidirectional Flow

Case 4: Flow aggregation Unidirectional (Forward)

Case 5: Flow aggregation Unidirectional (Reverse)

Case 6: Flow aggregation/segregation Unidirectional (Forward)

Case 7: Flow aggregation/segregation Unidirectional Reverse)

Case 8: Flow aggregation and flow segregation partial bidirectional

Case 9: Flow aggregation and flow segregation bidirectional

Case 10: Flow aggregation and flow segregation partial bidirectional

Case 11: Flow aggregation and flow segregation partial bidirectional

All tests utilise the client Ethernet interface that the TSON and OPST (beta) systems are deployed with. The Ethernet interface (10GE) provides both the interface point of access for the test equipment/traffic simulators and the interconnection interface between the TSON and OPST (beta) network domains. For each test case a range of Ethernet transmission and frame rates were applied, and receive performance results in terms transmission throughput, latency and jitter recorded.

3 TSON and OPST (Beta) networks

3.1 OPST (Beta) network

3.1.1 OPST (Beta) Topology

The OPST (Beta) topology for the ring to TSON integration tests consists of three OPST beta nodes configured in a ring architecture.

The 3 nodes are configured (positioned) to provide 2 traffic interconnect client interfaces and 1 TSON interconnect client interface, or 1 traffic interconnect client interface and 2 TSON interconnect client interfaces depending on the test case scenario, as illustrated below.

Figure 3-1: OPST Beta node configured for 2 test traffic ports and 1 TSON interconnect port

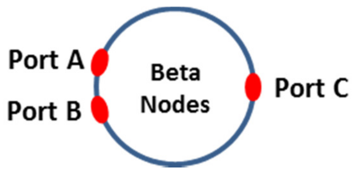


Figure 3-2: Logical representation of OPST (Beta node) configured for 2 test traffic ports and 1 TSON interconnect port

(Beta) node configured for 1 test traffic ports and 2 TSON interconnect port

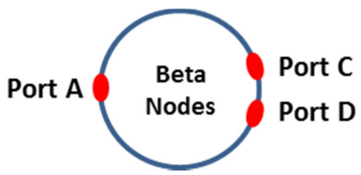


Figure 3-3: Logical representation OPST (Beta) node configured for 1 test traffic ports and 2 TSON interconnect port

3.1.2 System Overview

The OPST (Beta) platform contains all of the key elements of OPST technology, packaged in a compact prototype equipment format, for deployment in lab test facilities.

All of this innovative OPST technology is re-used for the commercial grade platform, iNX8000, with alternative packaging to provide the levels of resilience and operational readiness that a network operator would require for commercial deployment.

The implementation of the OPST Beta platform contains and demonstrates that the fundamental technical capabilities of OPST can be tested and proven.

Figure 3- 4: OPST Beta node

3.1.3 Introduction & Terminology

An OPST (Beta) Ring is comprised of optical fiber that supports the flow of traffic between client-facing interfaces that are connected by means of Intune Networks capability and innovation in burst mode optical transmission and switching.

The terminology of an OPST Node is given to any physical chassis which provides physical connectivity to the OPST Ring.

Each OPST Node is the host platform chassis that locates one or more client-facing interfaces which can access the ring. These interfaces are termed OPST Port - to describe the optical destination for this traffic across the ring.

Each OPST Port can take the form of single or multiple client-facing physical interfaces. The service and protocol agnostic nature of OPST supports client-facing interfaces that may be either asynchronous [Ethernet] or synchronous [SDH].

The goal of the prototype Beta platform is to provide a physical implementation to demonstrate key aspects of Intune Network's innovation and application of OPST. The prototype can be used to test the technology within a number of stated restrictions, which are fully identified in this document.

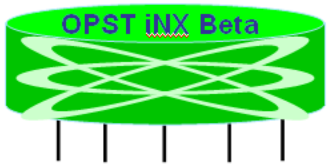
The OPST Beta platform is implemented with both common equipment components, that provide connectivity to the ring, and client-facing components on the same compact equipment chassis. Therefore by contrast to the commercial implementation, the Beta platform can be considered as <u>both</u> an OPST Node and an OPST Port due to the packaging of the prototype.

The commercial grade iNX8000 platform (a traditional multi-slot network equipment chassis), utilises a large proportion of OPST technology from the Beta prototype platform.

However, the physical implementation of the Beta prototype differs in how connectivity to the OPST Ring is provided - versus the ‘carrier-grade’ implementation of OPST on the commercial iVX8000 platform.

An OPST Ring can be considered to be an Ethernet switch with ‘N’ OPST ports that are distributed across multiple metro equipment locations, using optical fiber to create the fabric of the switch.

The beta implementation can logically be positioned as a five port Ethernet switch as illustrated on the right.



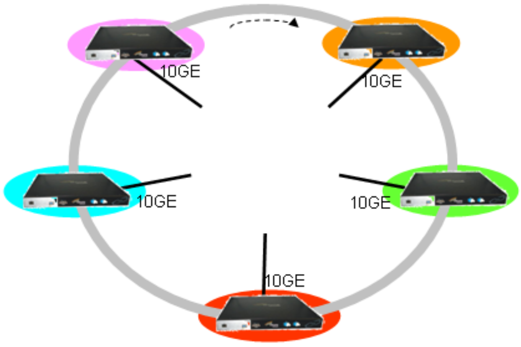
The Ethernet ports are physically distributed and exchange traffic with each other over the OPST fabric.

Each Beta platform hosts one 10Gbps Ethernet (10GE) client-facing traffic interface and a means to connect up to five chassis as an OPST fabric.

In this way, the Beta chassis operates as an OPST Node that houses a single OPST Port.

The figure on the right depicts how five Beta Nodes can be deployed over a circumference of 60km optical fiber in a physical ring topology.

The coloured background portrayed with each chassis, is used to help visualise that one optical wavelength is uniquely associated with each OPST Port on the system and is therefore used to determine the optical destination for all traffic destined for that Port.



OPST Physical Ring

OPST technology allows a full logical mesh of N:N connectivity between all OPST Ports using the optical burst mode transmission and distributed traffic scheduling functionality.

The figure on the right depicts the fully meshed logical flow topology that exists for an Beta ring comprised of five OPST ports.

For more details on the fundamental principles of OPST technology, the reader is referred to Reference 1.

OPST Logical Mesh

Each OPST port on the system makes use of a burst-mode optical laser, capable of rapid tuning across all wavelengths in the ITU C/L Band. This provides the capability to send traffic to multiple destinations and efficiently achieve a full logical mesh of connectivity between all of the OPST Ports.

The maximum speed at which each OPST Port can communicate to or from the ring fabric is derived from the 10Gbps line-rate capability of the client-interface and a capability termed as the OPST Scheduler. This device provides transmission overhead for the optical bursts and ensures that all OPST Ports have equal and fair opportunities to transmit onto the fiber medium.

Additionally, a mechanism to guarantee the bandwidth for high priority flows is provided by the Scheduler.

The mechanism by which each of the tuneable lasers in each OPST Port determines the correct optical destination for Ethernet traffic, as it arrives on the client interface, is the layer two MAC address.

Each OPST Port, implemented in the Beta platform, operates as a layer two traffic forwarding device. The capability to configure a simple traffic lookup table is provided on the OPST Beta platform enabling, once configured by software, the OPST hardware delivering the correct layer two optical packet forwarding. A partial example of this is illustrated for one OPST Port in the table below.

3.2 TSON network test bed

3.2.1 Introduction

The Time Shared Optical Network test bed is designed and implemented using TSON nodes described thoroughly in Figure 3-.The network implementation uses interconnected TSON nodes to provide a dynamic all-optical fast switching network achieving high optical resource efficiency by time-multiplexed connections. As shown in the **Figure 3-**, the network test bed communicates with the external networks/clients through TSON edge nodes using 10GE links, while TSON bypass node switches and transport data in the core of the network using the calculated and established time-shared light paths. The implementation of the TSON node, as per deliverable [1], includes the Layer 2 and bypass functionalities, which have been realised by using FPGAs for O-E-O processes, and optical components such as PLZT switches for time-slice data transport and switching. Therefore the nodes based on the network formation and topology can operate as both or either roles of Edge and Bypass node.

3.2.2 TSON Network System/Components/Operation:

The TSON network system is shown in a logical view in **Figure 3-6** (Left). The Anritsu traffic generator has been used as the client with 10GE ports. TSON edge nodes receive the time slot allocation information from the management plane (PC), and use them to transmit the Ethernet traffic they receive from the Client ports inside the TSON cloud all optically. The TSON Bypass node in the middle is controlled to switch on the calculated time slices. The control information for the TSON bypass node is sent from the management plane of the node to the controlling module in the Electronics layer of the node i.e. FPGA board, and then the FPGA send parallel commands using 24 pin ribbon cables to the PLZTs.

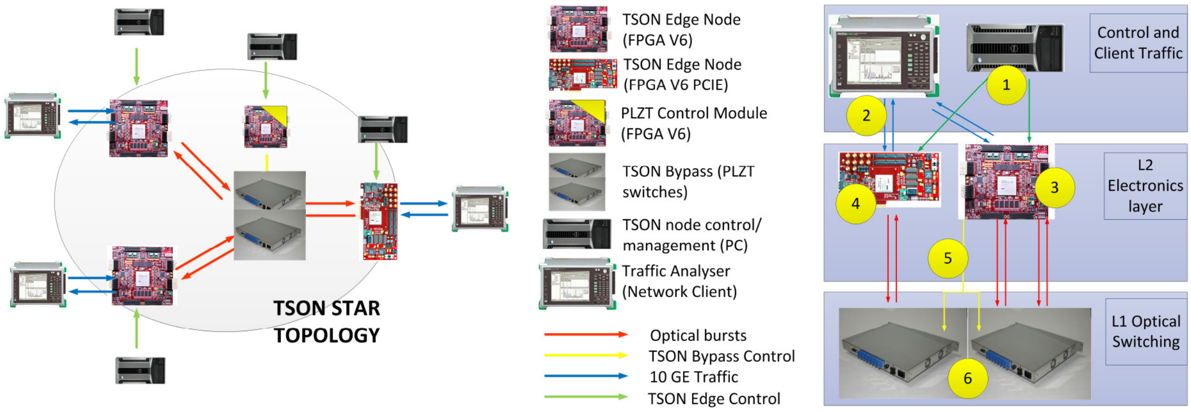


Figure 3-5: (Left) TSON network logical view. (Right): TSON network implementation and flow structure

The TSON network implementation is carried out using the following components:

- A Server PC using a web services-based management plane for controlling the node and network elements.
- Two high performance Virtex 6 FPGA boards, hosting three TSON Edge and one TSON bypass nodes.
- Two 2x2 PLZT fast switches for traffic bypass.
- An Anritsu Traffic Analyzer with 10GE ports as the client ports.

The work flow of the system is illustrated in **Figure 3-5** (Right) using the numbered labels.

1- At first, the management layer which is deployed on a PC, sends the allocated transmission time slices and PLZT switch control information to the FPGAs through 10GE control links. One of the FPGAs on a PCIE boards hosts one TSON node, and the other FPGA hosts two TSON edge nodes and one TSON bypass node since it has more optical transceivers mounted.

2- The Anritsu traffic analyser as the client is set on desired bit rates and traffic pattern to send Ethernet Traffic to the TSON nodes.

3- The bigger FPGA board hosting two TSON edge and one TSON bypass node receives the allocation information encapsulated in Ethernet packets from the management PC, and populates its look up tables designed for each node. Then it will start using the allocated time slices for data transmission on the edge nodes, and controlling the PLZT switches from the bypass node.

4- The PCIE board hosting one TSON edge node starts data transmission right after receiving the allocation table from the management plane.

5- The control information for the PLZT switches are sent from the FPGA based bypass node, so to switch and transport the light paths.

6- The PLZT switches are controlled and used in the bypass node.

3.2.3 TSON Test bed Topology

The network topology we have deployed at the current state of the TSON data plane is a star topology, supporting one wavelength (1544nm) and bi-directional switching as displayed in 7. This topology provides all to all-connectivity, enabling us to interleave data used mixed time slices for setting up the sub-wavelength light paths. Therefore we are able to demonstrate different scenarios of data aggregation, segregation, and concurrent data light paths and more. Some sample cases are illustrated in the **Figure 3-7**, and for the results extensive test cases have been trialled for pure TSON evaluation as well as TSON-Beta integration as demonstrated in later chapters.

The reason we have adopted a star topology at this stage of the implementation, was due to some critical technical issues we experienced using the PLZT switches. The PLZT fast switches were operational until very recent stages of the experiments apart from some poor performances, however it seems the chipsets inside a few of them was showing misconducts so we had to remove them from out test bed and shrink the test bed size consequently. However we have been in contact with the vendor and we are looking forward to receive replacement boards in a few weeks time.

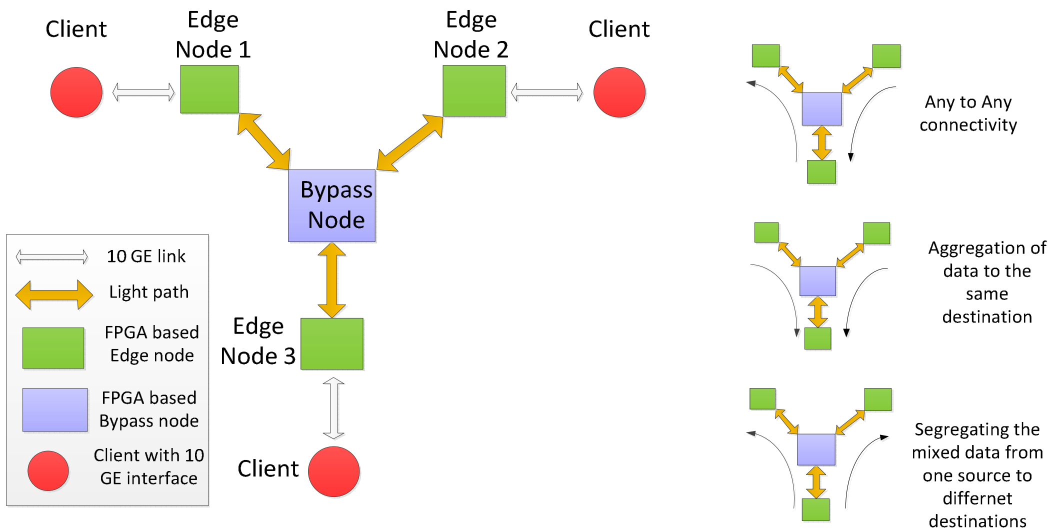


Figure 3-6: LEFT: Network topology of star, in which each of the edge nodes are able to connect to external network/clients. RIGHT: sample cases for different network functionalities.

3.2.4 Optical layer (Layer 1) of TSON test bed

The optical layer of the TSON network is built using a number of active and passive network elements. The active components include PLZ fast optical switches, TSON FPGA L2 implementations[1], EDFAs, while as passive components we have couplers and filters and so on. All these elements have been mounted on a 3D MEMS switch as an optical backplane, and are incorporated into the topology upon the need. **Figure 1-7** displays this configuration. In [2], we have presented the capabilities of such a configuration in providing flexibility and adaptiveness in dataplane.

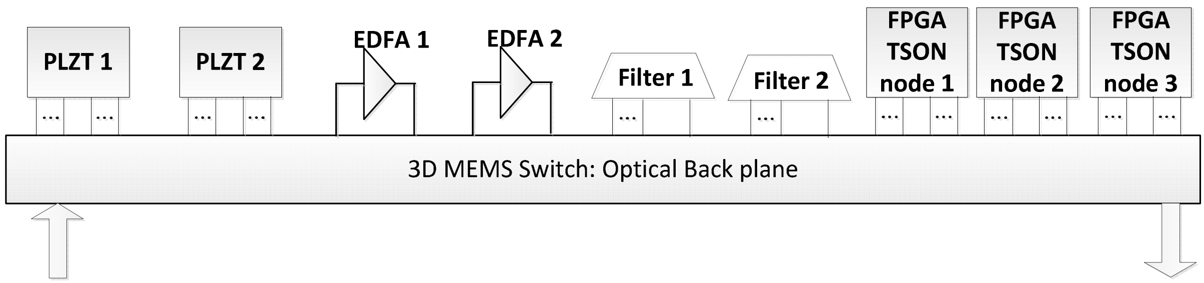


Figure 2-7: The backplane of network elements for realising dynamic topology formation.

The network formation which uses backplane configuration is shown in Figure 3-8. The cross-connections made on the 3D MEMS to include the desired components in the topology are displayed using dark star shapes on the links in Figure 3-8.

PLZT fast switches are the main optical components which enable fast time switching of the traffic time-slice data sets in the data plane. These switches are built to operate with switching speed of 10 ns. In order to make a star topology with three edge nodes a 3 input /3 output central bypass switching node is required. In this regard we have used two 2x2 PLZT switches in the central node, each of the switches to switch on one direction of the flows. The interconnection between the network elements is through LC-APC fiber links.

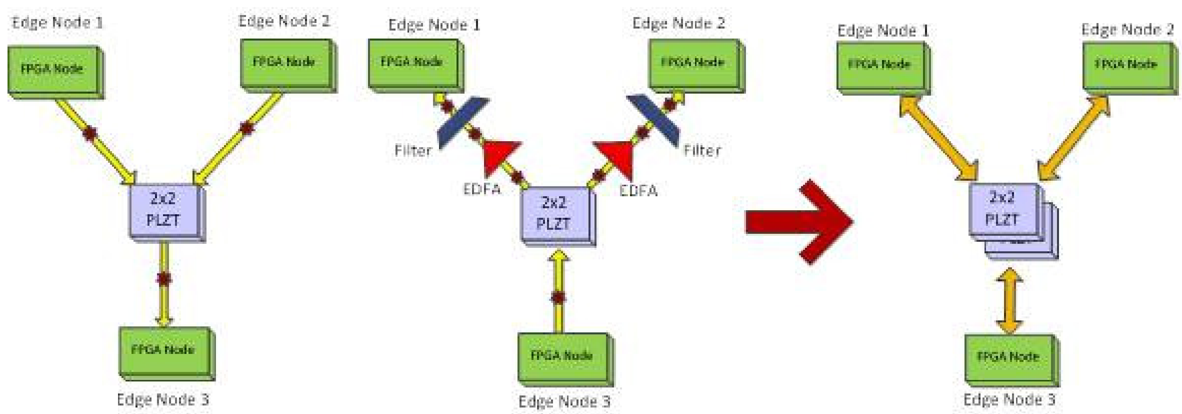


Figure 3-8: Bidirectional star topology is realised by integration of two unidirectional sub-topologies.

As displayed in Figure 3-8, the logical bidirectional star topology is formed by combining the unidirectional architectures. It can be seen that apart from FPGA nodes and PLZT switches, some other components such as filters (filtering channel of 1544 nm) and EDFAs have been used on some of the links as well to compensate for high and un-equal insertion loss of the PLZT fast switches. This complexity stems from the fact that the fast switches operate with un-equal levels of insertion loss cross talk and polarization-dependent loss (PDL) among different cross-connections. To add to the complexity, the TSON ingress/egress nodes developed using FPGA platform, show different sensitivity levels for signals as boards from different vendors exhibit different characteristics.

3.2.5 Electronic layer (Layer 2) of TSON test bed

In order to implement the layer 2 functionalities in the network, high performance Virtex-6 FPGAs are used, which have been described in [1][1]. To set up 3 TSON nodes for the network, 2 FPGA boards are deployed as the platform shown in **Figure 3-9** Besides the data processing functionalities of the edge nodes, Switch control functions have also been implemented in one of the FPGAs, as a separate module for the bypass node, to allow the optical switching.

Figure 3-9: FPGA boards hosting the TSON nodes

3.2.5.1 Global Frame synchronization on multiple FPGA-based TSON metro nodes

On the aspect of TSON networking, all the participating FPGA-based TSON nodes need to get synchronized, because though they work at the range of 156.25MHz, but there is always a slight difference (few Hz) between the clocks of the nodes in phase and frequency. Therefore it is important to make sure all nodes are frame aligned/synchronized so that the output time-slices would be switched at the bypass node without losing information (not be chopped by the PLZT switches). In this regard, we need to synchronize all the nodes and set time 0 as the start point for all the nodes, and then iterate every frame, which is 1ms, shown in **Figure 3-10**.

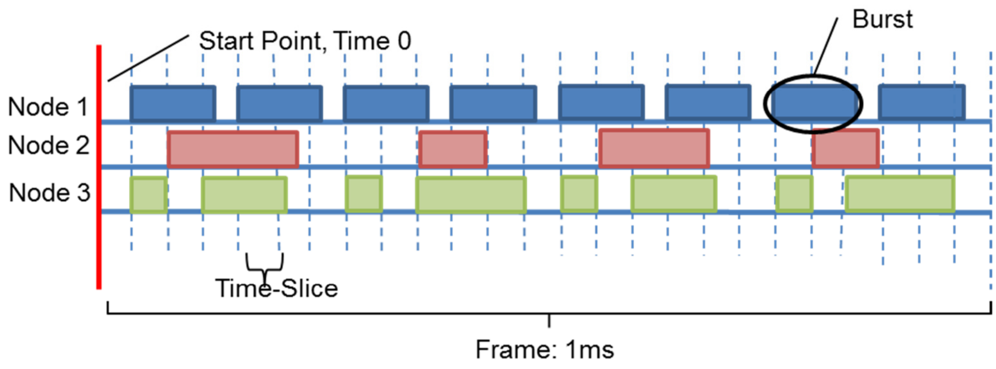


Figure 3- 10: Synchronization timeline for 3 nodes.

The overall mechanism for global node frame synchronization is based on an initial three way handshake between a chosen master clock with the slave nodes (clocks). The slave clocks will learn about the signal delay travelling from mater to them, and also align their frame with the mater node. Afterwards, the master clock will keep sending the "beginning of the frame" signals to the slave clocks for the realignment purposes and to evade a cumulative time drifts **Figure 3-11.**

To implement the synchronization among three nodes, for the hardware parts, the nodes are connected each other two custom-defined pins on the boards, one TX and one RX. As shown in **Figure 3-11** node A is set as the master clock node, and node B and C send a “FLAG” signal to A. After receiving the “FLAG” signal, node A sends back an “ACK” signal to node B and C, which can calculate the transport delay from A. After the calculation, node B and C send a “calculation done signal” to node A that completes the initialization stage. After A receives this, for every frame, node A periodically (for every frame) sends a time 0 “ send synch” signal to B and C. Node B and C are able to set their time based on the this signal and the calculated delay. The delay from node A to node B is 1 clock cycle (6.4ns), and from node A to node C is 12 clock cycles. The difference is because node A and node B are on the same FPGA board, node C is implemented in a separate FGPA board.

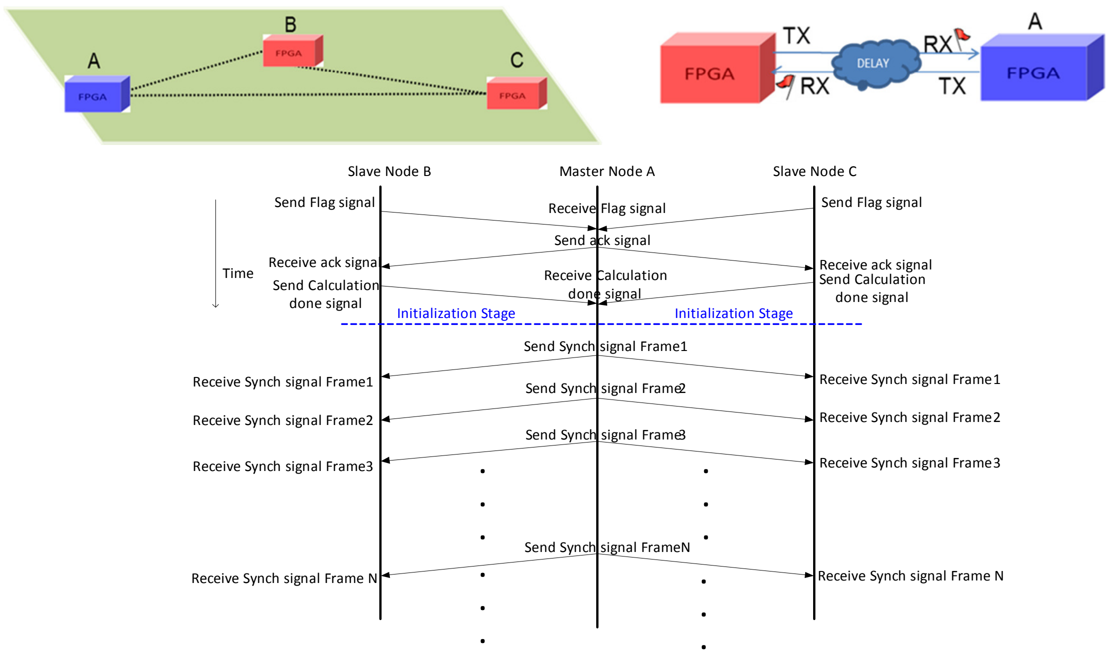


Figure 3-11: Nodes synchronization protocol

3.2.6 TSON network control

In order to communicate with the TSON nodes in the data plane, we have developed web-services based tool which sends time slice allocation information to the TSON nodes using the XML based information structure developed for GMPLS operation. To this purpose, the web-services tool is placed on a server as the management plane of the network. The management plane is connected to the TSON nodes using 10 GE links, and sends the time slice allocation information to each of the FPGAs for the installed nodes.

Apart from TSON edge nodes, network data plane is inclusive of a TSON bypass node with PLZT switches as described earlier. In order to set up paths and establish connection we use the developed web-services mechanism to send the control information to the bypass module on one of the FPGAs, and then that module directly controls the switches. Therefore the paths calculated using the management tool are translated to switch control bit map table and sent to the corresponding FPGA for making the switching paths for data transfer.

3.2.6.1 Server to FPGA interface

The communication between the management server and the FPGAs takes place using Ethernet raw sockets which are implemented using java library of JPCAP. The Ethernet frames are filled with the information for the network operation and are sent to the FPGAs to populate their look up tables.

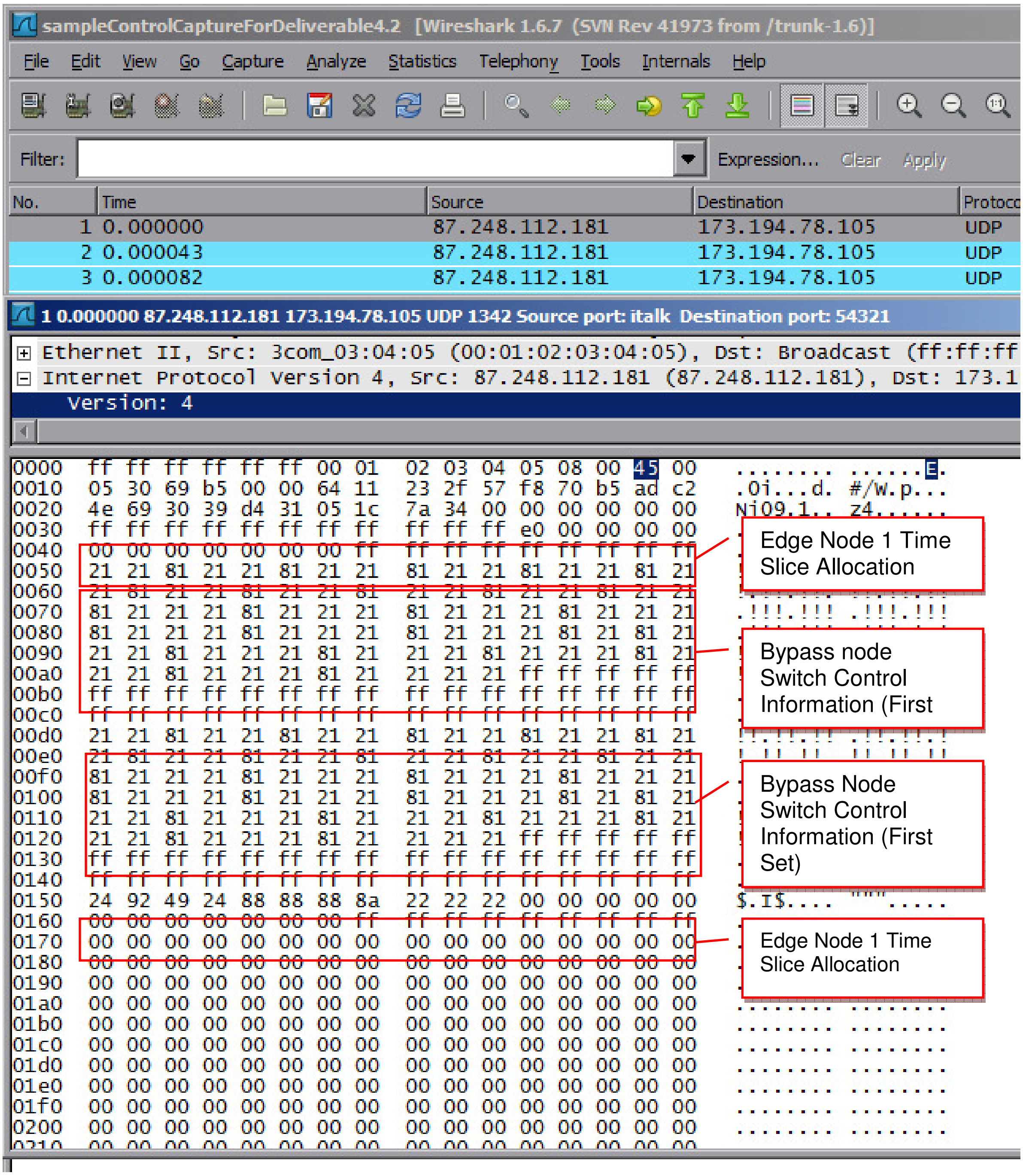


Figure 3-12: Wireshark capture: Ethernet packet structure, carrying control information from server to FPGA

As it is shown in 3-12, the Ethernet frame payload is set based on the time-slice allocation and switch control information. The first box shows the time slices allocation for the first node, which is represented in the hex format of the bit map allocation matrix. The two bigger boxes in the middle carry the switching information for the switching control. In our test bed in total we have two 4x4 PLZT switches, and four 2x2 PLZT switches. For the ease of control and practicality, we have grouped them in two sets of one 4x4 plus two 2x2

PLZT switches, each of the sets are commanded separately. The number such as 21 and 81 carry switching combination state information based on internal conventions.

A brief explanation on what the numbers such as 21 and 81 shown in Figure 3-12 represent: these numbers carry information for 8 bit data types designated for each set of switches, in hex format. So the numbers in decimal system are 33 and 129 respectively. The first 5 bit of every number is used for the 4x4 switch control, which has more combinations of ports. The last 3 bits are reserved for the control information of the two 2x2 switches. Therefore, a number of 33, which is translated to binary form of 00100001 with MSB on left, gives the first state of switching for the two 2x2 switches (001) . The other state "one" on the 5 bit part (00001) is used for the 4x4 switch control. The switching states then are translated to pin signal for switch control for each combination of input-output ports.

<table>
<tr><th colspan="33">LUT Address MAP in FPGA</th><th>Ethernet Frame lines (Captured by Wireshark)</th></tr>
<tr><td>ADDR\BYTE</td><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td></td></tr>
<tr><td>00</td><td colspan="32">RESERVED</td><td>10,20</td></tr>
<tr><td>01</td><td colspan="7"></td><td colspan="25">Time-slice Allocation for λ1 and λ2 (Destination MAC address1)</td><td>30,40</td></tr>
<tr><td>02</td><td colspan="32">1 to 32 PLZT switch control commands for λ1</td><td>50,60</td></tr>
<tr><td>03</td><td colspan="32">33 to 64 PLZT switch control commands for λ1</td><td>70,80</td></tr>
<tr><td>04</td><td colspan="32">65 to 96 PLZT switch control commands for λ1</td><td>90,a0</td></tr>
<tr><td>05</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td colspan="4">97 to 100 PLZT switch control commands for λ1</td><td>b0,c0</td></tr>
<tr><td>06</td><td colspan="32">1 to 32 PLZT switch control commands for λ2</td><td>d0,e0</td></tr>
<tr><td>07</td><td colspan="32">33 to 64 PLZT switch control commands for λ2</td><td>f0,100</td></tr>
<tr><td>08</td><td colspan="32">65 to 96 PLZT switch control commands for λ2</td><td>110,120</td></tr>
<tr><td>09</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td colspan="4">97 to 100 PLZT switch control commands for λ2</td><td>130,140</td></tr>
<tr><td>10</td><td colspan="7"></td><td colspan="25">Time-slice Allocation for λ1 and λ2 (Destination MAC address2)</td><td>150,160</td></tr>
<tr><td>11</td><td colspan="7"></td><td colspan="25">Time-slice Allocation for λ1 and λ2 (Destination MAC address3)</td><td>170,180</td></tr>
<tr><td>12</td><td colspan="7"></td><td colspan="25">Time-slice Allocation for λ1 and λ2 (Destination MAC address4)</td><td>190,1a0</td></tr>
</table>

Figure 3-13: LUT Address MAP in FPGA

The information inside the Ethernet frames when received by the FPGA, are used to populate the Time-slice allocation and PLZT switch LUT in the FPGA. The LUT address map in the FPGA is shown in Figure 3-14**Figure 3-**. The FPGA-based TSON metro node can support Ethernet streams with up to 4 destination MAC addresses, so the time-slice allocations are assigned with different destination MAC address, such as address 01,10,11,12, shown in Figure 3-3.

3.2.6.2 FPGA to PLZT interface

The fast switches, as the active components of the TSON Layer 1 data plane which need to be configured for switching at correct times to direct the bursts of traffic, need to get the switching information in a timely and effective manner.

For this purpose, a separate switch control module on FPGA has been developed for passing the switch control commands to the switches in parallel with traffic generation.

Each of the fast PLZT switches (two switches, one per direction) is configurable through 48 pins on two DB25 connectors. For this purpose, the expansion connectors on a daughter board card attached to the FPGA board are connected to the DB25 connectors on the PLZT switch using a custom made ribbon cables.

The desired switch configuration is then sent to the PLZT switch from the FPGA at the

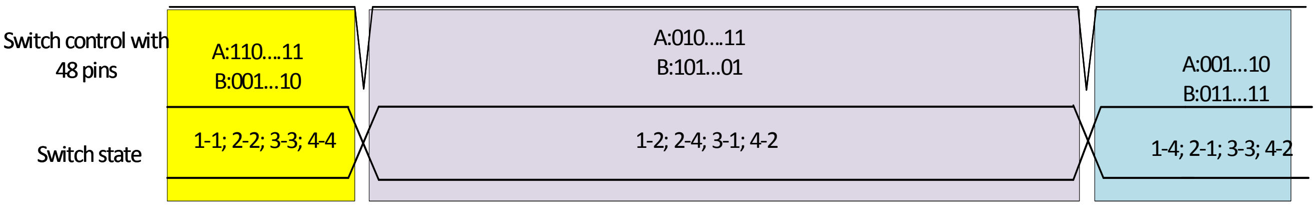


Figure 3-14: PLZT control bits and the correspondence switch state change

specific timings determined by the central controller.

Figure 3-14 illustrates the 48 control bits (split in two parts of A and B due to the PLZT functional structure), and the correspondence change in the state of the switch is displayed. So, as long as the control bits sent are constant, the switch remains at the same state. The control bits are raw zeros and ones (3.3 V peak-to-peak) directly applied to the switch driver.

3.3 TSON-Beta Integrated test bed

3.3.1 System overview:

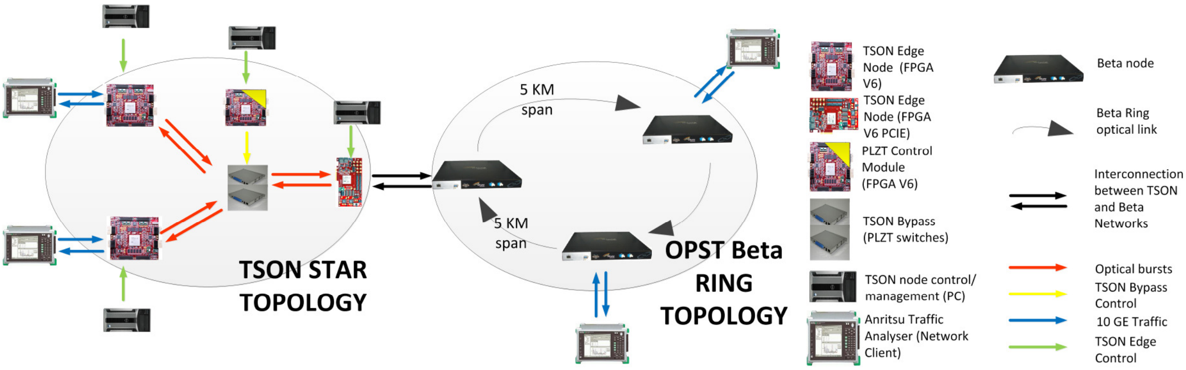


Figure 3-15: TSON and Beta networks integration

The test configuration had a ring of three BETA nodes and a star of three TSON Nodes (**Figure 3-15**). TSON and Beta nodes were connected using 10 GE links of LC connectors. The Ethernet traffic between TSON and Beta networks was transferred transparently as if the networks were connected to any client with 10 GE connectivity.

Beta network setup: The two spans (out of three) of BETA node optical fibres had a span of 5km each. The three node ring of BETA nodes was brought up with one node set as master. The MAC addresses of the BETA nodes were set to match the LAB LAN settings. The Intune OPST Beta Connection Manager and Intune Photonic Manager applications were installed on one of the servers in the LAB. The Photonic Manager was used during installation for setting optical parameters on all the BETA nodes. The Intune OPST

Connection Manager GUI was connected to the BETA node ring to add or delete CoS based virtual connections between BETA node endpoints.

TSON network setup: The TSON network was connected to the OPST Beta network through the TSON node 3 on the FPGA PCIE board (indicated by the yellow label on 5). Therefore client traffic from Anritsu traffic generator was fed into the TSON nodes 1 or 2, and the bypass node and the traffic flow on TSON was towards TSON node 3 to be transferred to the OPST Beta network.

For the traffic generation and reception the two ports from Anritsu Traffic generator (MD1230_MP1590_ET1100) were used. The port configuration for the traffic generator involved input of IPv4 address, Netmask, Gateway and MAC address (of the client port in case of BETA nodes). Transmit, receive of the port from Anritsu was connected to the client port's (of the BETA node) receive, transmit respectively using. The bit rate was also set using the traffic generator.

4 TSON/ OPST (Beta) Test Cases and results: Multi-Technology point-to-point sub-wavelength transport service

4.1 Overview & Terminology

A range of test cases have been defined to provide a first stage evaluation of the data path interoperability between two differing sub wavelength systems, OPST (Beta) and TSON. Our approach is to where possible run similar tests on the OPST (Beta) configuration, TSON configuration and then finally on an OPST (Beta) connected to TSON configuration. This approach allows the team to compare and contrast between configurations, identifying any anomalies that arise.

4.2 Tests conditions and characteristics:

4.2.1 Scenario representation:

The integration of the two test beds is visualised in Figure 4-1, where the ring shape represents the Beta test bed, while the square shape represents the TSON network. It should be noted that using this shapes are solely for logical connectivity presentation purposes and also a means of differentiation between the test beds. These shapes are not supposed to carry any real topological information. For the ease of illustration and presentation the ports on either test bed representational shapes have been labelled with an alphabetical character, as it can be seen in Figure 4-2.

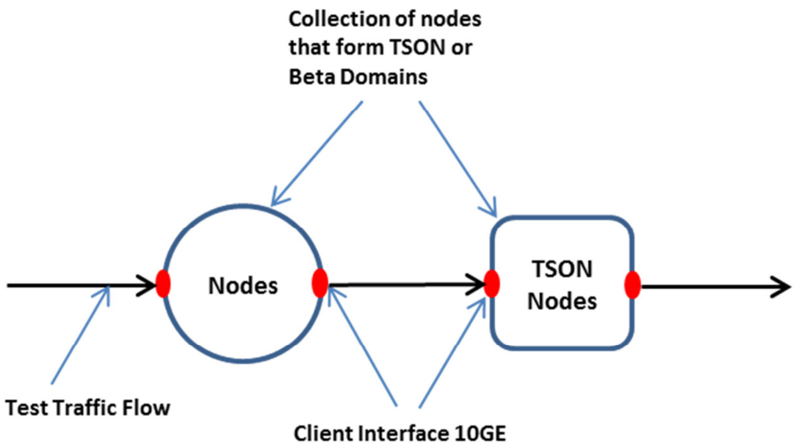


Figure 4-1: Test case logical representation overview

Figure 4-2: Test case logical representation overview with port nomenclature

4.2.1 Bitrate selection:

A number of tests have been conducted for each test case. A range of test traffic loadings have been applied for each test case and the receiving throughput recorded along with the latency and jitter characteristics.

Frame Size (byte)	Transmit rate
64	1G
1500	1G
64	2.2G
1500	2.8G

Table 4-1:

As summarized in **Table 4-1**, a maximum 2.8 Gbps with 1500 bytes, 2.2 Gbps with 64 bytes of frame size, and 1Gbps data rates with 64 and 1500 frame sizes have been applied to test the data plane performance. Also, for TSON network, since various time slice allocation scheme can affect the network performance, a set of patterns have been used over the selected bitrates as well.

Max bitrate 2.8Gbps: In the TSON network environment, actively switching light paths cause the FPGA GTH receivers of the TSON nodes to lose the clock, so they need some time ahead of the of data signal to recover the clock when receiving the bursts. In this regard we have designed the nodes to have "Keep Alive messages" on the network flowing whenever there is no data to send to enable the receivers to maintain clock recovery on active switch network environment, and to be able to get the highest network efficiency by not using too much space of the data time slices by clock recovery signals. However, since the "keep alive messages" are generated from sources with slightly different clocks, and also due to unbalanced powers insertion loss on different light switch cross-connection paths, the received keep alive signals can be different in phase and amplitude from the data coming from other sources. The implemented "keep alive messages" may fall short from their intended purpose. Therefore we have had to include the "keep alive messages" inside the actual data burst time slice dataset as well. In the datasheet of Virtex-6 FPGA board, the maximum clock recovery phase acquisition time is 20us. This is the maximum period it takes to lock to data after PLL has locked to the reference clock. It is highly influenced by the noise of the data lines. Considering that the aim of our integration tests are to showcase the Layer 2 characteristics with regards to QoS, in the experiment, to avoid any data loss we considered, in 1 frame (1ms) with, the number of Time-slices reduced to 30 time-slices, then, each burst time-slice is 33us, with 10us data and 23us K-characters to safeguard the clock recovery. So the maximum link capacity is 30%, the maximum bit rate for Ethernet frame size 1500B is 2.8G, and the maximum bit rate for Ethernet frame size 64B is 2.2G.

4.2.1 TSON time slice allocation patterns:

For TSON or TSON+BETA, four different time-slice allocations are used for TSON to evaluate the impact on latency and jitter results. Different patterns can cause different latency and jitter results in the TSON network because of the aggregation/segregation mechanisms.

Time-slice Allocation1: 010101010101010101010101010101

Time-slice Allocation2: 001100110011001100110011001100

Time-slice Allocation3: 111111000000001111111100000000

Time-slice Allocation4: 111111111111111111111111111111

4.3 Test Approach

The tests defined have been designed to check the basic data path interworking of sub wavelength systems and provide a reference base line set of test data to identify any 'standout' anomalies. At this stage the tests have been designed to test the integrated system general behaviour rather than performance.

Four groups of tests were identified that could be achieved within the limitations of the available equipment in terms of numbers ports and test equipment, as follows,

- Chapter 5 focuses on all end-to-end point-to-point sub-wavelength services
- Chapter 6 on Multipoint-to-point sub-wavelength services
- Chapter 7 on Multipoint-to-Multipoint sub-wavelength services
- Chapter 8 on Multi-Path interconnected technology domains and transport services

5 Point-to-point sub-wavelength services

5.1 Case 1: Single Unidirectional Flow

The single unidirectional flow test case provides a simple entry level test to provide a reference set of results for comparison with the follow on test cases. The characteristics of a unidirectional flow are recorded for a range of transmission values, each for the TSON network domain and the beta network domain. The collective of these 2 sets of results is then compared with the integrated TSON and Beta end to end results.

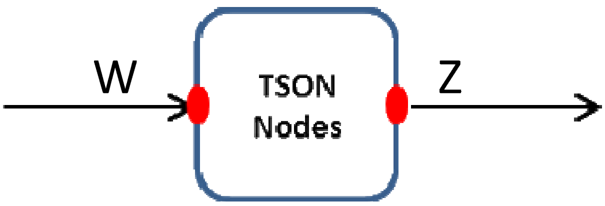


Figure 5-1: TSON Case 1 single unidirectional flow

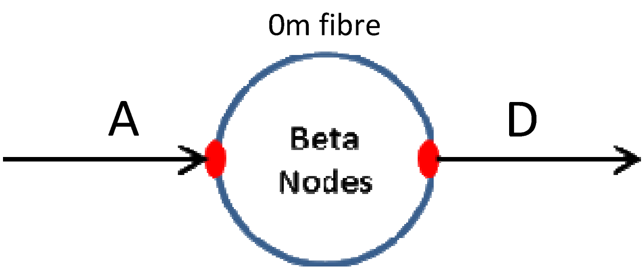


Figure 5-2: BETA Case 1 single unidirectional flow

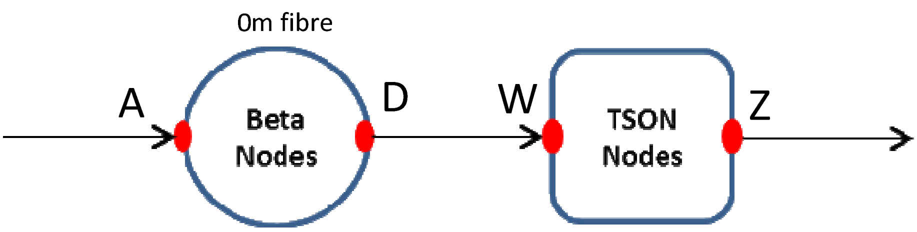


Figure 5-3: Integrated TSON and BETA Case 1 single unidirectional flow

The following tests have been conducted.

Scenario	Route BETA	Route TSON	Specification	Packet Size (byte)	Transmit rate(bps)	Receive rate(bps)
1a	A->D	W->Z	TSON Time-slice Allocation 1/2/3, BETA	64	1G	1G
			TSON Time-slice	64	1G	1G

			Allocation 4, BETA		2.2G	2.2G
1b	A->D	W->Z	TSON Time-slice Allocation 1/2/3, BETA	1500	1G	1G
			TSON Time-slice Allocation 4, BETA	1500	1G	1G
					2.8G	2.8G

Table 5-1: Test Scenarios

5.1.1 Latency result

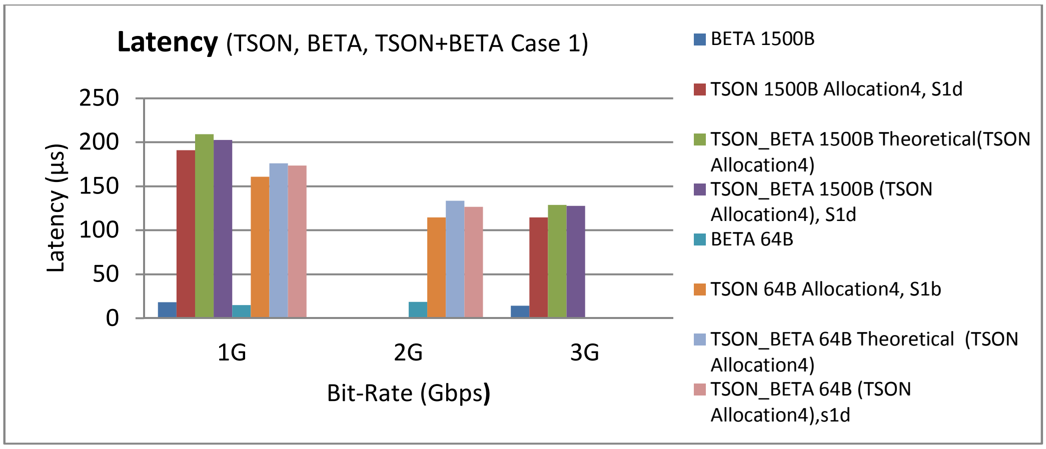


Figure 5-4: Latency Results for TSON, BETA, TSON_BETA case1

Case 1 Latency Analysis

The figure above shows the measured latency results for different Ethernet stream bit rate with Ethernet frame size 64B and 1500B in time slice allocation4. The latency with 3Gbps 1500B input stream of BETA is 14.192µs, and TSON is 114.492µs. The difference is because of TSON's burst mode needs more time to aggregate and segregate the bursts. The latency of TSON+BETA in 3Gbps (127.525µs) is a bit lower than the theoretical sum of TSON latency and BETA latency (128.684µs).

5.1.2 Jitter Result

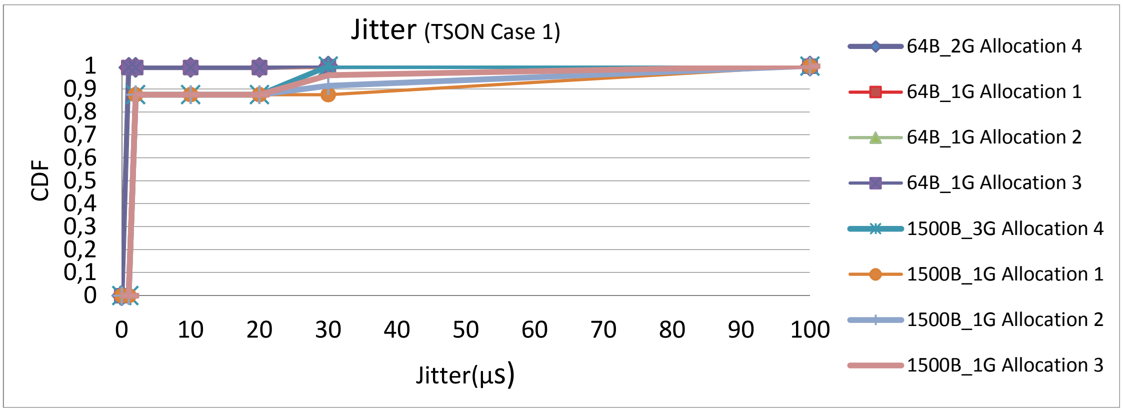


Figure 5-5: Jitter Results for TSON case1

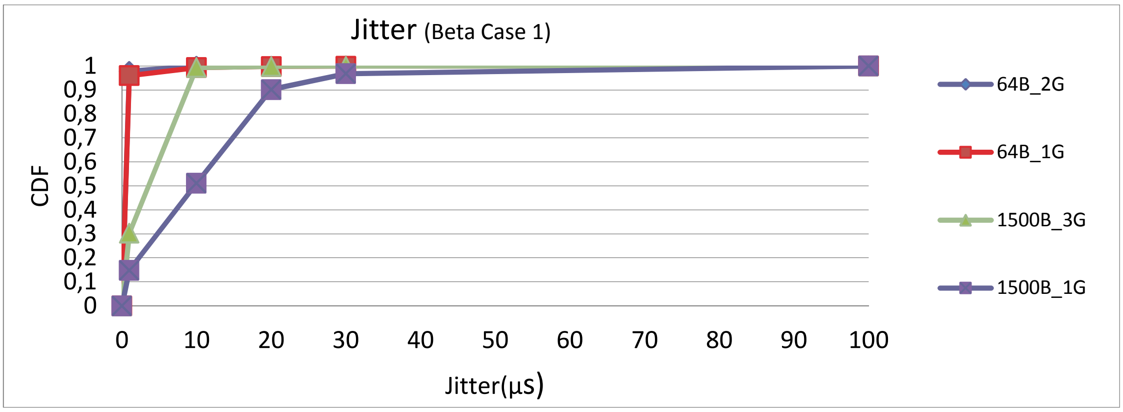


Figure 5-6: Jitter Results for BETA case1

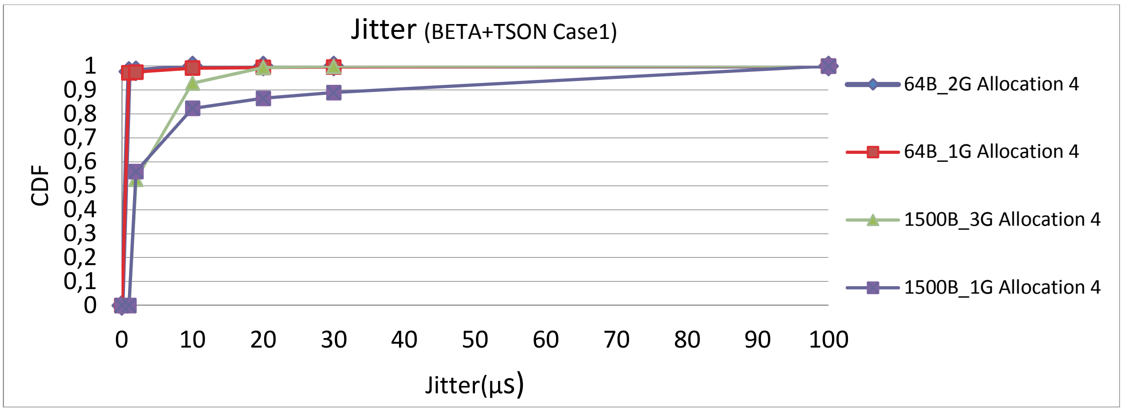


Figure 5-7: Jitter Results for BETA+TSON case1

Case 1 Jitter Analysis

The figures above show the measured jitter results for different Ethernet stream bit rate with Ethernet frame size 64B and 1500B in time slice allocation 1,2,3,4. For TSON, in any time-slice allocation, for Ethernet frame size 64B, about 99.35% of the Ethernet frames arrive in 1μs; for Ethernet frame size 1500B, about 87.5% of the Ethernet frames arrive in

2µs. For BETA with Ethernet frame size 64B, about 96% of the Ethernet frames arrive in 1µs; when the Ethernet frame size is 1500B, about 90% of the Ethernet frames arrive in 20µs. The jitter of TSON+BETA is about 97% received in 1µs for 64B traffic and about 82% received in 10µs for 1500B traffic

5.2 Case 2: Single Unidirectional Flow

The single unidirectional flow in the reverse direction test case provides a simple entry level test to provide a reference set of results for comparison with the forward direction single unidirectional flow described in 5.1 and the follow on test cases. The characteristics of a unidirectional flow are recorded for a range of transmission values, each for the TSON network domain and the beta network domain. The collective of these 2 sets of results is then compared with the integrated TSON and Beta end to end results.

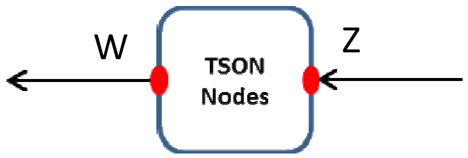


Figure 5-1: TSON case 2 single unidirectional flow.

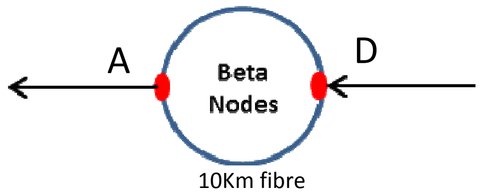


Figure 5-2: BETA case 2 single unidirectional flow

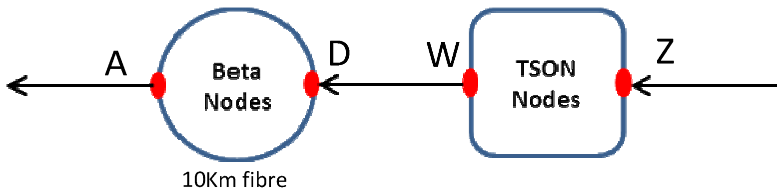


Figure 5-3: Integrated TSON and BETA case 2 single unidirectional flow

The following tests have been conducted.

Scenario	Route BETA	Route TSON	Specification	Packet Size (byte)	Transmit rate(bps)	Receive rate(bps)
1a	D->A	Z->W	TSON Time-slice Allocation 1/2/3, BETA	64	1G	1G
			TSON Time-slice Allocation 4, BETA	64	1G	1G
					2.2G	2.2G

1b	D->A	Z->W	TSON Time-slice Allocation 1/2/3, BETA	1500	1G	1G
			TSON Time-slice Allocation 4, BETA	1500	1G	1G
					2.8G	2.8G

Table 5-2: Test Scenarios

5.2.1 Latency Result

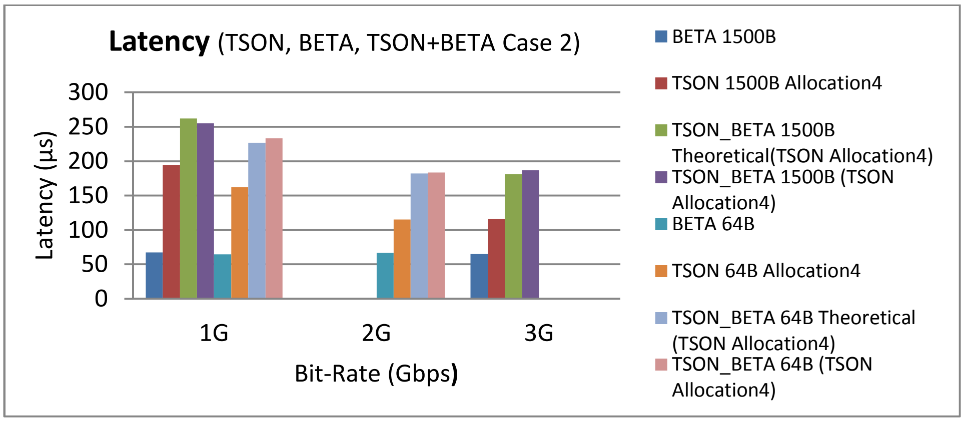


Figure 5-4: Latency results for TSON, BETA, TSON+BETA case 2

Case 2 Latency Analysis

The figure above shows the measured latency results for different Ethernet stream bit rate with Ethernet frame size 64B and 1500B in time slice allocation4. The difference between case 1 and case 2 is, for BETA, there is a 10Km fiber connected in the link. So the latency of BETA is about 50µs more than case 1. The latency with 3Gbps 1500B input stream of BETA is 65.235µs, and TSON is similar to case 1, 115.995µs. The latency of TSON+BETA in 3Gbps (186.902µs) is a bit lower than the theoretical sum of TSON latency and BETA latency (181.23µs).

5.2.2 Jitter Result

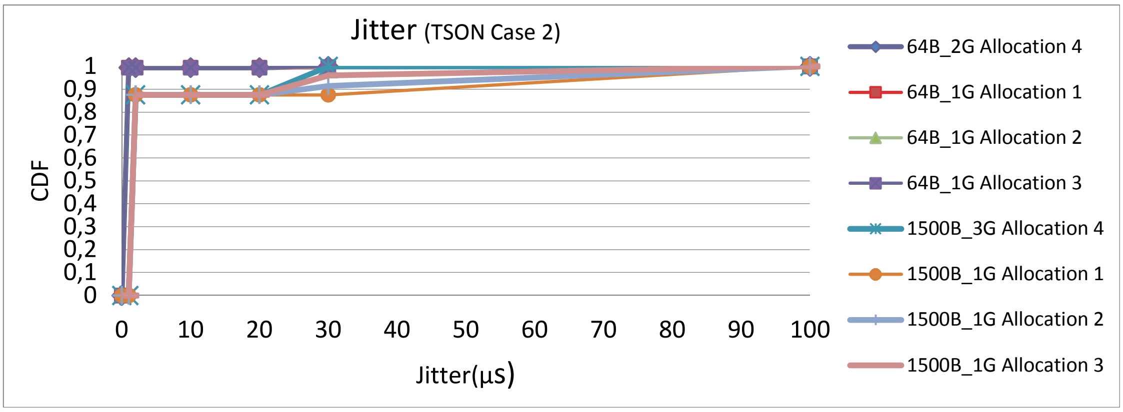


Figure 5-5: Jitter results for TSON case 2

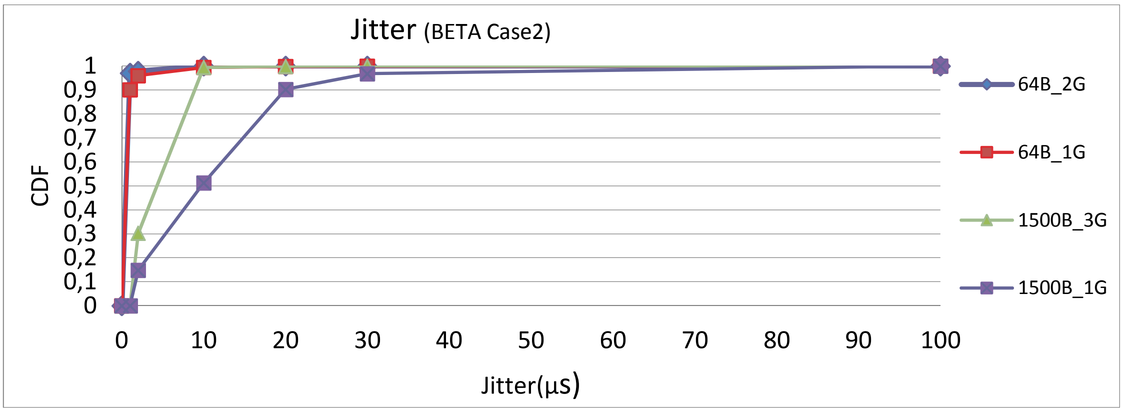


Figure 5-6: Jitter results for BETA case 2

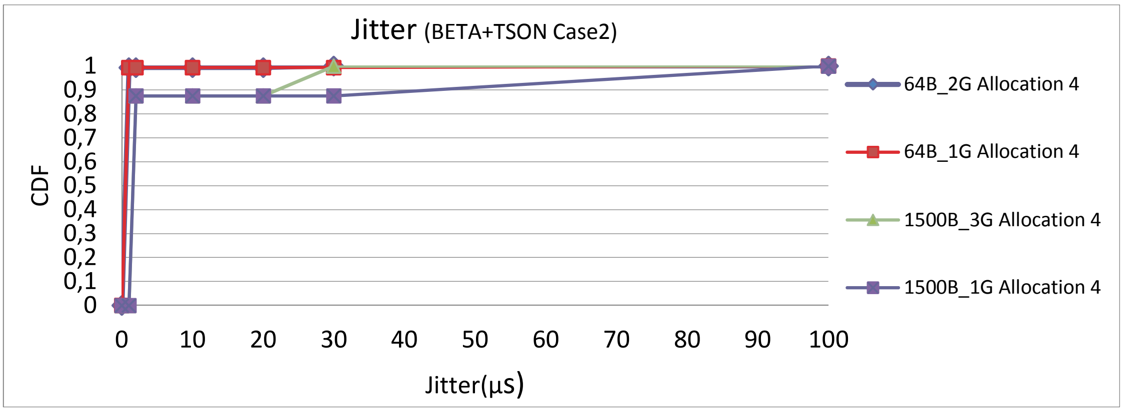


Figure 5-7: Jitter results for TSON+BETA case 2

Case 2 Jitter Analysis

The figures above show the measured jitter results for different Ethernet stream bit rate with Ethernet frame size 64B and 1500B in time slice allocation 1,2,3,4. The result is similar to Case1. For TSON, in any time-slice allocation, for Ethernet frame size 64B, about 99.35% of the Ethernet frames arrive in 1µs; for Ethernet frame size 1500B, about 87.5% of the Ethernet frames arrive in 2µs. For BETA with Ethernet frame size 64B, about 90% of the Ethernet frames arrive in 1µs; when the Ethernet frame size is 1500B, about 90% of the Ethernet frames arrive in 20µs. The jitter of TSON+BETA is about 99.35% received in 1µs for 64B traffic and about 87.5% received in 2µs for 1500B traffic.

5.3 Case 3: Single bidirectional Flow

The single bidirectional flow test case increases the loading on each system. The characteristics of a bidirectional flow are recorded for a range of transmission values, each for the TSON network domain and the beta network domain. The collective of these 2 sets of results is then compared with the integrated TSON and Beta end to end results, and the unidirectional end to end results.

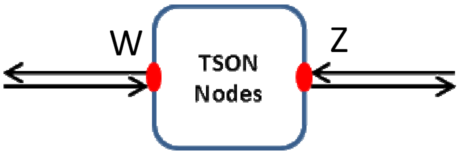


Figure 5-15: TSON case 2 **bi**directional flow

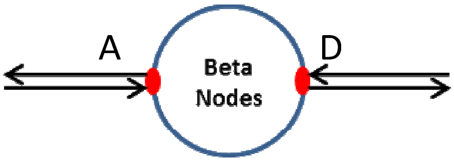


Figure 5-8 : BETA case 3 bidirectional flow

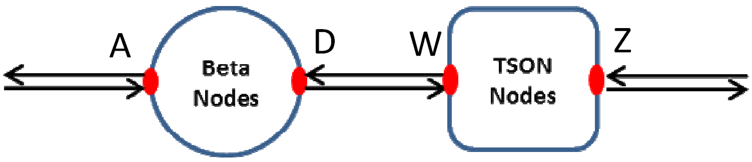


Figure 5-9 : Integrated TSON and Beta case 2 bidirectional flow

For case 3, it is a combination of case 1 and case 2. The following tests have been conducted.

Scenario	Route BETA	Route TSON	Specification	Packet Size (byte)	Transmit rate(bps)	Receive rate(bps)
1a	A->D	W->Z	TSON Time-slice Allocation 1/2/3, BETA	64	1G	1G
			TSON Time-slice Allocation 4, BETA	64	1G	1G
					2.2G	2.2G
1b	A->D	W->Z	TSON Time-slice Allocation 1/2/3, BETA	1500	1G	1G
			TSON Time-slice Allocation 4, BETA	1500	1G	1G
					2.8G	2.8G
2a	D->A	Z->W	TSON Time-slice Allocation 1/2/3, BETA	64	1G	1G
			TSON Time-slice Allocation 4, BETA	64	1G	1G
					2.2G	2.2G
2b	D->A	Z->W	TSON Time-slice Allocation 1/2/3, BETA	1500	1G	1G
			TSON Time-slice	1500	1G	1G

			Allocation 4, BETA		2.8G	2.8G

Table 5-3: Test Scenarios

5.3.1 Latency Result

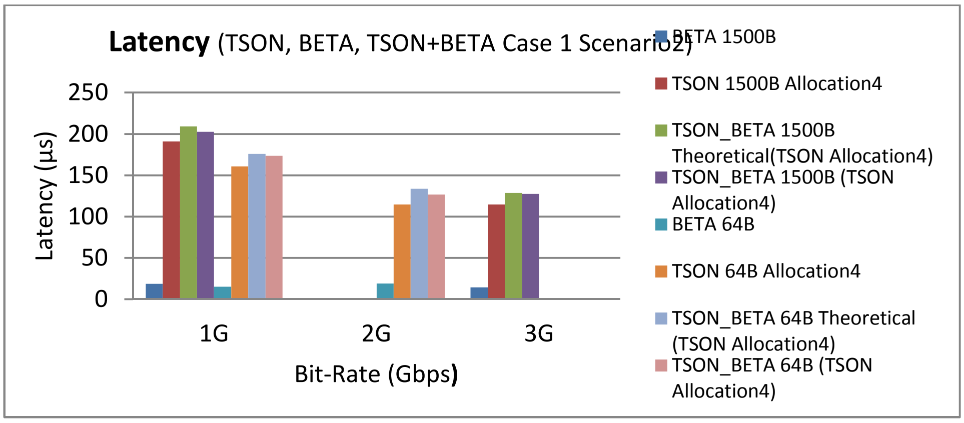


Figure 5-10: Latency results for TSON, BETA, TSON+BETA case 3, Scenario 1

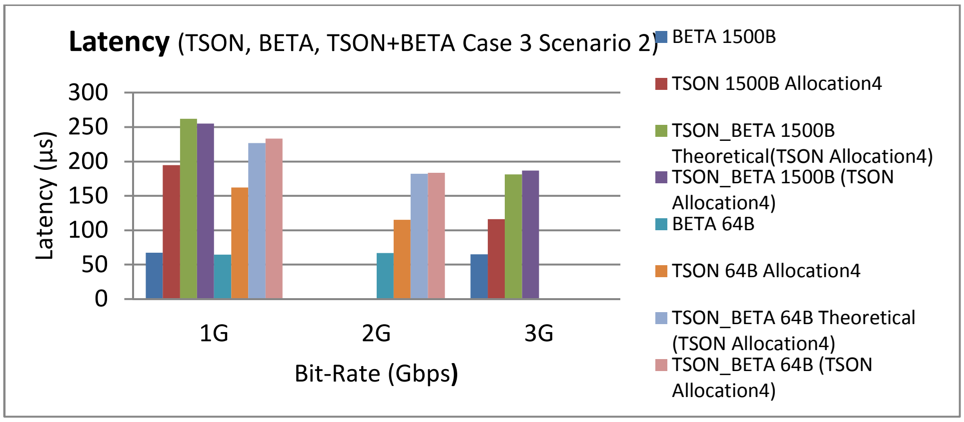


Figure 5-11: Latency results for TSON, BETA, TSON+BETA case 3, Scenario 2

Case 3 Latency Analysis

Case3 is a combination of case 1 and case 2.The latency result of scenario 1 is the same experiment of case1, and the latency result of scenario 2 is the same experiment of case2.

5.3.2 Jitter Result

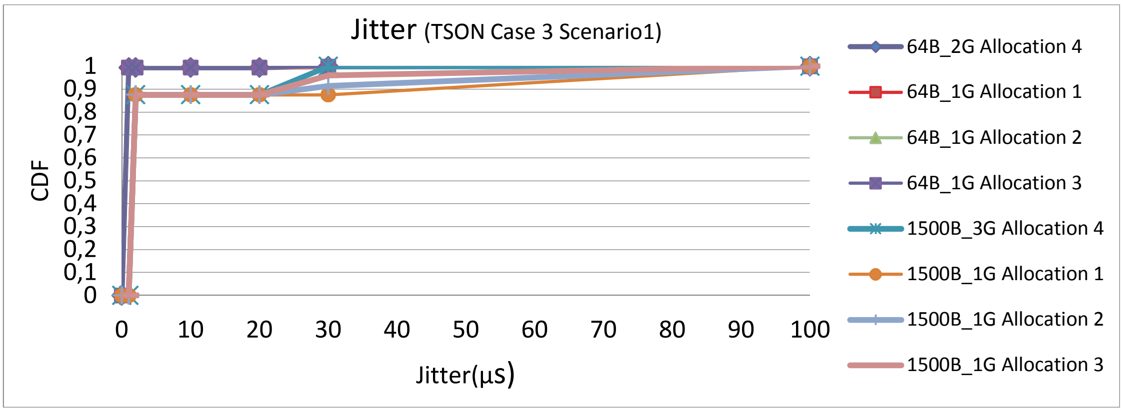


Figure 5-12: Jitter results for TSON case 3, Scenario1

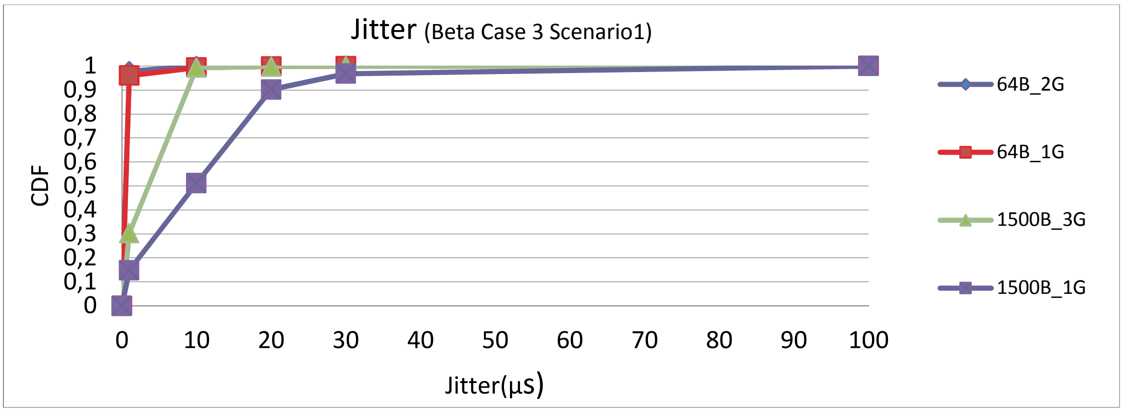


Figure 5-13: Jitter results for BETA case 3, Scenario1

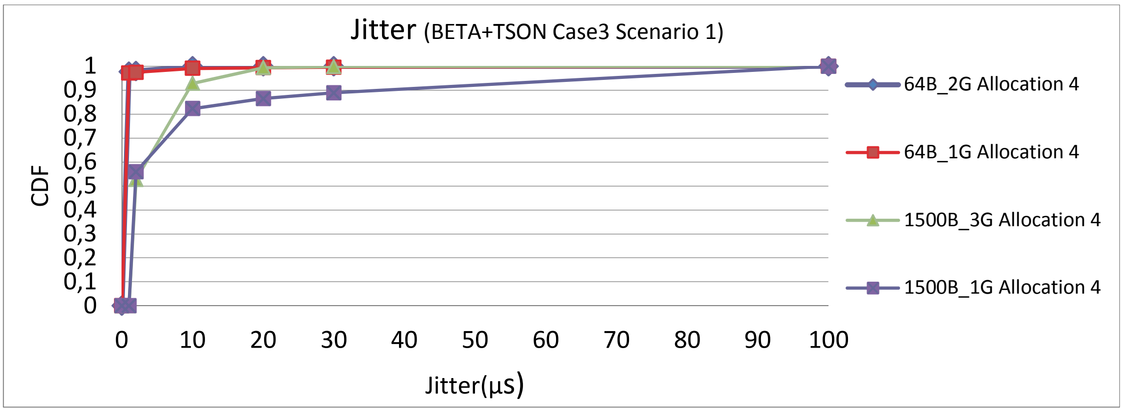


Figure 5-14: Jitter results for BETA+TSON case 3, Scenario1

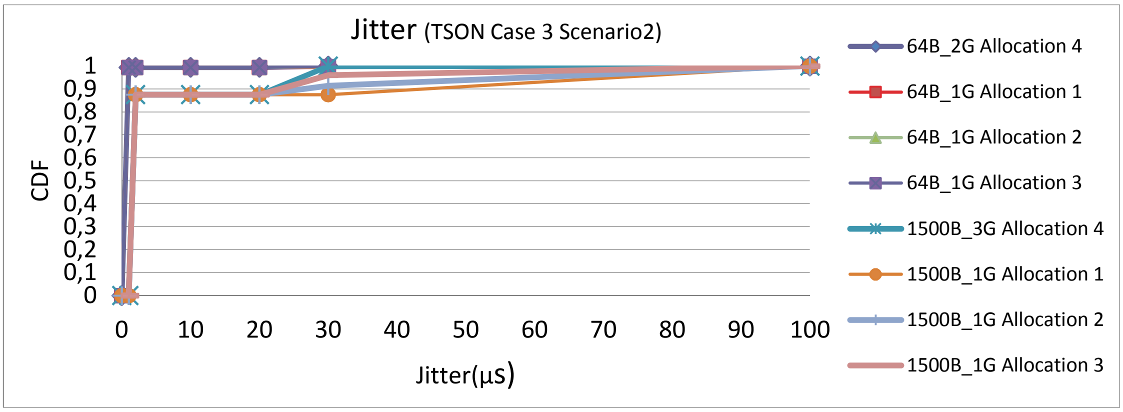


Figure 5-15: Jitter results for TSON case 3, Scenario2

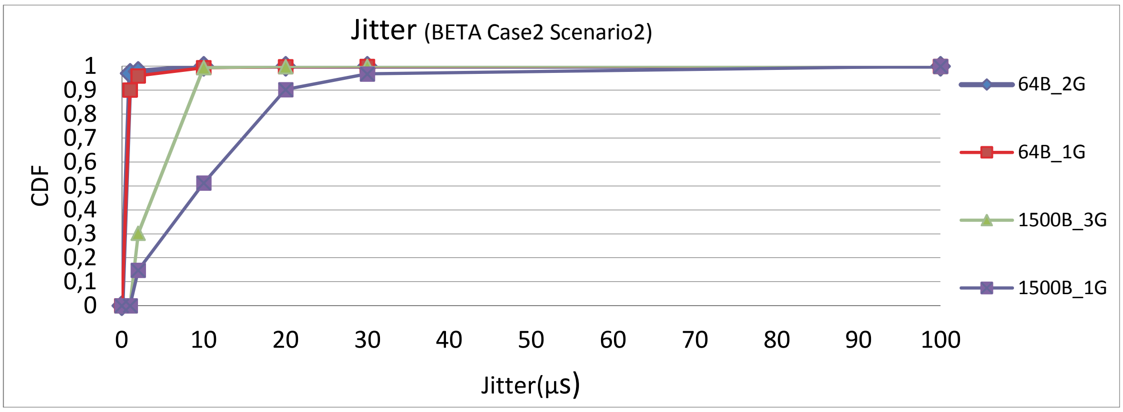


Figure 5-24: Jitter results for BETA case 3, Scenario2

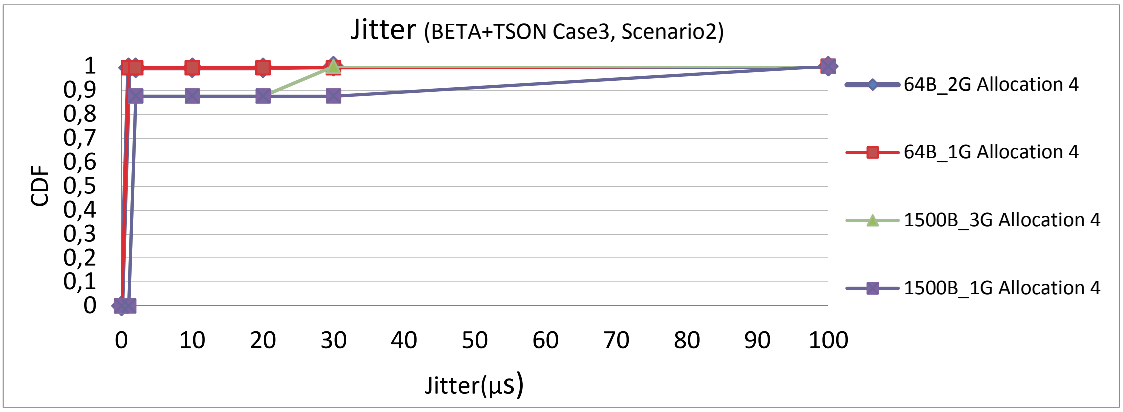


Figure 5-16: Jitter results for TSON+BETA case 3, Scenario2

Case 3 Jitter Analysis

Case3 is a combination of case 1 and case 2.The jitter result of scenario 1 is the same experiment of case1, and the jitter result of scenario 2 is the same experiment of case2.

6 TSON/ OPST (Beta) Test Cases and analysis: Multi-Technology multipoint-to-point sub-wavelength transport service

6.1 Case 4: Flow aggregation Unidirectional

The flow aggregation unidirectional test case considers the aggregation of two independent flows ingressing via two single ports in the first sub lambda domain (Beta) and traversing the second sub wavelength domain (TSON) as two independent flows but now via a single ingress and egress port. The characteristics of the two aggregated flows are recorded for a range of transmission values, each for the TSON network domain and the beta network domain.

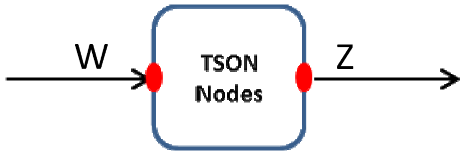


Figure 6-1 : TSON case 1 single unidirectional flow

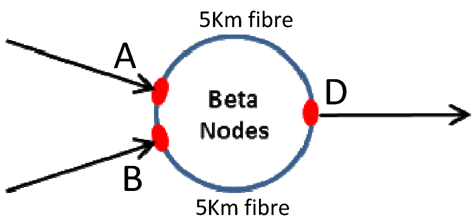


Figure 6-2 : BETA case 4 aggregation unidirectional flow

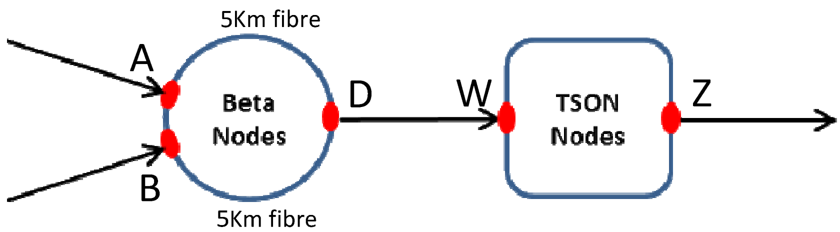


Figure 6-3: Integrated TSON and Beta case 4 aggregation unidirectional flow

The following tests have been conducted.

Scenario	Route BETA	Route TSON	Specification	Packet Size (byte)	Transmit rate(bps)	Receive rate(bps)
1a	A->D B->D	W->Z	TSON Time-slice Allocation 1/2/3, BETA	64	1G	1G
			TSON Time-slice	64	1G	1G

			Allocation 4, BETA		2.2G	2.2G
1b	A->D B->D	W->Z	TSON Time-slice Allocation 1/2/3, BETA	1500	1G	1G
			TSON Time-slice Allocation 4, BETA	1500	1G	1G
					2.8G	2.8G

Table 6-1: Test Scenarios

6.1.1 Latency Result

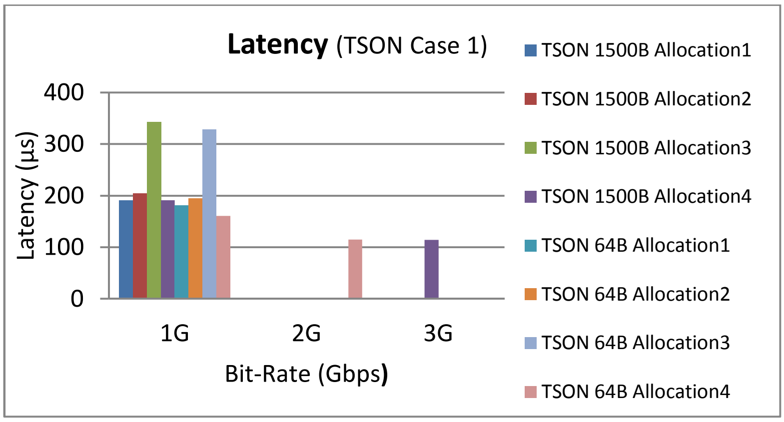


Figure 6-4: Latency results for TSON case 1

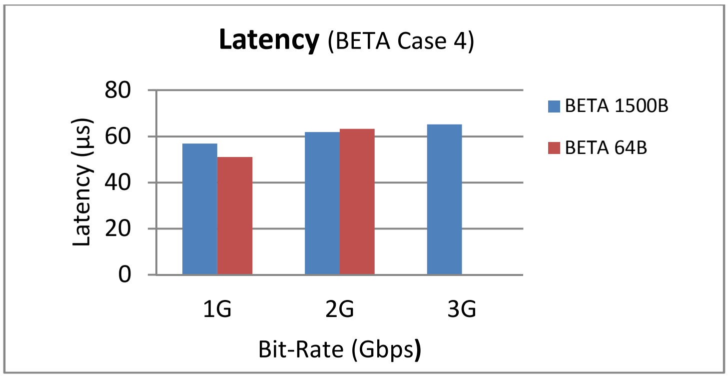


Figure 6-5: Latency results for BETA case4

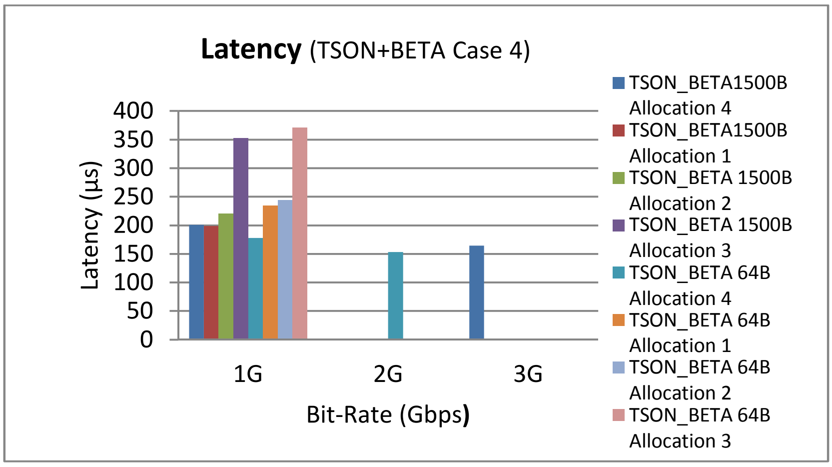


Figure 6-6: Latency results for TSON+BETA case 4

Case 4 Latency Analysis

The figure above shows the measured latency results for different Ethernet stream bit rate with Ethernet frame size 64B and 1500B in different time slice allocation 1,2,3,4. The latency with 3Gbps 1500B input stream of BETA is 65.283µs, and the latency of TSON is affected by the time-slice allocation, for 1Gbps 1500B input stream, the latency of time-slice allocation2 (204.304µs) is a bit higher than time-slice allocation1 and time-slice allocation4 (190.824µs), while the latency of time-slice allocation3 (342.447µs) is the highest among the 4 allocations. The latency of TSON+BETA is also affected by the TSON time-slice allocation. And the worst latency is on time-slice allocation3, which is 356.862µs for 1Gbps 1500B stream, and the best is on time-slice allocation4, which is 178.061µs for 1Gbps 1500B stream.

6.1.2 Jitter Result

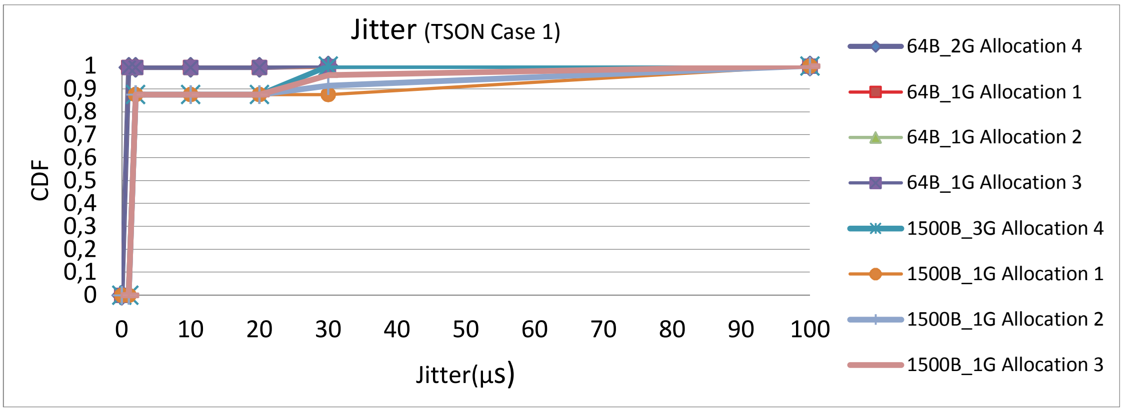


Figure 6-7: Jitter results for TSON case 4

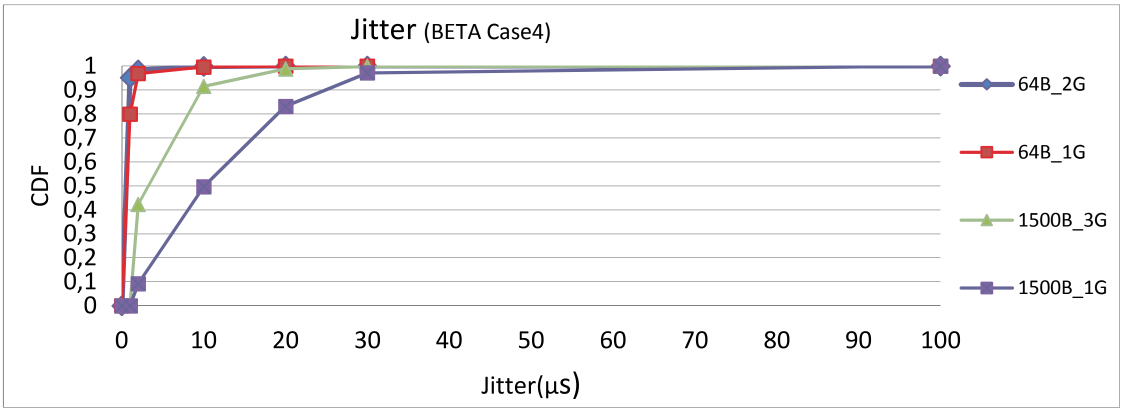


Figure 6-8: Jitter results for BETA case 4

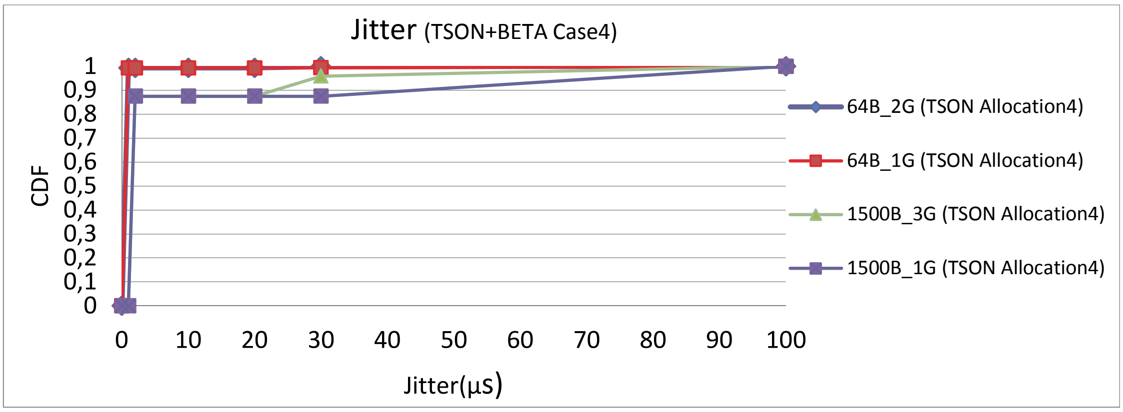


Figure 6-9: Jitter results for TSON+BETA case 4

Case 4 Jitter Analysis

The figures above show the measured jitter results for different Ethernet stream bit rate with Ethernet frame size 64B and 1500B in time slice allocation 1,2,3,4. For TSON, it is with single unidirectional flow, the jitter results are similar to the case1. For BETA, it is with aggregation unidirectional flow, with Ethernet frame size 64B, about 79% of the Ethernet frames arrive in 1µs; when the Ethernet frame size is 1500B, about 83% of the Ethernet frames arrive in 20µs. The jitter of TSON+BETA is about 99.35% received in 1µs for 64B traffic and about 87.5% received in 2µs for 1500B traffic.

6.2 Case 5: Flow aggregation Unidirectional

The Flow aggregation unidirectional in the reverse direction test case provides results for comparison with the forward direction flow aggregation unidirectional test case as described in 8.1 The flow aggregation unidirectional test case considers the aggregation of two independent flows ingressing via two single ports in the first sub lambda domain (TSON) and traversing the second sub wavelength domain (Beta) as two independent flows but now via a single ingress and egress port. The characteristics of the two aggregated flows are recorded for a range of transmission values, each for the TSON network domain and the beta network domain.

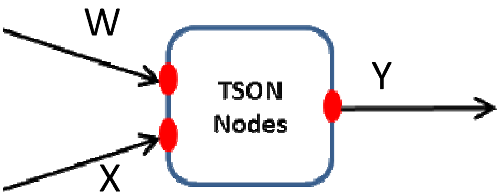


Figure 6-10: TSON case 4 aggregation unidirectional flow

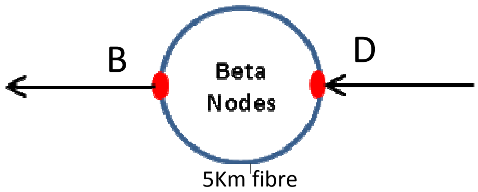


Figure 6-11: BETA case 2 single unidirectional flow

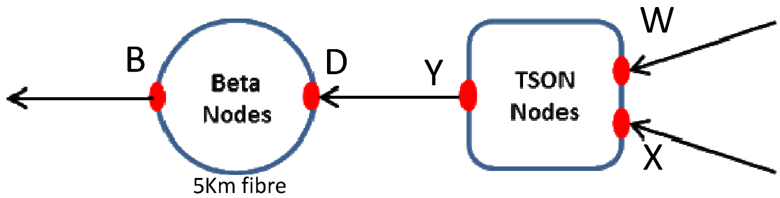


Figure 6-12: Integrated TSON and Beta case 5 unidirection

50%	33%
50%	66%

The following tests have been conducted.

Scenario	Route BETA	Route TSON	Specification	Packet Size (byte)	Transmit rate(bps)	Receive rate(bps)
1a	D->B	W->Y	TSON Time-slice Allocation 1/2/3, BETA	64	1G	1G
			TSON Time-slice Allocation 4, BETA, flow 50%/50%,33%/66%	64	1G	1G
					2.2G	2.2G
1b	D->B	X->Y	TSON Time-slice Allocation 1/2/3, BETA	1500	1G	1G
			TSON Time-slice Allocation 4, BETA, flow 50%/50%,33%/66%	1500	1G	1G
					2.8G	2.8G

Table 6-2: Test Scenarios

6.2.1 Latency Result

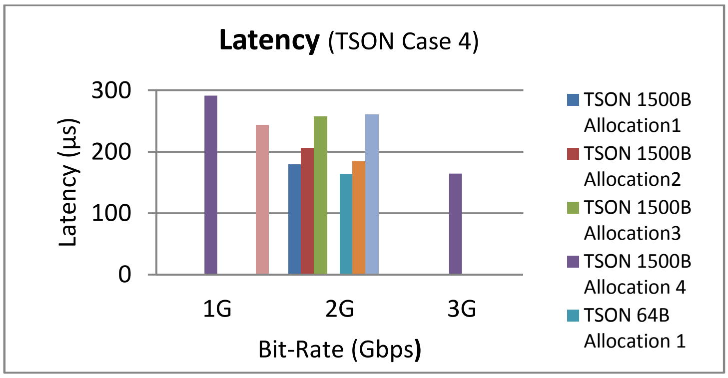


Figure 6-13: Latency results for TSON case4

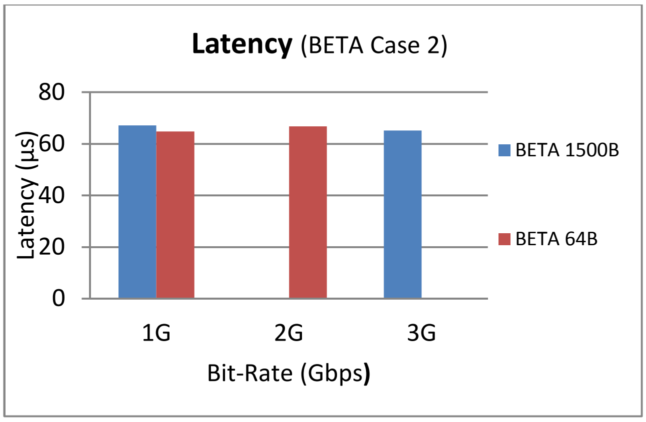


Figure 6-14: Latency results for BETA case2

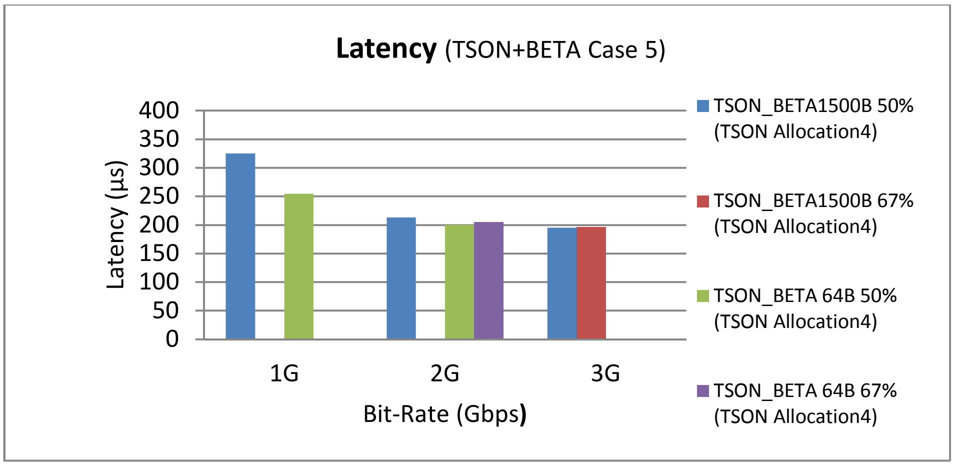


Figure 6-15: Latency results for TSON+BETA case5

Case 5 Latency Analysis

The figure above shows the measured latency results for different Ethernet stream bit rate with Ethernet frame size 64B and 1500B in different time slice allocation 1,2,3,4. The measurement also considered the condition of different percentage of the two independent aggregation flows, such as 50%/50% and 33%/67%. The latency of BETA is same as case2, and the latency of TSON is as case4. The latency of TSON+BETA is affected by the percentage of the two aggregation flows, for 2Gbps, Ethernet frame size 64B, the traffic with 67%/33% input has slightly higher latency(204.723µs) than 50%/50%(200.609µs.).

6.2.2 Jitter Result

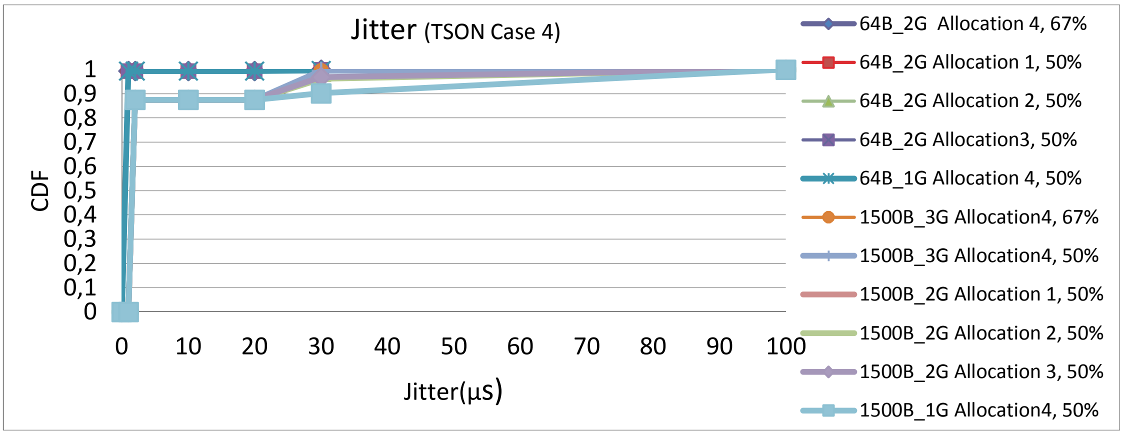


Figure 6-16: Jitter results for TSON case4

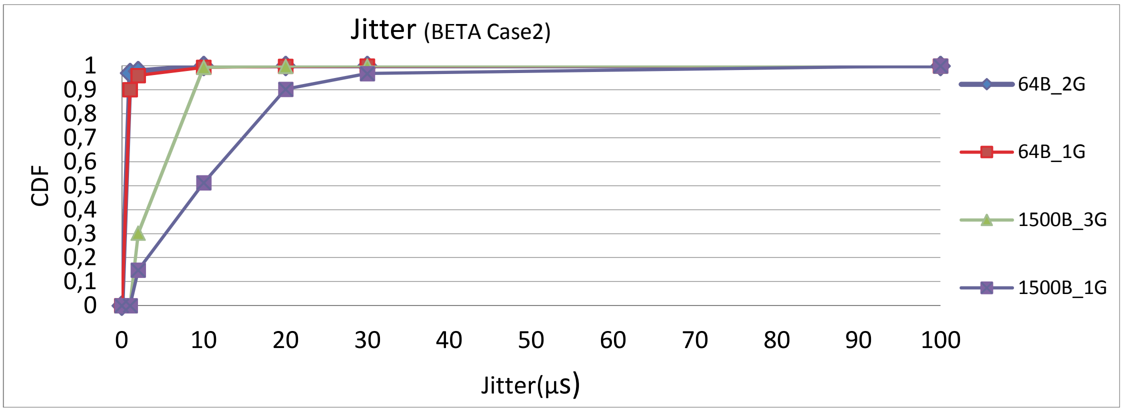


Figure 6-17: Jitter results for TSON case2

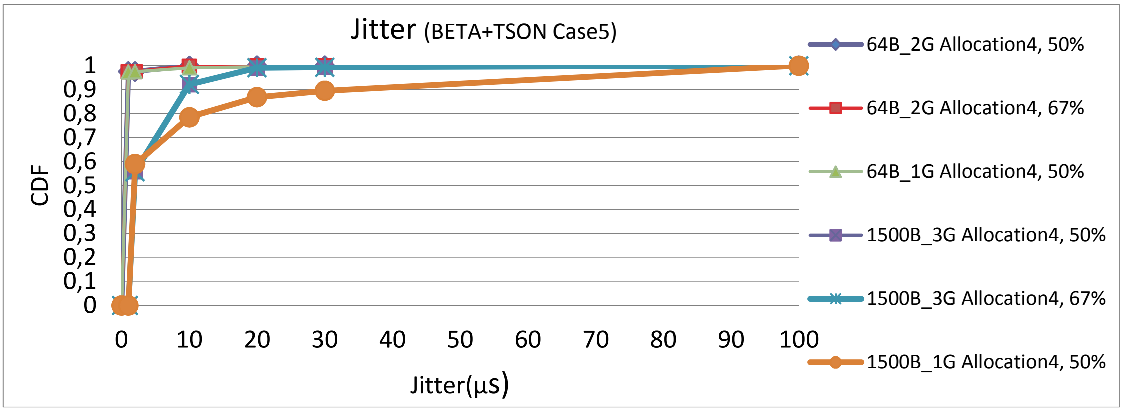


Figure 6-18: Jitter results for TSON case5

Case 5 Jitter Analysis

The figures above show the measured jitter results for different Ethernet stream bit rate with Ethernet frame size 64B and 1500B in time slice allocation 1,2,3,4. The measurement also considered the condition of different percentages of the two independent aggregation flows, such as 50%/50% and 33%/67%. For TSON, it is the same case as case4. For BETA, it is as same as case2. For the jitter of TSON+BETA, the different percentages of the two independent aggregation flows don't affect much on the jitter results. In case of 2Gbps Ethernet frame size 64B, with 50%/50% flows, the jitter result is about 97.54% received in 1µs, while with 33%/67%, the jitter result is about the same.

7 TSON/ OPST (Beta) Test Cases and analysis: Multi-Technology multipoint-to-multipoint sub-wavelength transport service

7.1 Case 6: Flow aggregation/segregation Unidirectional

The flow aggregation/segregation unidirectional flow test case considers the aggregation of two independent flows ingressing via two single ports in the first sub lambda domain (TSON), egressing the first domain and ingressing the second sub wavelength domain (Beta) via a single port, and egressing the second sub wavelength domain (Beta) as two independent flows via two separate ports. The characteristics of the two aggregated and segregated flows are recorded for a range of transmission values.

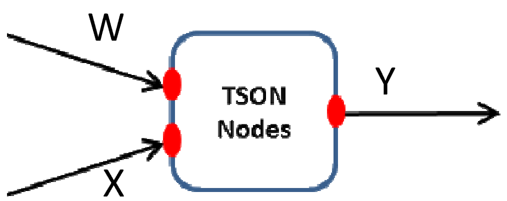


Figure 7-1: TSON case 4 aggregation unidirectional flow

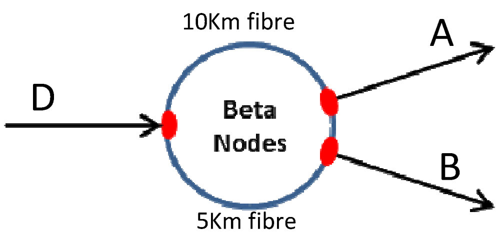


Figure 7-2: BETA case 5 segregation unidirectional flow

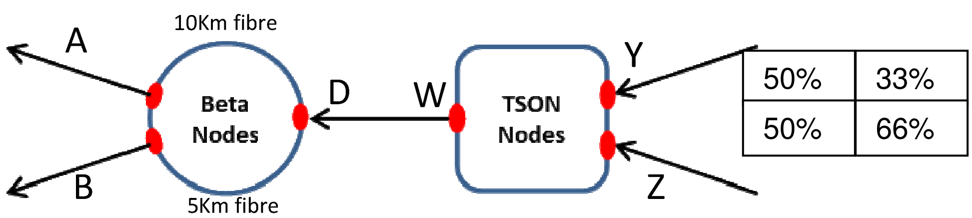


Figure 7-3: Integrated TSON and Beta case6 aggregation/segregation unidirectional flow

The following tests have been conducted.

Scenario	Route BETA	Route TSON	Specification	Packet Size	Transmit rate(bps)	Receive rate(bps)

				(byte)		
1a	D->A D->A	Y->W Z->W	TSON Time-slice Allocation 1/2/3, BETA	64	1G	1G
			TSON Time-slice Allocation 4, BETA, flow 50%/50%,33%/66%	64	1G	1G
					2.2G	2.2G
1b	D->A D->B	Y->W Z->W	TSON Time-slice Allocation 1/2/3, BETA	1500	1G	1G
			TSON Time-slice Allocation 4, BETA, flow 50%/50%,33%/66%	1500	1G	1G
					2.8G	2.8G

Table 7-1: Test Scenarios

7.1.1 Latency Result

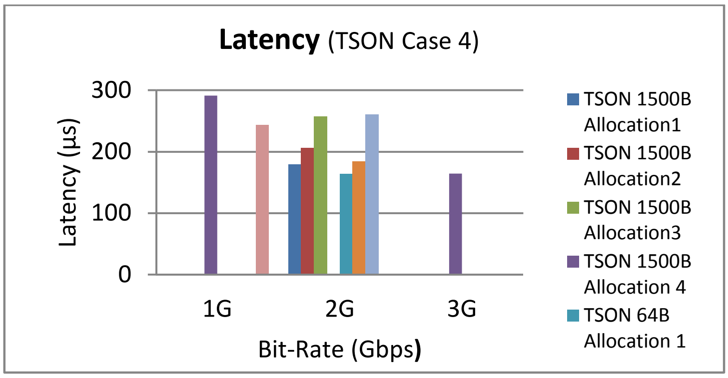


Figure 7-4: Latency results for TSON case4

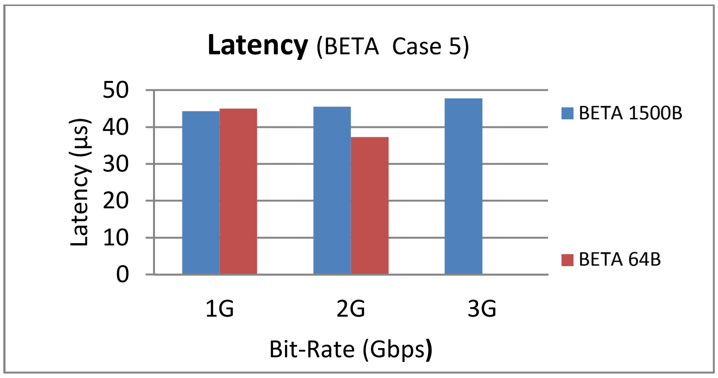


Figure 7-5: Latency results for TSON case 5

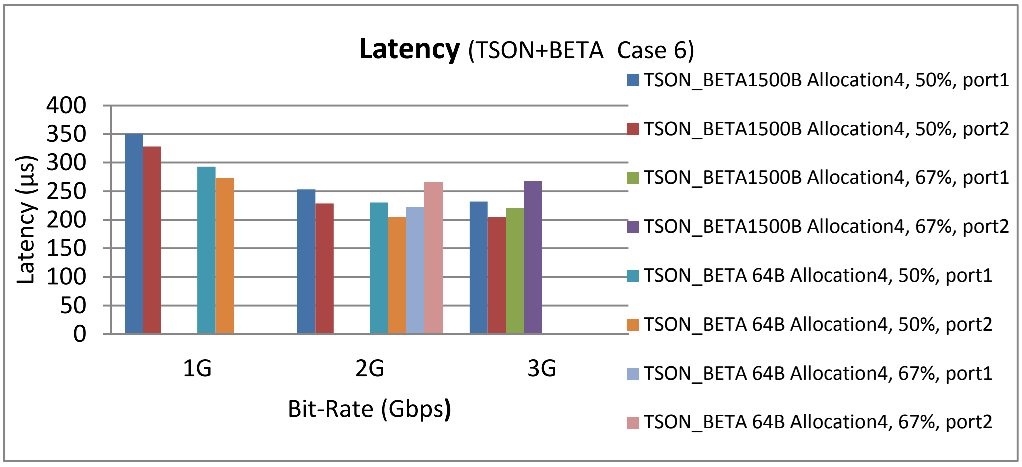


Figure 7-6: Latency results for TSON+BETA case 6

Case 6 Latency Analysis

The figure above shows the measured latency results for different Ethernet stream bit rate with Ethernet frame size 64B and 1500B in different time slice allocation 1,2,3,4. The measurement also considered the condition of different percentage of the two independent aggregation flows, such as 50%/50% and 33%/67%. The latency of BETA is same as case 5, and the latency of TSON is as case4. The latency of TSON+BETA is affected by the percentage of the two aggregation flows, for 2Gbps, Ethernet frame size 64B, the traffic with 50%/50% input has higher latency(230.258µs) than 33%/67%(222.793µs.). For TSON+BETA, the latency difference between PORT1 and PORT2 is because of the 5Km fiber.

7.1.2 Jitter Result

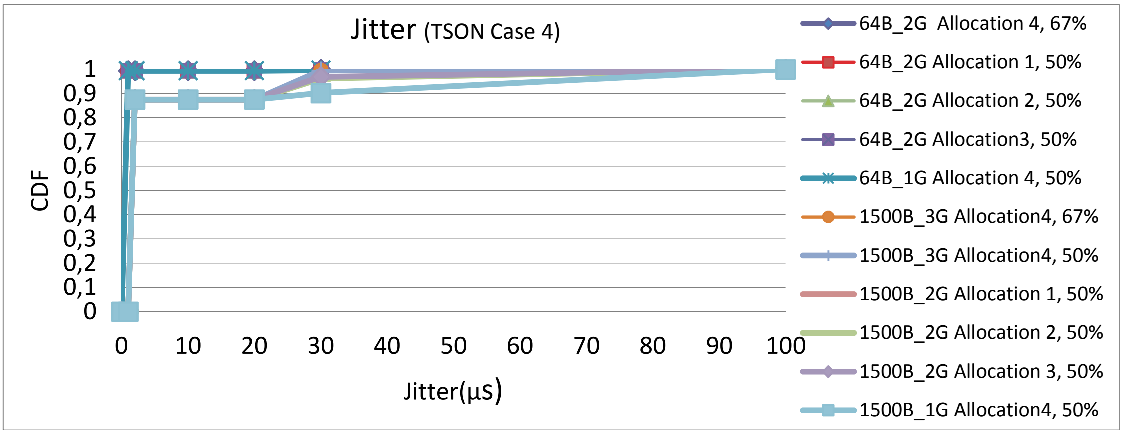


Figure 7-7: Jitter results for TSON case 4

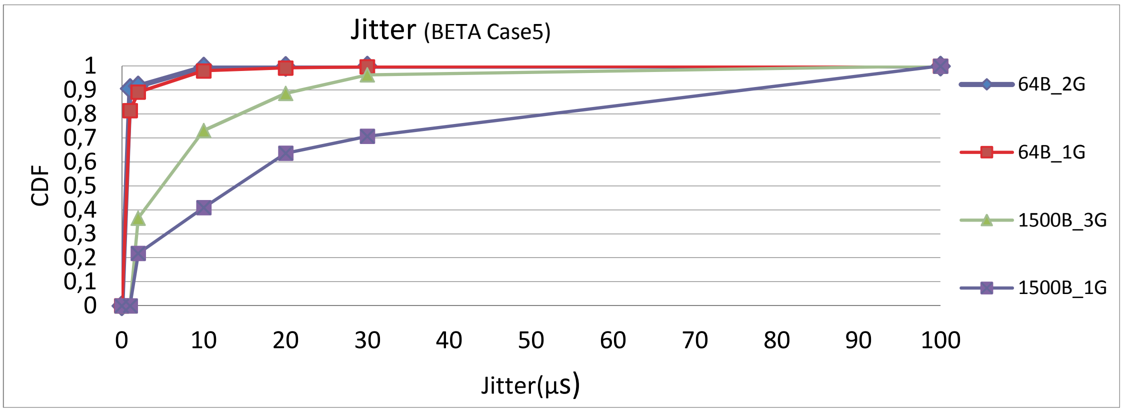


Figure 7-8: Jitter results for BETA case 5

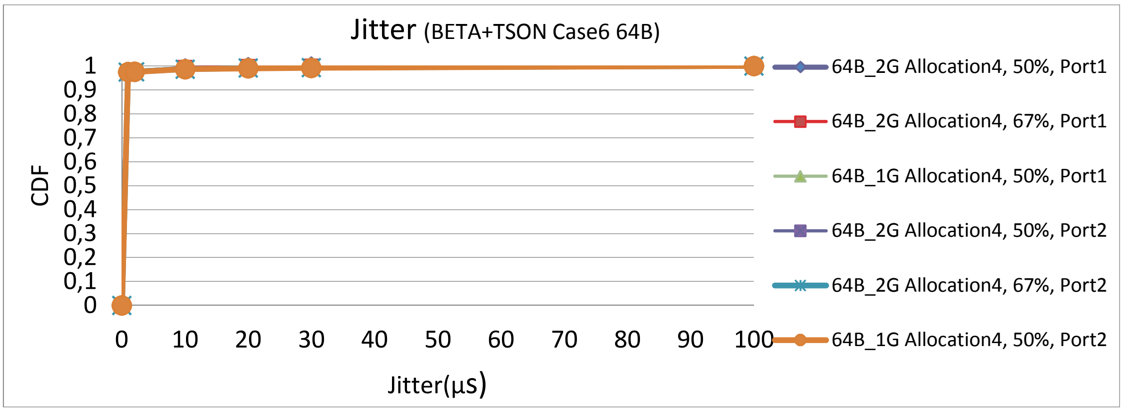


Figure 7-9: Jitter results for BETA+TSON case 6, Ethernet Frame size 64B

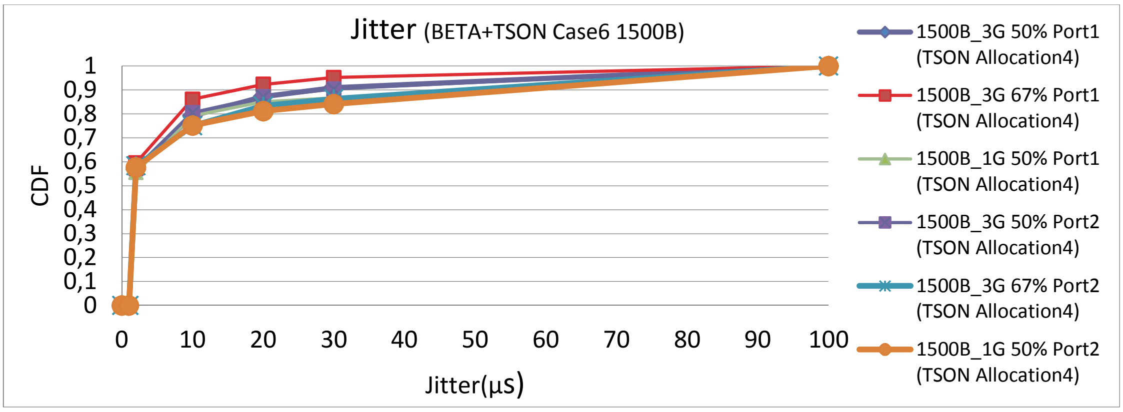


Figure 7-10: Jitter results for BETA+TSON case6, Ethernet Frame size 1500B

Case 6 Jitter Analysis

The figures above show the measured jitter results for different Ethernet stream bit rate with Ethernet frame size 64B and 1500B in time slice allocation 1,2,3,4. The measurement also considered the condition of different percentages of the two independent aggregation flows, such as 50%/50% and 33%/67%. For TSON, it is the same case as case4. For

BETA, it is as same as case5. For the jitter of TSON+BETA, the different percentages of the two independent aggregation flows don't affect much on the jitter results. In case of 3Gbps Ethernet frame size 1500B, with 50%/50% flows, the jitter result is about 56.84% received in 2µs, while with 33%/67%, the jitter result is about 59.576%.

7.2 Case 7: Flow aggregation/segregation Unidirectional

The flow aggregation/segregation unidirectional flow in the reverse direction test case considers the aggregation of two independent flows ingressing via two single ports in the first sub lambda domain (Beta), egressing the first domain and ingressing the second sub wavelength domain (TSON) via a single port, and egressing the second sub wavelength domain (TSON) as two independent flows via two separate ports. The characteristics of the two aggregated and segregated flows in the reverse direction are recorded for a range of transmission values.

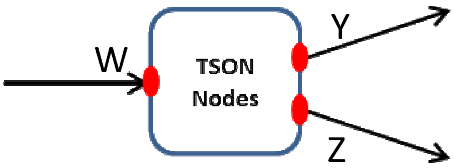


Figure 7-11: TSON case 5 segregation unidirectional flow

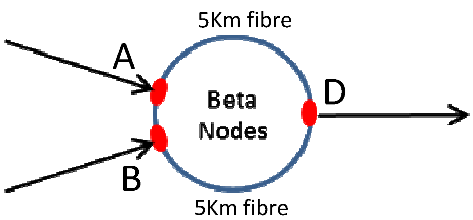


Figure 7-12: BETA case 4 aggregation unidirectional flow

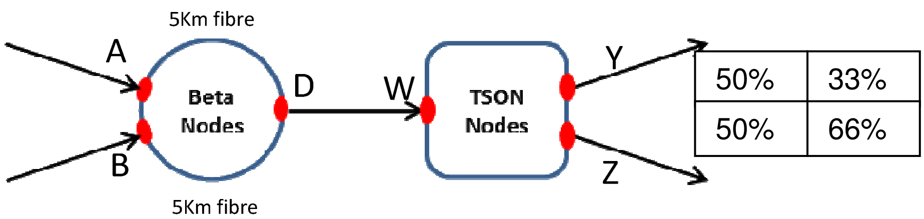


Figure 7-13: Integrated TSON and Beta case 7 aggregation/segregation unidirectional flow

The following tests have been conducted.

Scenario	Route BETA	Route TSON	Specification	Packet Size (byte)	Transmit rate(bps)	Receive rate(bps)
1a	A->D B->D	W->Y W->Z	TSON Time-slice Allocation 1/2/3, BETA	64	1G	1G
			TSON Time-slice Allocation 4, BETA, flow 50%/50%,33%/66%	64	1G	1G
					2.2G	2.2G
1b	A->D B->D	W->Y W->Z	TSON Time-slice Allocation 1/2/3, BETA	1500	1G	1G
			TSON Time-slice Allocation 4, BETA, flow 50%/50%,33%/66%	1500	1G	1G
					2.8G	2.8G

Table 7-2: Test Scenarios

7.2.1 Latency Result

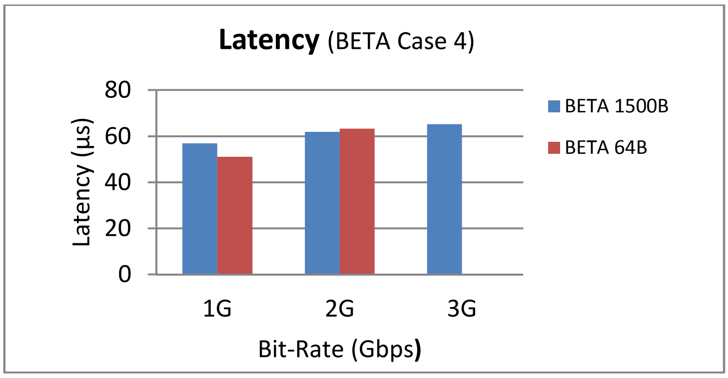


Figure 7-14: Latency results for BETA case4

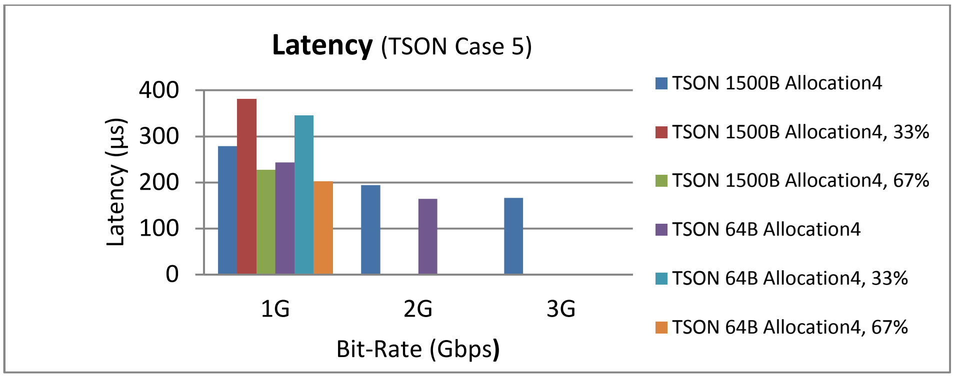


Figure 7-15: Latency results for TSON case5

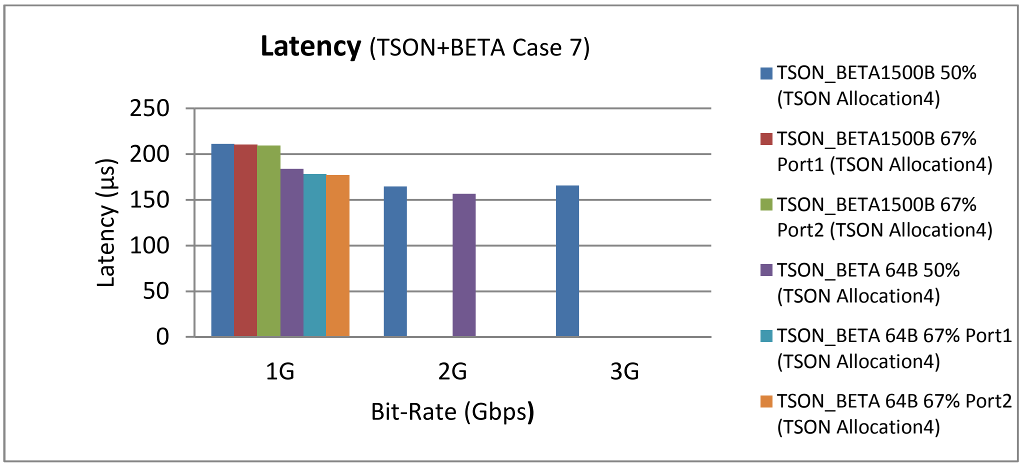


Figure 7-16: Latency results for TSON+BETA case7

Case 7 Latency Analysis

The figure above shows the measured latency results for different Ethernet stream bit rate with Ethernet frame size 64B and 1500B in different time slice allocation 1,2,3,4. The measurement also considered the condition of different percentage of the two independent aggregation flows, such as 50%/50% and 33%/67%. The latency of BETA is same as case4, and the latency of TSON is as case5. For the latency of TSON+BETA, it is not affected by the percentage of the two aggregation flows, such as for 1Gbps, Ethernet frame size 1500B, the traffic with 50%/50% and 33%/67% has similar latency in the output side(211.176µs, 210.611µs, 209.433µs).

7.2.2 Jitter Result

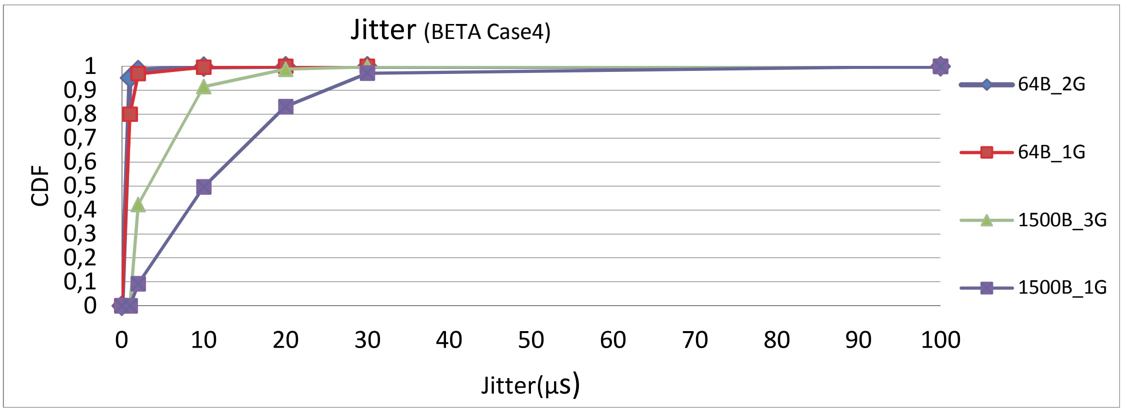


Figure 7-17: Jitter results for BETA case4

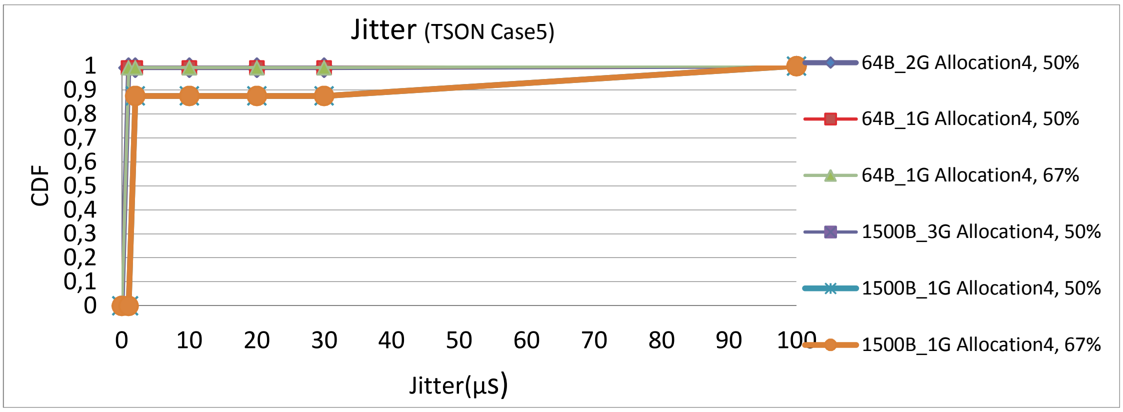


Figure 7-18: Jitter results for TSON case5

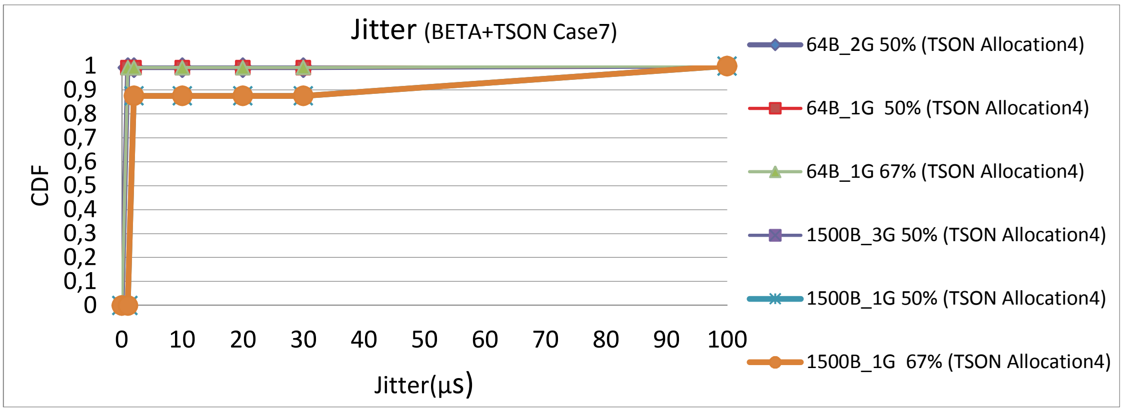


Figure 7-19: Jitter results for BETA+TSON case7

Case 7 Jitter Analysis

The figures above show the measured jitter results for different Ethernet stream bit rate with Ethernet frame size 64B and 1500B in time slice allocation1,2,3,4.The measurement also considered the condition of different percentages of the two independent aggregation flows, such as 50%/50% and 33%/67%. For TSON, it is the same case as case5. For BETA, it is as same as case4. For the jitter of TSON+BETA, the result is similar to the TSON jitter results because it goes through BETA first then TSON. The different percentages of the two independent aggregation flows don't affect much on the jitter results.

7.3 Case 8: Flow aggregation and flow segregation partial bidirectional

The flow aggregation and flow segregation partial bidirectional tests case further stresses the sub wavelength network domains by introducing two independent unidirectional flows operating in alternate directions via independent ports, but utilising a single interconnect port operating bidirectional flows. The characteristics of the two aggregated and segregated flows are recorded for a range of transmission values.

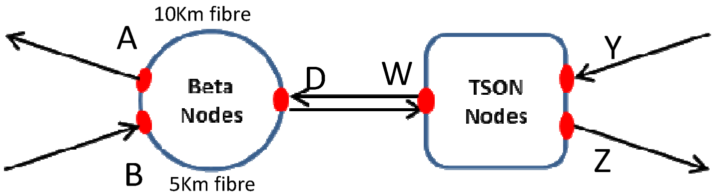


Figure 7-20: Integrated TSON and Beta case 4 aggregation and segregation partial bidirectional flow

Scenario	Route BETA	Route TSON	Specification	Packet Size (byte)	Transmit rate(bps)	Receive rate(bps)
1a	B->D	W->Z	TSON Time-slice Allocation1, BETA	64	1G	1G
					2.2G	2.2G
1b	B->D	W->Z	TSON Time-slice Allocation1, BETA	1500	1G	1G
					2.8G	2.8G
2a	D->A	Y->W	TSON Time-slice Allocation1, BETA	64	1G	1G
					2.2G	2.2G
2b	D->A	Y->W	TSON Time-slice Allocation1, BETA	1500	1G	1G
					2.8G	2.8G

Table 7-3: Test Scenarios

7.3.1 Latency Result

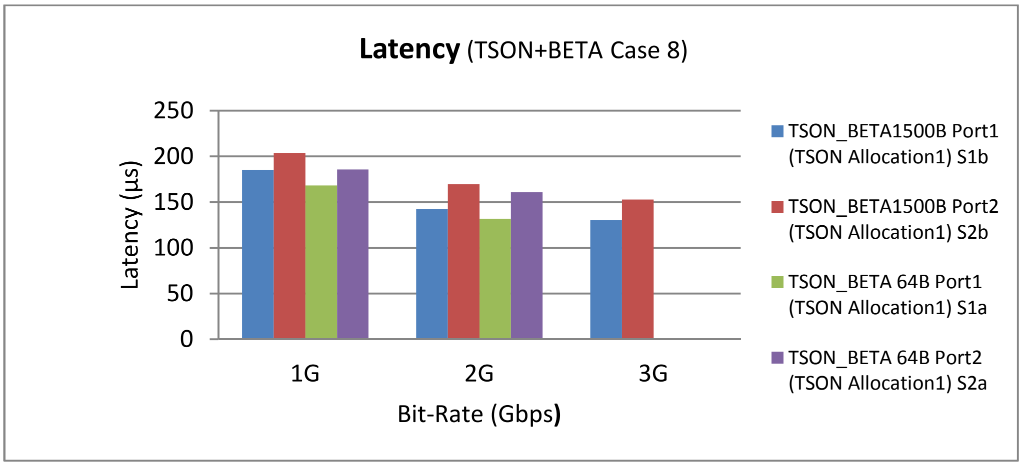


Figure 7-21: Latency results for TSON+BETA case

Case 8 Latency Analysis

The figure above shows the measured latency results for different Ethernet stream bit rate with Ethernet frame size 64B and 1500B. For the latency of TSON+BETA, with 1Gbps Ethernet frame size 1500B, for Scenario 1b, the latency is 185.259µs, and for Scenario 2b, the latency is 204.038µs. The difference is because of the 5Km fibre distance difference of the two routes.

7.3.2 Jitter Result

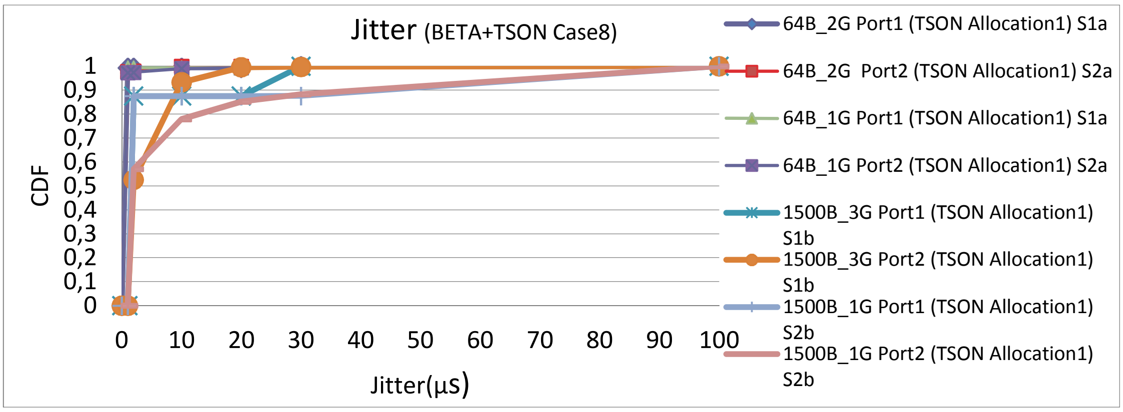


Figure 7-22: Jitter results for TSON+BETA case

Case 8 Jitter Analysis

The figures above show the measured jitter results for different Ethernet stream bit rate with Ethernet frame size 64B and 1500B. For the jitter of TSON+BETA, when the route goes through BETA first then TSON, such as measurements on Port1, the jitter result is similar to the TSON result measured in previous cases. With 1Gbps 1500B traffic, the jitter of Port1 is 87.5% frames received in 2µs.When the route goes through TSON first then BETA, such as measurements on Port2, the jitter result is similar to the TSON result measured in previous cases. The jitter of Port2 is about 57% frames received in 2µs.

7.4 Case 9: Flow aggregation and flow segregation bidirectional

The flow aggregation and flow segregation full bidirectional tests case further stresses the sub wavelength network domains by introducing two bidirectional flows operating in alternate directions via independent ports, but utilising a single interconnect port operating bidirectional flows. The characteristics of the two aggregated and segregated flows in the reverse direction are recorded for a range of transmission values.

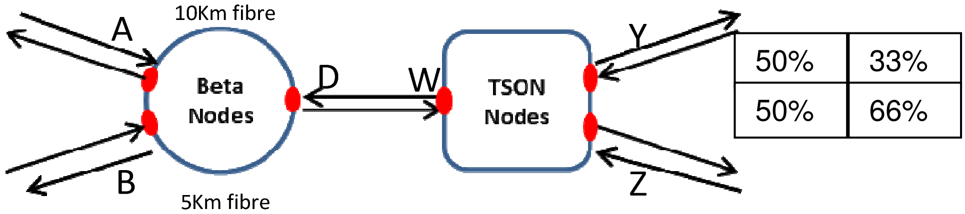


Figure 7-23: Integrated TSON and Beta case 4 aggregation and segregation partial bidirectional flow

Case 9 is combinational of integrated TSON and BETA test case 6 and case 7.

Scenario	Route BETA	Route TSON	Specification	Packet Size (byte)	Transmit rate(bps)	Receive rate(bps)
1a	D->A D->A	Y->W Z->W	TSON Time-slice Allocation 1/2/3, BETA	64	1G	1G
			TSON Time-slice Allocation 4, BETA, flow 50%/50%,33%/66%	64	1G	1G
					2.2G	2.2G
1b	D->A D->B	Y->W Z->W	TSON Time-slice Allocation 1/2/3, BETA	1500	1G	1G
			TSON Time-slice Allocation 4, BETA, flow 50%/50%,33%/66%	1500	1G	1G
					2.8G	2.8G
2a	A->D B->D	W->Y W->Z	TSON Time-slice Allocation 1/2/3, BETA	64	1G	1G
			TSON Time-slice Allocation 4, BETA, flow 50%/50%,33%/66%	64	1G	1G
					2.2G	2.2G
2b	A->D B->D	W->Y W->Z	TSON Time-slice Allocation 1/2/3, BETA	1500	1G	1G
			TSON Time-slice	1500	1G	1G

			Allocation 4, BETA, flow 50%/50%,33%/66%		2.8G	2.8G

Table 7-4: Test Scenarios

7.4.1 Latency Result

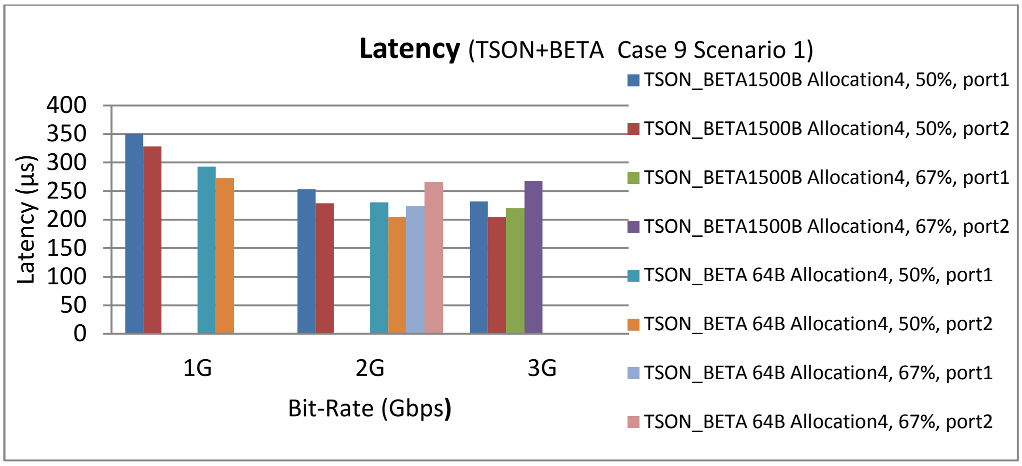


Figure 7-24: Latency results for TSON+BETA case9 scenario1

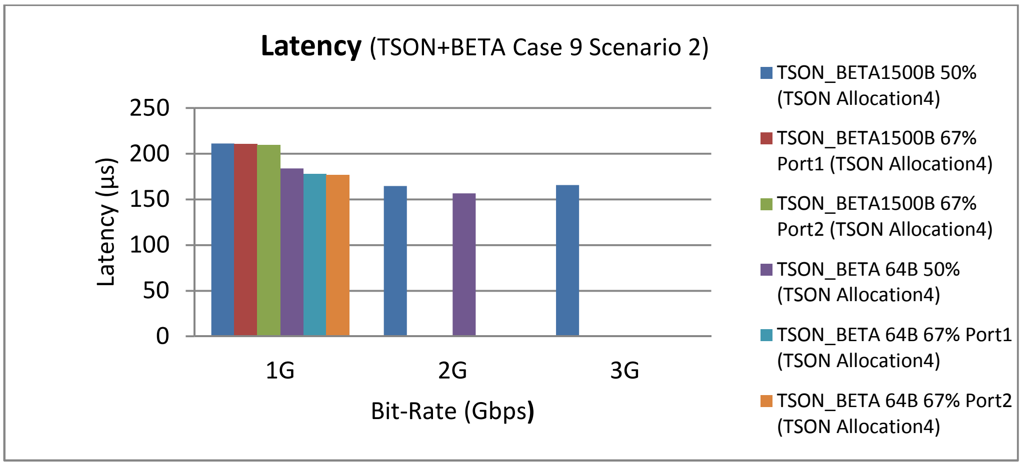


Figure 7-25: Latency results for TSON+BETA case9 scenario2

Case 9 Latency Analysis

Case 9 is a combination of case6 and case7.The latency result of scenario 1 is the same experiment of case6, and the latency result of scenario 2 is the same experiment of case7.

7.4.2 Jitter Result

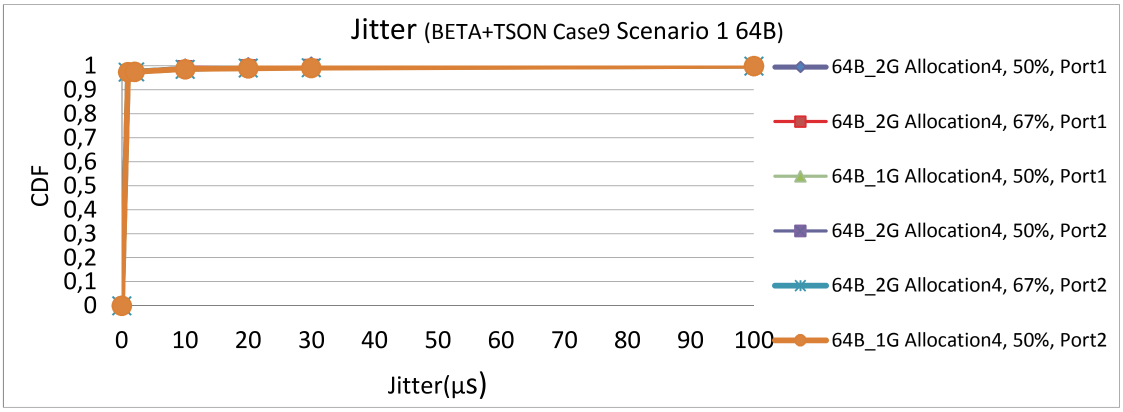


Figure 7-26: Jitter results for TSON+BETA case9 scenario1, 64B

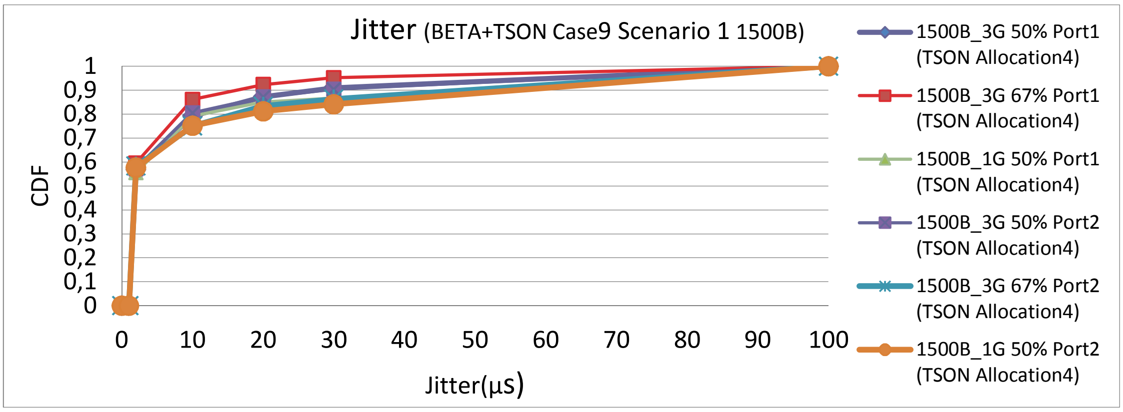


Figure 7-27: Jitter results for TSON+BETA case9 scenario1, 1500B

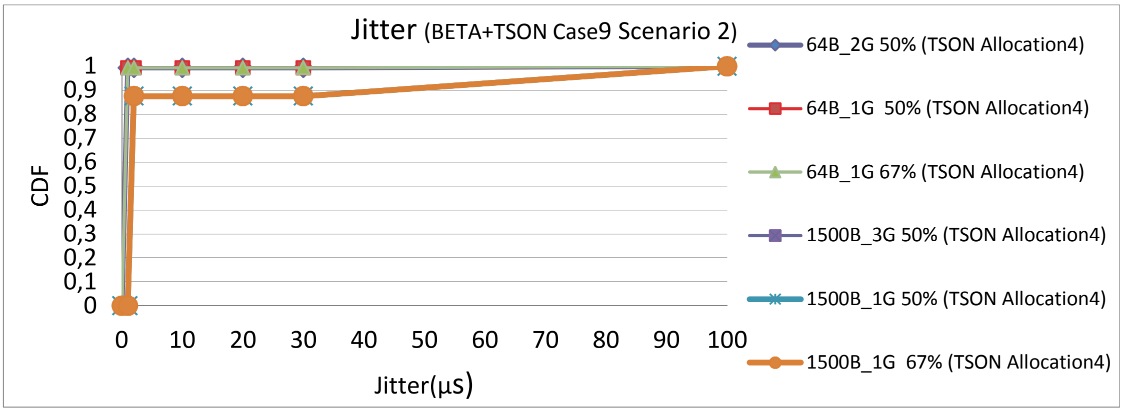


Figure 7-28: Jitter results for TSON+BETA case9 scenario2

Case 9 Jitter Analysis

Case 9 is a combination of case6 and case7.The jitter result of scenario 1 is the same experiment of case6, and the jitter result of scenario 2 is the same experiment of case7.

8 TSON/ OPST (Beta) Test Cases and analysis: Multi-Path interconnected technology domains and transport services

In the final set of test cases multi-Path interconnected technology domains and transport services are introduced. Both the final two tests cases operate flow aggregation and flow de-aggregation partial bidirectional similar to test case 4.

8.1 Case 10: Flow aggregation and flow segregation partial bidirectional

In this test case dual ports are configured to provide the interconnect between the sub wavelength network domains. The flow aggregation/segregation unidirectional flow test case considers the processing of two aggregated flows ingressing via a single port on the first sub lambda domain (Beta), and the flows separated and egressing the first domain and ingressing the second sub wavelength domain (TSON) via two ports, and then re-aggregated in the second sub wavelength domain (TSON) and egressing via a single port. The characteristics of the two aggregated, segregated and aggregated flows are recorded for a range of transmission values.

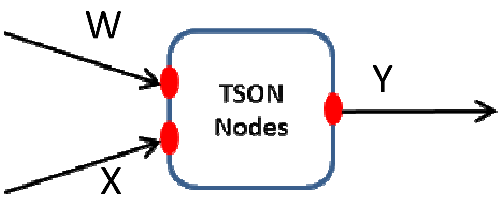


Figure 8-1: TSON case 4 aggregation unidirectional flow

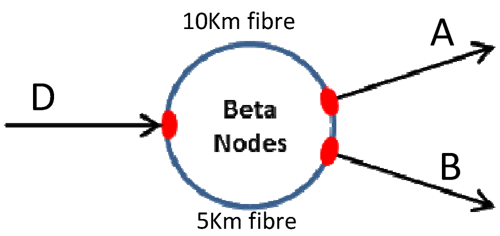


Figure 8-2: BETA case5 segregation unidirectional flow

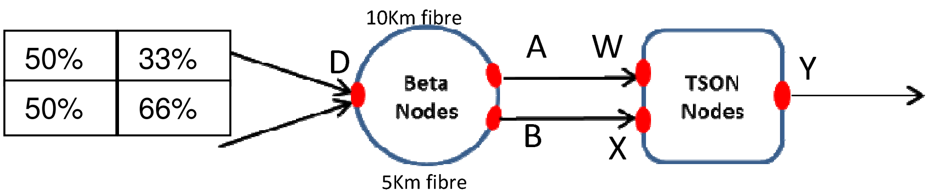


Figure 8-3: Integrated TSON and Beta case 10 aggregation and segregation partial bidirectional flow

The following tests have been conducted.

Scenario	Route BETA	Route TSON	Specification	Packet Size (byte)	Transmit rate(bps)	Receive rate(bps)
1a	D->A D->B	W->Y X->Y	TSON Time-slice Allocation 1/2/3, BETA	64	1G	1G
			TSON Time-slice Allocation 4, BETA, flow 50%/50%,33%/66%	64	1G	1G
					2.2G	2.2G
1b	D->A D->B	W->Y X->Y	TSON Time-slice Allocation 1/2/3, BETA	1500	1G	1G
			TSON Time-slice Allocation 4, BETA, flow 50%/50%,33%/66%	1500	1G	1G
					2.8G	2.8G

Table 8-1: Test Scenarios

8.1.1 Latency Result

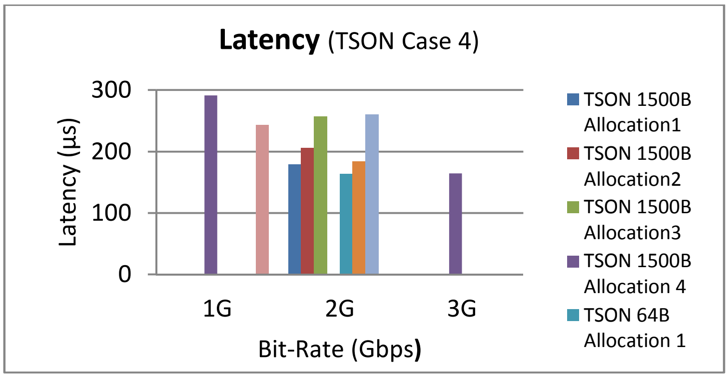


Figure 8-4: Latency results for TSON case4

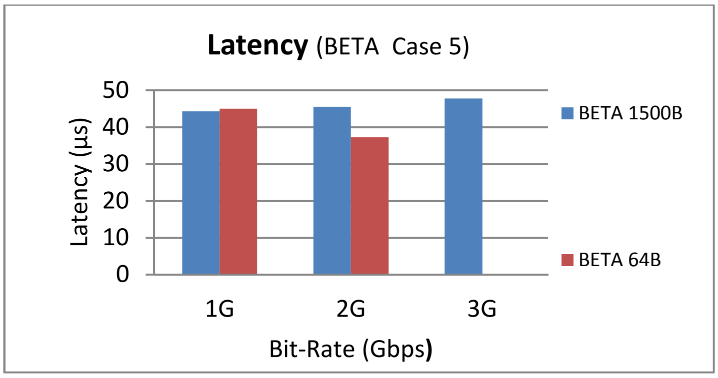


Figure 8-5: Latency results for BETA case5

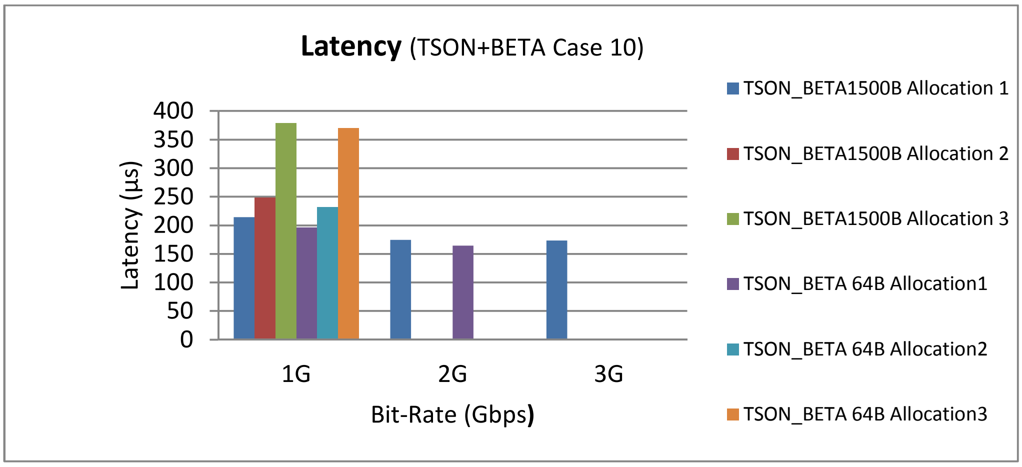


Figure 8-6: Latency results for TSON+BETA case10

Case 10 Latency Analysis

The figure above shows the measured latency results for different Ethernet stream bit rate with Ethernet frame size 64B and 1500B in different time slice allocation 1,2,3,4. The latency of BETA is same as case5, and the latency of TSON is as case4. The latency of TSON+BETA is affected by the TSON time-slice allocation. As above cases with different TSON time-slice allocations, the latency result of time-slice allocation3> time-slice alloction2> time-slice allocation1> time-slice allocation 4.

8.1.2 Jitter Result

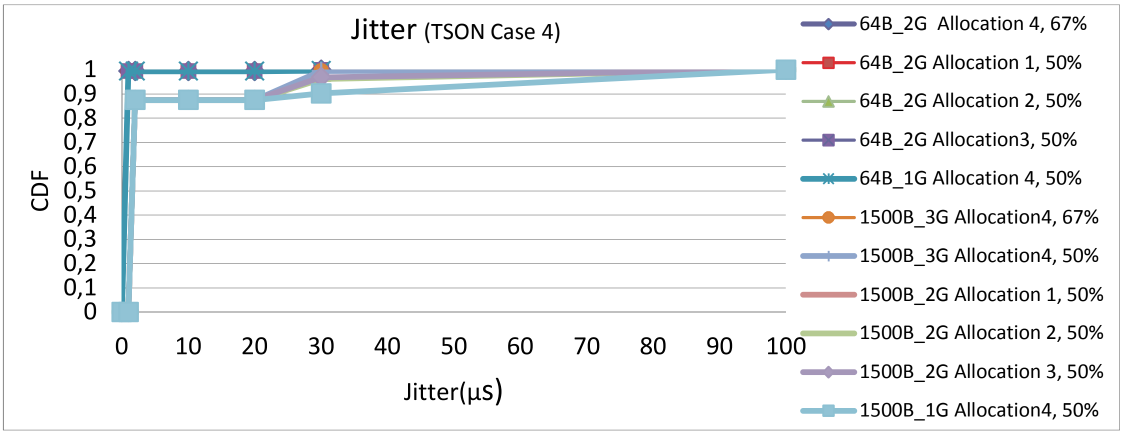


Figure 8-7: Jitter results for TSON case 4

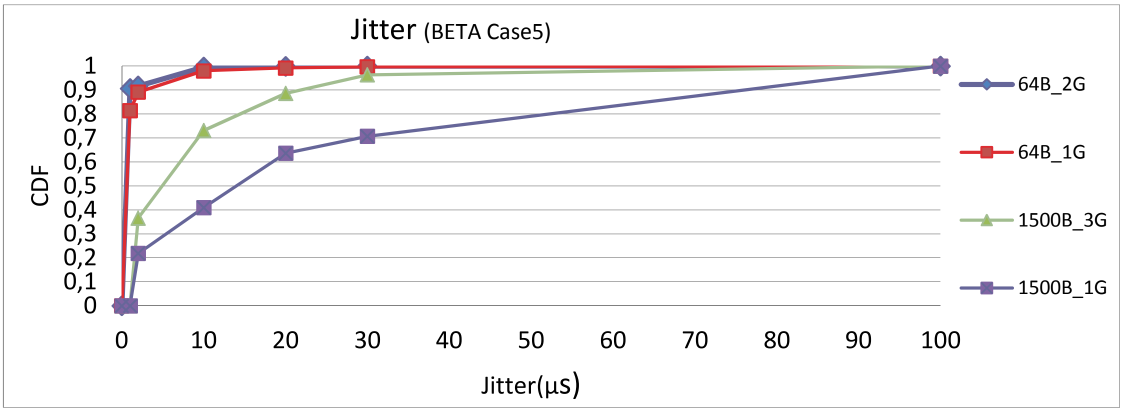


Figure 8-8: Jitter results for BETA case5

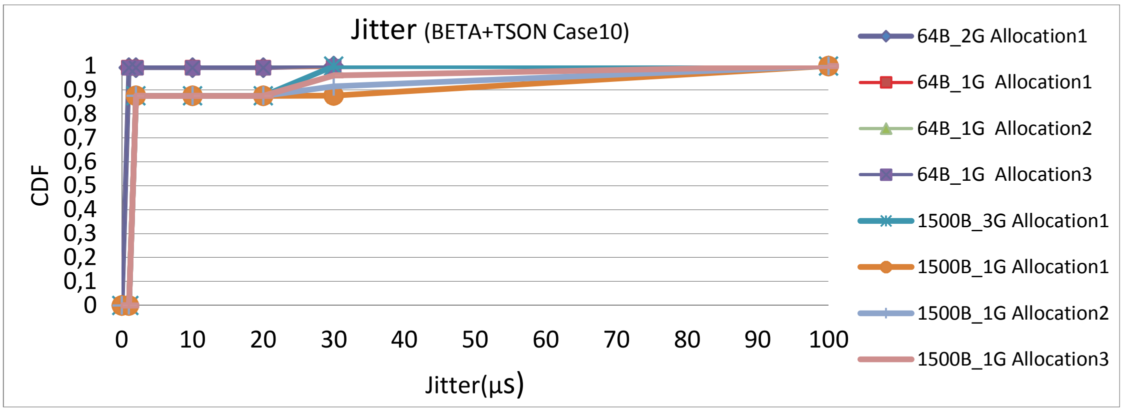


Figure 8-9: Jitter results for TSON+BETA case10

Case 10 Jitter Analysis

The figures above show the measured jitter results for different Ethernet stream bit rate with Ethernet frame size 64B and 1500B in time slice allocation 1,2,3,4. For TSON, it is the same case as case4. For OPST (Beta), it is as same as case5. For the jitter of TSON+BETA, the TSON time-slice allocations don't affect much on the jitter results.

8.2 Case 11: Flow aggregation and flow segregation partial bidirectional

In the final multi-Path interconnected technology domains and transport services test case, test case 10 is repeated but with configuration operating in the reverse direction as follows.. The flow aggregation/segregation unidirectional flow test considers the processing of two aggregated flows ingressing via a single port on the first sub lambda domain (TSON), and the flows separated and egressing the first domain and ingressing the second sub wavelength domain OPST (Beta) via two ports, and then re-aggregated in the second sub wavelength domain OPST (Beta) and egressing via a single port. The characteristics of the two aggregated, segregated and aggregated flows are recorded for a range of transmission values.

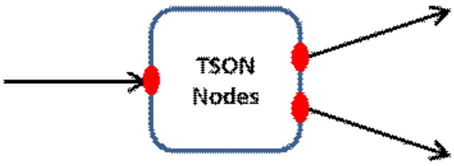


Figure 8-10: TSON case 5 segregation unidirectional flow

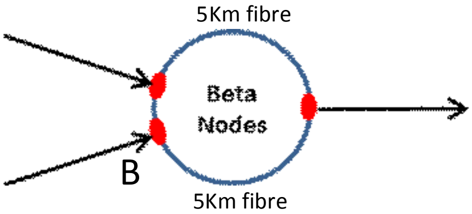


Figure 8-11: BETA case 4 aggregation unidirectional flow

Integrated TSON and Beta

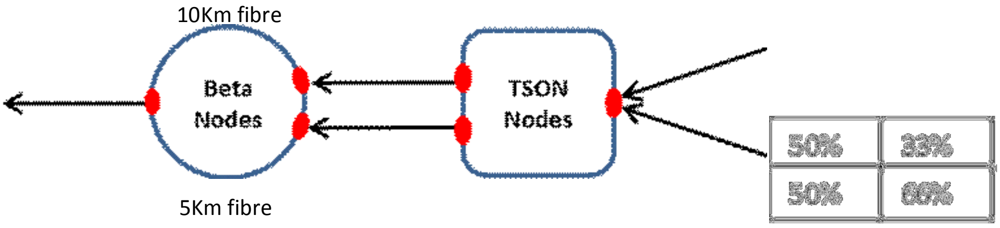


Figure 8-12: Integrated TSON and Beta case 11 aggregation and segregation partial bidirectional flow

The following tests have been conducted.

Scenario	Route BETA	Route TSON	Specification	Packet Size (byte)	Transmit rate(bps)	Receive rate(bps)
1a	A->D B->D	W->Y W->Z	TSON Time-slice Allocation 1/2/3, BETA	64	1G	1G
			TSON Time-slice Allocation 4, BETA, flow 50%/50%,33%/66%	64	1G	1G
					2.2G	2.2G
1b	A->D B->D	W->Y W->Z	TSON Time-slice Allocation 1/2/3, BETA	1500	1G	1G
			TSON Time-slice Allocation 4, BETA, flow 50%/50%,33%/66%	1500	1G	1G
					2.8G	2.8G

Table 8-2: Test Scenarios

8.2.1 Latency Result

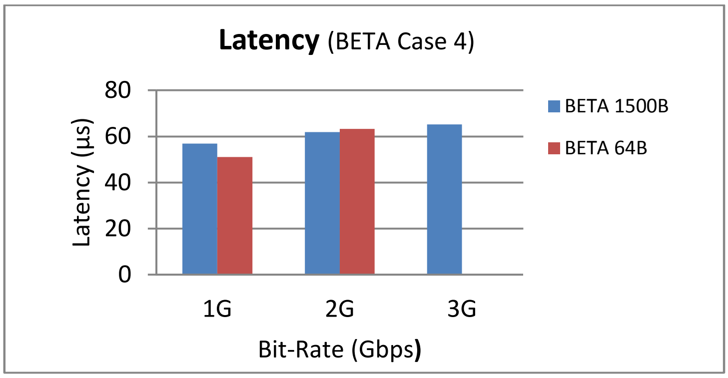


Figure 8-13: Latency results for BETA case4

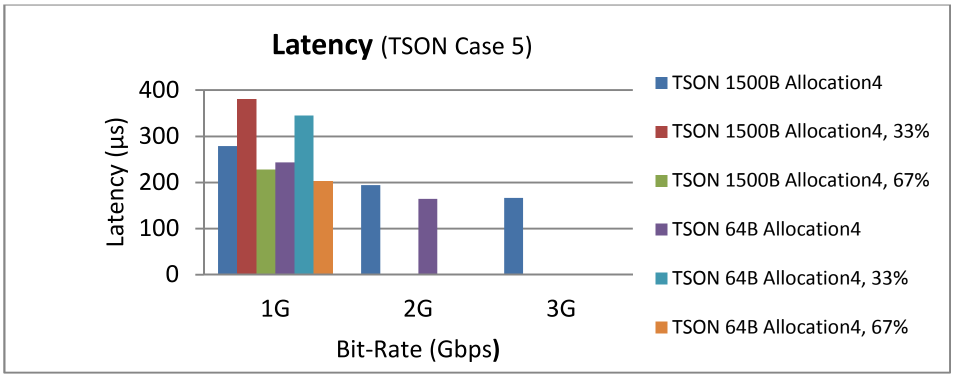


Figure 8-14: Latency results for TSON case5

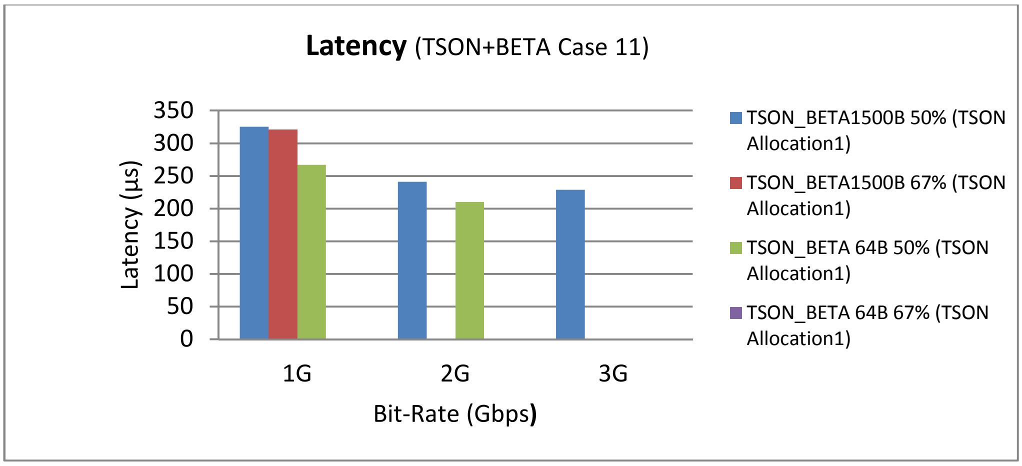


Figure 8-15: Latency results for TSON+BETA case11

Case 11 Latency Analysis

The figure above shows the measured latency results for different Ethernet stream bit rate with Ethernet frame size 64B and 1500B in different time slice allocation 1,2,3,4. The measurement also considered the condition of different percentage of the two independent aggregation flows, such as 50%/50% and 33%/67%. The latency of BETA is same as case4, and the latency of TSON is as case5. For the latency of TSON+BETA, it is not affected by the percentage of the two aggregation flows, such as for 1Gbps, Ethernet frame size 1500B, the traffic with 50%/50% and 33%/67% has similar latency (325.468µs, 320.885µs).

8.2.2 Jitter Result

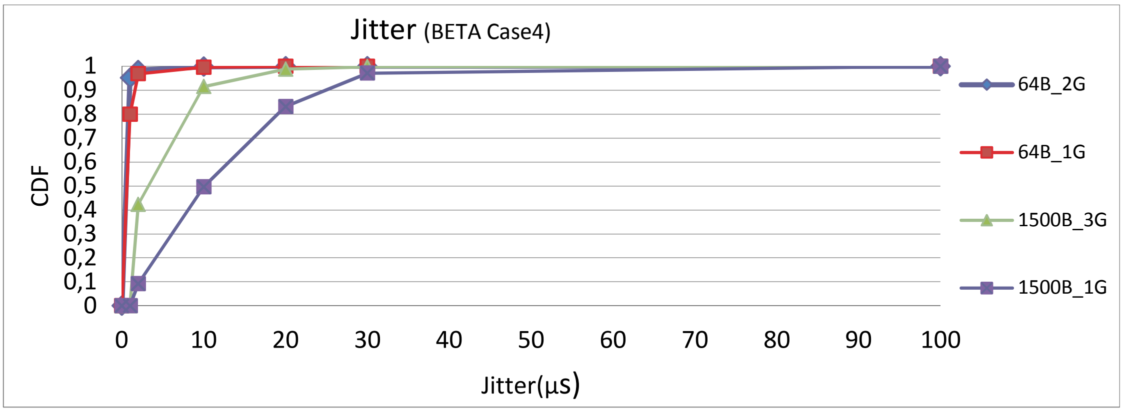


Figure 8-16: Jitter results for BETA case 4

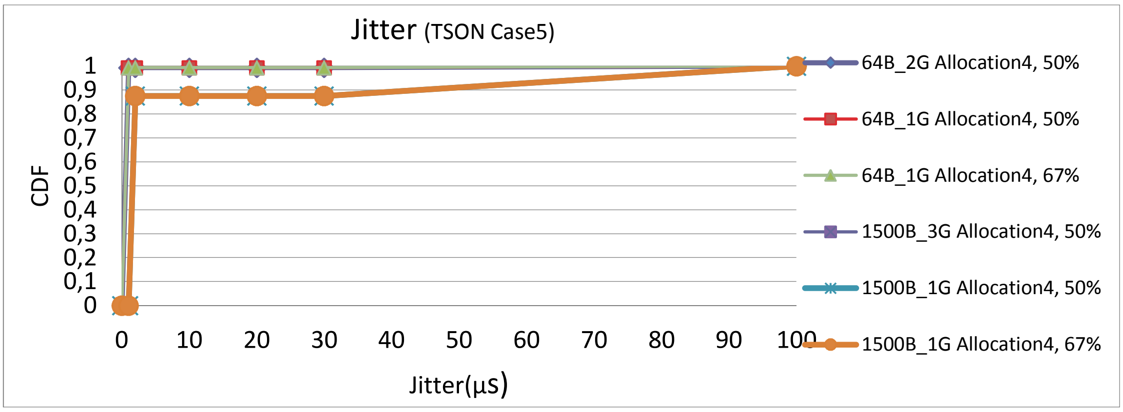


Figure 8-17: Jitter results for TSON case 5

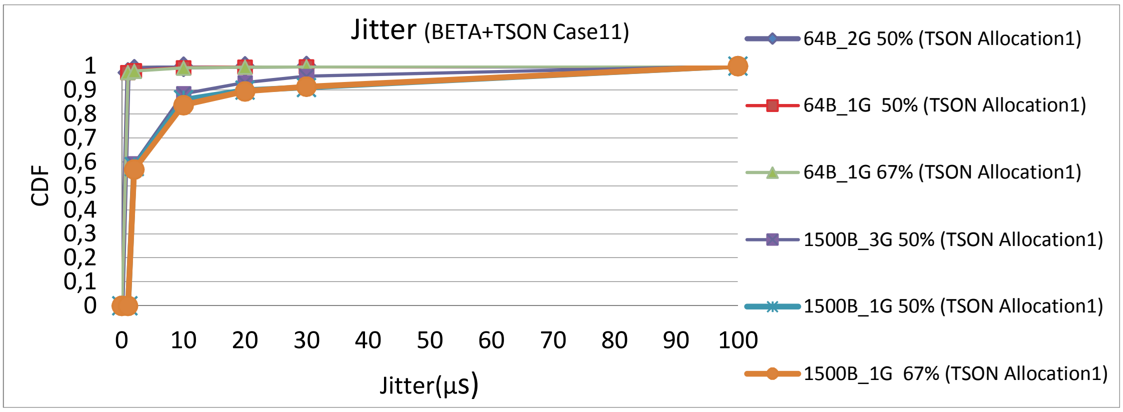


Figure 8-18: Jitter results for BETA+TSON case11

Case 11 Jitter Analysis

The figures above show the measured jitter results for different Ethernet stream bit rate with Ethernet frame size 64B and 1500B in time slice allocation1,2,3,4.The measurement also considered the condition of different percentages of the two independent aggregation flows, such as 50%/50% and 33%/67%. For TSON, it is the same case as case5. For BETA, it is as same as case4. For the jitter of TSON+BETA, the different percentages of the two independent aggregation flows don’t affect much on the jitter results.

9 Observations, Analysis and Conclusions

The MAINS team were delighted at the ease in which they were able to interconnect and interoperate the two differing sub wavelength systems at the data plane level. The Ethernet interface chosen by the team operating a MAC addressing scheme provided a proven and robust method of interconnecting disparate sub wavelength systems

Further analysis and further tests will be considered in the later stages of the project, building on the range of connectivity patterns utilised during this first stage of the introductory test plan.

All of the tests ran error free at a variety of transmission rates and packet sizes although no long term BER tests were performed in excess of 1 hour or 24 hours, so at this stage long term stability was not verified. However, the team expected to see a variation in latency and jitter characteristics between the TSON and OPST (Beta) systems, due to their implementation differences.

For the TSON network domain different time slice allocation models are expected to cause different latency and jitter characteristics. TSON allocation pattern 1 was expected to provide the best performance, whilst a pattern with a high proportion of condensed "ones" and "zeroes" was expected to cause longer latency and higher jitter on end-to-end paths. These characteristics were reflected in the recorded results. This characteristic is simply due to the aggregation and segregation procedures in the TSON system, where the Ethernet frames are buffered for transmission, and then buffered and separated to recreate the flows. The faster the packets are processed in and out of the buffers as in pattern1, improved latency and jitter results were expected, and this was observed and recorded in the experiments.

In contrast the OPST (Beta) provides a near instantaneous send and receive of data packets over a dedicated wavelength (burst mode receiver operates with a fixed wavelength) and the latency and jitter over the same length of spans varied very little. However, with two spools of 5KM fibre spans between two nodes of the OPST (Beta) network, created a variation in latency and depending on the routes the traffic had to travel and which links were used. It is evident from the results by observation that the latency of traffic travelling over 5KM paths acquires an additional 25us of latency to the original latency of 15-20 us, and furthermore passing over the two spans, which is equal to 10 KMs, the additional delay accounts for approximately 50us. Therefore the latency within an OPST (Beta) domain is very deterministic and is dependent on span (fiber) length.

It was observed in comparison to the results presented in [1], that the latency through TSON is higher (about 114us in 3G) than previously observed in deliverable [1] which recorders 60.319 us in 3G. However as mentioned above, because there are now 30 time slices in 1 frame, about 23us K-characters for recovering the clock, when an Ethernet frame enters the ingress node it needs to wait for at least 23us before generating the K-characters. When the burst enters the egress node, it needs another 23us to receive the K-characters. So a total of 46us additional latency is introduced. But when using 91 time slices in 1 frame, only 1us K-characters for ingress node/egress node I introduced.

In all cases the end-to-end latency never exceeds 400us and the jitter 40us an extremely good result. Excluding all propagation delays roughly 85% (~115 us) of the latency is due to TSON technology and 15% (~20 us) due to Beta technology. The test cases that consider asymmetric traffic flows on P2P, P2MP, MP2MP scenarios (33%- 66% balance between flows) also have the equivalent latency. The lower the load (e.g. 33%) the higher the latency.

On the other hand 85% (~ 30 us) of the jitter is due to Beta technology and 15% (~ 5 us) due to TSON technology. In particular, the Beta jitter performance is dependent on the Ethernet frame length and as such the longer the incoming Ethernet frames and higher the jitter. TSON due to its time-sliced allocation can inherently better restrict the jitter.

10 References

ITU-T www.itu.int

IEEE www.ieee.org

IETF www.ietf.org

[1] MAINS deliverable D4.1

[2] Amaya , et al, "Field Trial of a 1.5 Tb/s Adaptive and Grid-less OXC Supporting Elastic 1000-Fold Bandwidth GranularityECOC 2011)

11 Acronyms

AN	Access Node
CN	Concentration Node
CO	Central Office
CAPEX	Capital Expenditures
CPE	Customer Premises Equipment
CS	Concentration Switch
DWDM	Dense Wavelength Division Multiplexing
E2E	End to End
EN	Edge Node
E-NNI	External Network-to-Network Interface
FTTH	Fiber To The Home
GbE	Gigabit Ethernet
GMPLS	Generalized Multi-Protocol Label Switching
L2VPN	Level 2 Virtual Private Network
NCS	Network Centric Services
OAM	Organization Administration and Management
OBS	Optical Burst Switching
OBST	Optical Burst Switching Technology
OLT	Optical Line Termination
OPEX	Operational Expenditures
OPST	Optical Packet Switching Technology
PCE	Path Computation Element
QoE	Quality of Experience
QoR	Quality of Resilience
QoS	Quality of Service
SC	Sub-wavelength Concentrator
SOAP	Simple Object Access Protocol
TE	Traffic Engineering
TMF	Telemanagement Forum
TNA	Transport Network Address
VM	Virtual Machine
VoD	Video on Demand

WADL	Web Application Description Language
WSON	Wavelength Switched Optical Network