C22. Avionics Full-duplex switched Ethernet (AFDX): Modeling and Simulation

Nour El-Din Safwat, Abdelhalim Zekry, Mohamed Abouelatta
Egypt Air Company, Cairo, Egypt, eng.nours@gmail.com
Faculty of Engineering, Ain Shams University, Cairo, Egypt, aaazekry@hotmail.com
Faculty of Engineering, Ain Shams University, Cairo, Egypt, m.abouelatta@eng.asu.edu.eg

ABSTRACT

This paper develops a comprehensive simulation model for Avionics Full-duplex switched Ethernet (AFDX) network based on OPNET platform. Accordingly, the performance of AFDX networks is analyzed. The effect of frame size, switching delay and changing frames transmission order on the AFDX network performance are investigated. It is found that using faster switches and small frame size reduces the fixed part of the end to end delay. Also, it affects the variable part of end to end delay due to changing of delay in switches' buffers and frames transmission order.

Keywords: AFDX, Endsystem, simulation, end-to-end delay

I. INTRODUCTION

Aircraft Data Networks (ADN)'s has experienced a great development since the start of Avionics as early as the 18th century [1]. The increase of the complexity of Avionics systems made the older network protocols as ARINC429 [2] [3] and ARINC629 [4] don't fulfil the new (ADN)'s requirements.

AFDX [5], the new aircraft data network, is based on conventional Ethernet as it has well established protocols. But although Ethernet supports high speed and low cost parts, it possesses Probabilistic protocols as CSMA/CD and non-redundant architecture which don't meet the main avionics network requirements of determinism and reliability. AFDX modifies the conventional Ethernet by adding new mechanisms to it such that it becomes applicable to Avionics Networks.

AFDX [5] ensures a deterministic behaviour through traffic control. Traffic control is achieved by guaranteeing the bandwidth of each logical communication channel, called a Virtual Link (VL), thereby limiting the jitter and transmit latency. The virtual link parameters [6] are:

- Bandwidth Allocation Gap (BAG), a timeslot confining the VL's bandwidth by defining the minimum gap time between two consecutive frames. The BAG value must be in the range 1 - 128ms and must be a power of 2 according to the standards [7].
- L_{max}, the largest Ethernet frame, in bytes, that can be transmitted on the virtual link.

To satisfy the requirement of reliability for Avionics, the AFDX network has a duplicated network [7], i.e. a redundant network say network A and network B, transmitting exactly the same data. The purpose of the redundant network is to mitigate the consequences of potential network failures caused by damaged cables and connectors or switches. Since ultimately only one valid data stream is required by the End System (ES) application, the ES implements Integrity Checking (IC) and Redundancy Management (RM) [7] to ensure data integrity and that only one data stream is forwarded to the upper protocol layers and from there to the application.

AFDX network performance analysis is very important to estimate and determine whether AFDX can meet the real-time requirements of the Avionics networks. In addition it is required to better understand the real behaviour of the AFDX network. Therefore a simulation model for AFDX network may be very helpful. There have been appreciable research efforts to build a simulation model for AFDX. One model was built on the network simulator QNAP [8] A second one was built on the simulator NS2 [9]. Alternatively, there was a model built on the network simulator OPNET [10], but it misses. Integrity check and redundancy check in the receiving part of ES.

In this paper a comprehensive simulation model for AFDX is built. The AFDX performance is consequently investigated. The rest of the paper is organized as the follows: the AFDX network modelling is presented in section two. AFDX network simulation and network performance analysis is introduced in section three. Final section is devoted to the conclusion.

ISBN 978-1-4799-7723-9/15/$31.00 © 2015 IEEE

II. AFDX Network modelling

In this section a comprehensive model for the AFDX network will be introduced. The AFDX network consists of two main subsystems: the End System (ES) and Switch.

A. End System (ES)

It provides an "interface" between the Avionics Subsystems and the AFDX Interconnect. The End System guarantees a secure and reliable data interchange with other Avionics Subsystems. It is comprises two parts: the transmitting ES and the receiving ES.

1) *Transmitting ES*

Figure 1 shows the functional block diagram of the transmitting ES. It consists of two main functions: The traffic shaping function and the redundancy function.

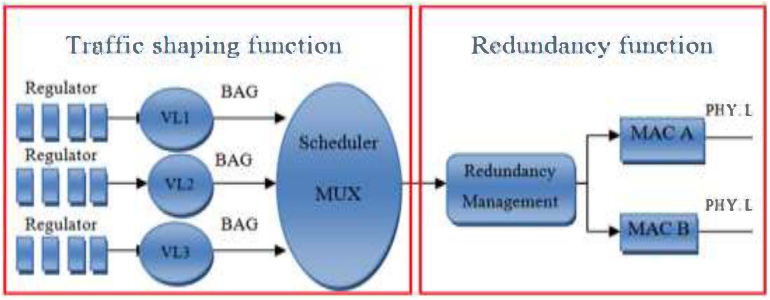

Fig. 1: Model of transmitting ES.

***Traffic shaping function*:**

The aim of the traffic shaping function is to limit the instantaneous frame rate of the Virtual Links by spacing the frames using [5][6] regulators and schedulers or multiplexer.

The Regulator: it is responsible for controlling the bandwidth given to a Virtual Link. It performs according to the rule that the minimum time interval between the first bits of two consecutive frames on the same VL must not be smaller than BAG. Figure 2 shows frames flow at the input and the output of regulator.

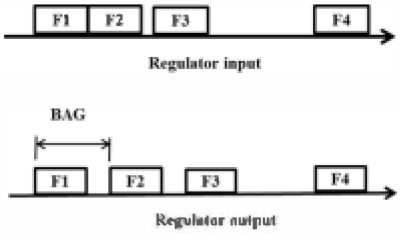

Fig. 2: Flow at the input and the output of the regulator

***The Scheduler (Multiplexer)*:**itmultiplexes frames received from Regulators. Frames arriving at the input of the scheduler at the same time will experience queuing delay (jitter).

Redundancy function [7][6]:
End Systems communicate over independent and redundant network such that data flows are protected against the failure of any network component such as a link or a switch. This is done using:
Redundancy Management (RM): The redundancy scheme is operated on a per Virtual Link basis. Its functions are:

- Adding SN to frames received from scheduler. The SN is a value in the range (0 – 255) and is handled separately for each VL on each of network A and B. SN is incremented by one for each consecutive frame, whether fragmented or not, on the same VL. With SN = 255 in the last transmitted frame, the SN is wrapped around to 1 in the following frame. Upon a reset or start-up of the transmitting ES, the SN is set to 0 in the first transmitted frame.

- After adding SN, the frame is duplicated and sent to MAC interface.

2) *Receiving ES*

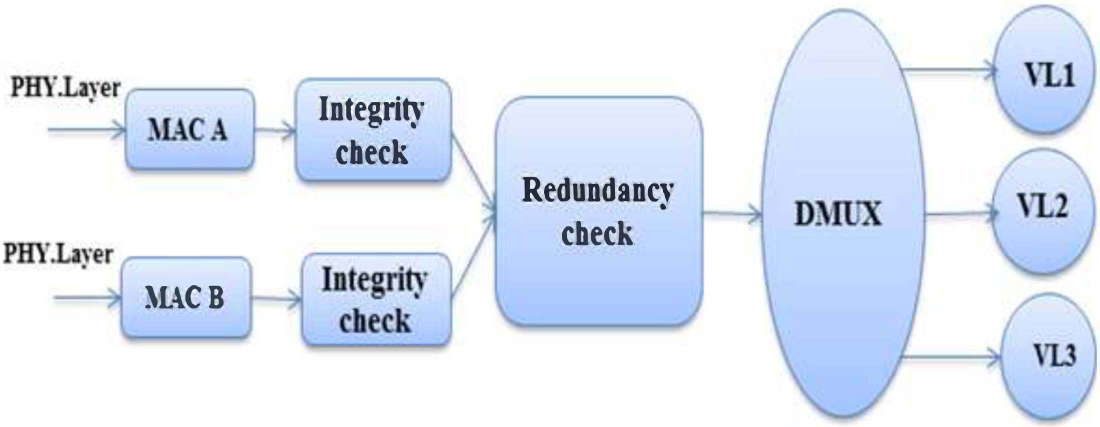

Fig. 3: Model of receiving ES.

The receiving ES performs the inverse processes of the transmitting ES. As shown in Figure 3, the receiving ES consists of integrity check, redundancy check, demultiplexer and MAC interface.

Integrity Check(IC)[7]: The function of IC is to detect and eliminate invalid frames. SN is the basis for the IC algorithm.

It is used to determine if frames have been lost or whether a bubbling switch is causing the same frame with the same SN to be transmitted over and over again.

Redundancy check[7]: It evaluates the two frame sequences delivered by the IC, discard possible duplicate frames, and forward only one copy of each frame to demultiplexer. In the case where the Redundancy Management is disabled, both frame sequences are forwarded directly from the IC to de-multiplexer

Demultiplexer**:** it delivers frames according to Virtual Links to upper layers.

MAC interface**:** In transmitting, it adds addresses to the frame and transmits it to switch.

While in receiving, it checks the destination address in the frame. If it doesn't match the ES address, it destroys the frame. If there is another frame in processing, it inserts the frame in an in/out queue.

B. AFDX switch[7],[6]

As shown in Figure 4, switch model consists of:

- I/O queuing ports.
- The switching function which implements a filtering and policing function to ensure that only valid incoming frames are forwarded to the right physical ports.
- A configuration static which is responsible for setup of the switching function.

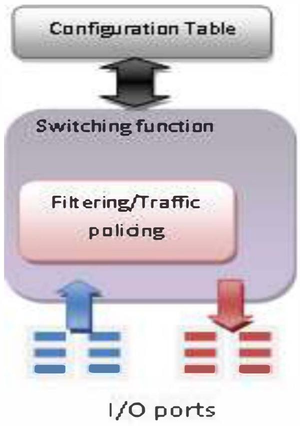

Fig. 4: Switch model.

C. AFDX model implementation

In this section, a comprehensive AFDX Model that contains also the previously missing integrity check and redundancy check in the receiving part of ES is developed using the network simulator OPNET.

II.C.1 Model specifications

The AFDX Model has the following specifications is built in OPNET.

- It is based on OPNET Ethernet model.
- It covers up to data link layer in AFDX protocol.
- The model consists of ES model and AFDX switch

II.C.2 Es model

Figure 5 shows the OPNET ES model in which is based on the functional descriptionin section 2.1. It consists of:

- ***Data generator***

It is a process module .it represents sub end systems which generates Frames. One data generator identifies one VL. In this model, one ES consists of three sub end systems. Data generation is controlled by:

– traffic generation parameter which control start time of the frame

– Packet generation parameters which determine packet size and time of packet generation.

Each data generator module generates data for one VL.

- ***Regulator***

It is a queue module. It receives frames from its data generator and retransmits them by time spacing not less than the BAG time of the corresponding VL. This done by setting the service time of the frame based on BAG.

- ***Scheduler***

It is a multiplexer which receives frames from Regulators. It processes frames based on FIFO policy. The service time depends on frames size.

- ***Redundancy management***

This module inserts the SN to the AFDX frame then it duplicates the frame. Finally, it forwards the AFDX frame and its duplicated version to MAC interface.

- ***Integrity check***

It receives frame and extracts the SN. Then, it compares it with the previous one of the same VL to check that:

– SN = Previous SN + 1

– SN = Previous SN + 2

All frames not complying with these criteria are discarded.

- ***Redundancy check***

It receives the two redundant frames. It forwards only one valid frame from the two redundant frames. There are many algorisms introduced in[11] to select the valid frame. The most of them are based on *"first-valid-wins"* policy. The algorism used in this model just listens on both networks and synchronize to the faster one. This approach is very robust against critical sequence number resets [11].

- ***MAC interface***

It supports full duplex transmission. When it receives frame, it checks if the frame coming from the higher layer or the physical layer. If it comes from higher layer, it forwards the frame to switch after encapsulation and adding addresses. Else, if it comes from physical layer, it receives the packet and extracts the destination address and compares it with the ES address. In case of matching it accepts the frame, if not it destroys it.

II.C.3 Switch model

Figure 6 shows the OPNET switch model which is based on the functional description in sec 2.2. It consists of the following modules:

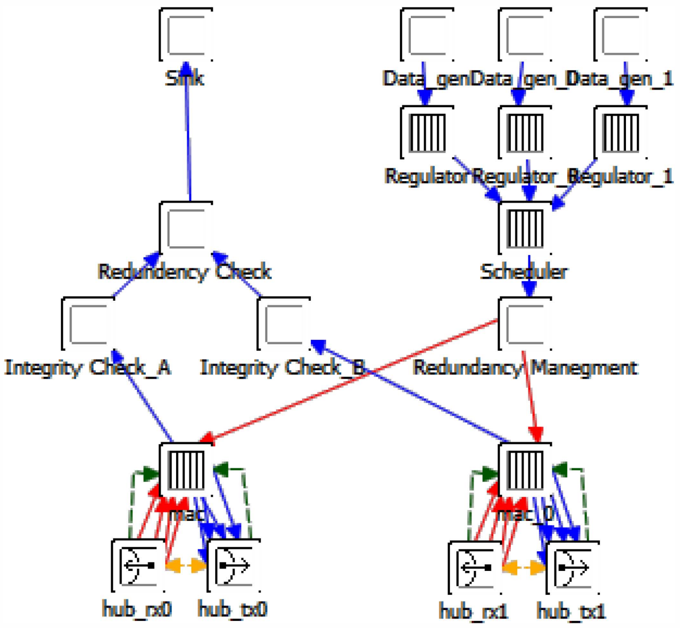

Fig. 5: AFDX End System OPNET node model.

- ***MAC module***

It contains the input and the output queues. It implements a filtering and policing function to ensure that only valid incoming frames are forwarded to the right physical ports.

- ***Switch module***

Switch module is responsible for routing frames to output ports based on static configuration table. The service time for frames is configurable because it is a technological parameter.

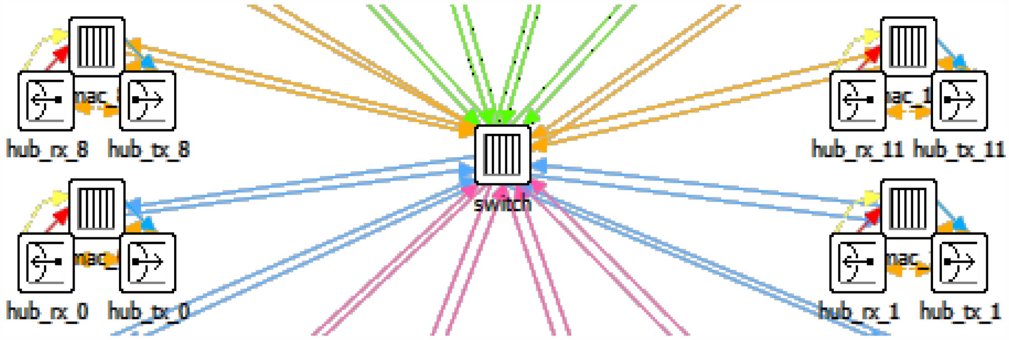

Fig. 6: Switch OPNET node model.

III. AFDX NETWORK SIMULATION

A. Simulation

1) AFDX network case study

The case study is a part of flight management system (FMS) which represent realistic AFDX network traffic [12].This system manages part of the displays in the cockpit. It provides some information on a waypoint requested by the pilot and periodically refreshes dynamic data related to this waypoint, e.g. distance and estimated time of arrival.

2) Network topology and configuration

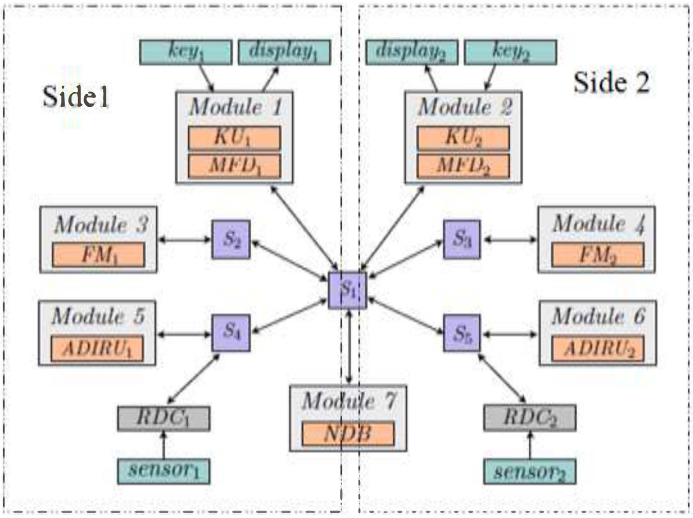

Fig. 7: Flight management system (FMS) network [12].

Figure 7 shows the proposed AFDX network which consists of 7 modules representing 7 End systems. Module 1 and Module 2 each consists of 2 sub-end system. Sensor 1and 2 are interfaced with AFDX network through Remote Data Concentrator RDC. In the cockpit, the pilot and the co-pilot use a personal keyboard and display to interact with the FMS. The FMS uses a redundant implementation of its functions which are segregated on each side of the plane, e.g. side 1 and side 2 as in Figure 7.

At any time, the pilot can request some information on a given waypoint. The Keyboard and Cursor Control Unit KU1, controls the physical device used by the pilot to enter his requests. When KU1 receives a request, it broadcasts it to FM1 and FM2. The Flight Managers FMs, manage the flight plan, i.e. the trajectory between successive way points. When there is a request, both instruct the Navigation Database

NDB, to retrieve the static information on the waypoint such as the latitude and the longitude. NDB separately answers each FM by sending a message containing the requested data. When receiving this message, each FM computes two complementary dynamic data: the distance to the way point, and the estimated time of arrival. This information is periodically sent to the respective Multi-Functional Display MFDs of FM, which periodically elaborates the pages to be displayed on the screens. To compute these data, the FMs use the position and the speed of the aircraft which are periodically delivered by the Air Data Inertial Reference Unit ADIRUs.The ADIRUs determine the speed and position of the aircraft thanks to data provided by sensors. The sensors Interconnection with the AFDX network is managed by Remote Data Concentrators RDCs.

Setup on OPNET is shown in Figure 8. The network interconnecting between End systems is done through 5 switches. Table1 contains the VLs configurations parameters for each virtual link. Each Virtual Link is defined by a source, a destination or destinations, a path andBandwidth Allocation Gap (BAG).

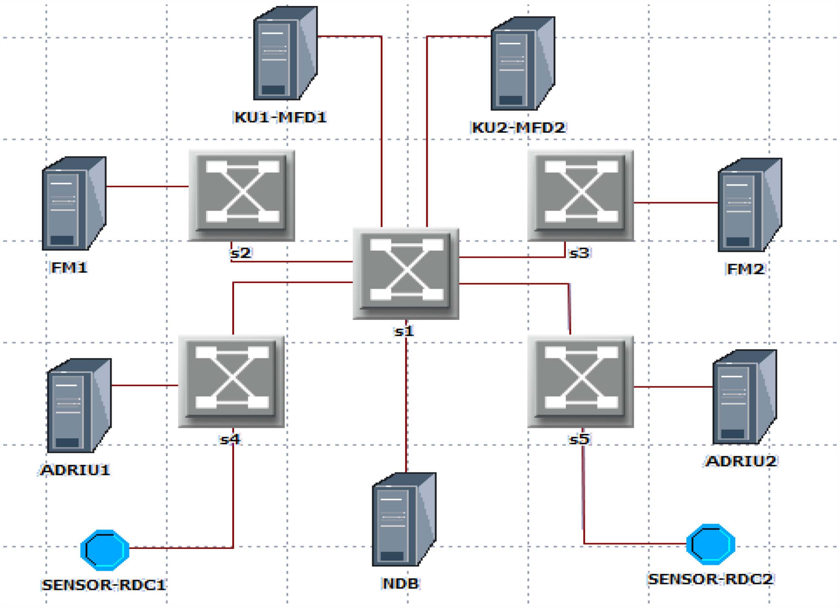

Fig. 8: Network setup on OPNET.

Table1: Virtual links configuration parameters

Virtual Links	Source	Destination(s)	BAG(ms)	path(s)
VL1	KU1	FM1,FM2	32	S1,S2 , S1,S3
VL2	KU2	FM1,FM2	32	S1,S2 , S1,S3
VL3	FM1	MFD1	8	S2,S1
VL4	FM1	NDB	16	S2,S1
VL5	FM2	MFD2	8	S3,S1
VL6	FM2	NDB	16	S3,S1
VL7	NDB	FM1	64	S1,S2
VL8	NDB	FM2	64	S1,S3
VL9	RDC1	ADRIU1	32	S4
VL10	RDC2	ADRIU2	32	S5
VL11	ADRIU1	FM1,FM2	32	S4,S1,S2 , S4,S1,S3
VL12	ADRIU2	FM1,FM2	32	S5,S1,S3 , S5,S1,S2

B. simulation Results

The OPNET simulation results of end-to-end delay of VL1 to VL11 are displayed in Figure 9 as a function of the simulation time. The change in end to end delay of each VL is due to changing the start time to send frames of some VLs in the simulation configuration. This is done to cover wide range of scenarios because there are modules don't send frames periodically. For example KU1 doesn't send frame until pilot request. Table2 shows our simulated end-to-end minimum and maximum delay compared to the end –to-end delay given in reference [12]. By inspecting the values given in the table, one sees that our simulation results are laying within the minimum and maximum computed end –to-end delay values in [12] which is acceptable for simulation approach.

The end to end delay distribution of VL is of interest to characterize a network. Therefore, Figure 10 shows a typical simulated delay distribution of VL1. It is clear from the Figure that the delay distribution is close to the calculated BCTT (Best Case Transmission Time) and far from the calculated WCTT (West Case Transmission Time). This is mainly due to the fact that the AFDX network is lightly loaded. Thus, the probability that several frames reach the same output port at the same time is very low.

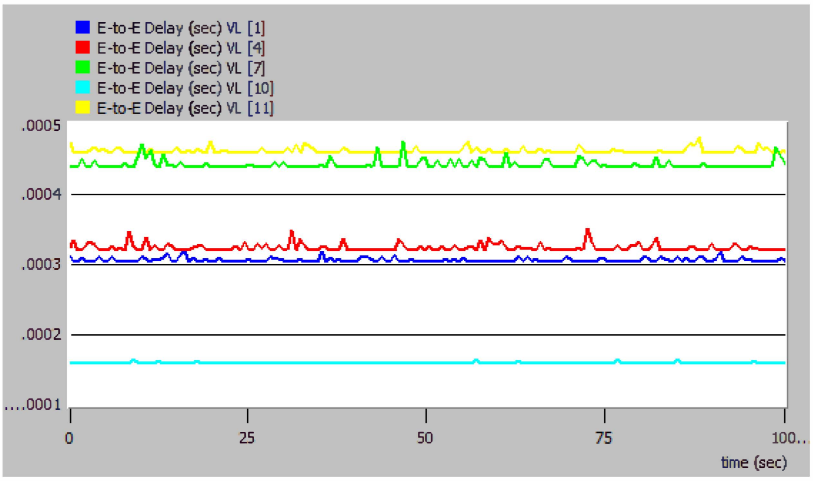

Fig. 9: Virtual links End-to-End delays.

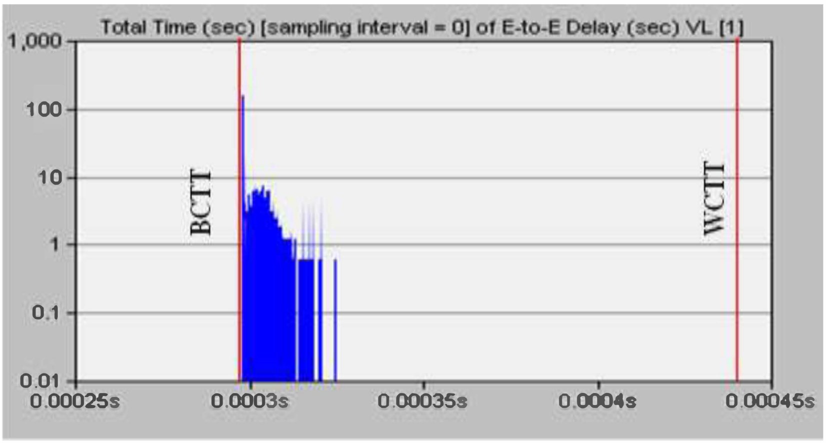

Fig. 10: End to end delay distribution of VL1.

Table2: Simulated Vs. calculated End-to-End delay

VL	Source	Destination	BCTT(µs)	WCTT(µs)	Simulation(µs)	
					Min	Max
1	KU1	FM1	298	444	298	328
4	FM1	NDB	310	450	310	341
7	NDB	FM2	400	508	400	428
10	RDC2	ADRIU2	150	156	152	155
11	ADRIU1	FM1	452	584	452	473

C. End-to-End delay Analysis

It is useful to analyze the end-to-end delay in its components such that one can control the delay time budget. The end to end delay $D(F_v, P_x)$ of a frame Fvtransmitted on a path $P_x$is the sum of the transmission delay, the switching delay and the buffering delay. It can be expressed by [13]:

$D(F_v, P_x) \;=\; LD(F_v, P_x) \;+\; SD(F_v, P_x) \;+\; WD(F_v, P_x)$ [13](1)

Where:

$LD(F_v, P_x)$ is the transmission delay over the links. It is given by:

$LD(F_v, P_x) = nbl(P_x\;) \times (t_{byte} \;\times\; s(F_v\;)),$[13] (2)

With t_{byte} is the transmission time of one byte. For AFDX at speed of 100 Mbit/sec t_{byte} =.08 µs, s(F_v) is F_v length, and

$nbl(P_x\;)$is the number of links in P_x.

$SD(F_v, P_x)$ is the delay in switches between input and output ports. It can be written as:

$SD(F_v, P_x) = nbs(P_x\;) \;\times\; t_d$[13](3)

With *nbs(*P_x*)* is the number of switches in P_x,t_d is the delay in a switch from an input port to an output port. It is a technology parameter. At this case study t_d=140 µs [12]

$WD(F_v, P_x)$ is the delay in switches and end system output buffers. It can be put in the form:

$WD(F_v, P_x) \;=\; WD(F_v, P_x, e_v) \;\; + \sum_{sk\,\epsilon\,\Psi_{p_x}} WD(F_v, P_x, sk)$[13](4)

With e_v is F_v source end system,Ψ_{p_x} is the set of switches in P_x, $WD(F_v, P_x, e_v)$ is the delay in e_v output buffer, and$WD(F_v, P_x, sk)$ is the delay in sk output port buffer.

Consequently, $WD(F_v, P_x)$ can be divided into a fixed part comprising $LD(F_v, P_x) \;+\; SD(F_v, P_x)$ and a variable part$WD(F_v, P_x)$. The fixed part can be statically computed since it depends on the path P_x, the length of the frame F_v and the bandwidth of the links. The variable part depends on dynamic characteristics such as the sequence of frames which are emitted by each VL and the offsets between the different VLs, i.e. the emission instant of the first frame of each VL.

The previous formula shows that the parameters affect end to end delay are frame size, switching delay and changing frames transmission order. Table 3 shows the results of end to end delay analysis for the fixed and variable parts of VL11, 7, 9 which represent max and min delays of VLs in the case study network. It is noticed from the table that:

- The switching delay SD(F_v, P_x) is the dominate part in the end to end delay of VLs. VL11 is the max end to end delay as it is the longest path (ADRIU2 - S5- S1-S2- FM1).
- The Transmission delay $LD(F_v, P_x)$ is high in VL7 having the path (NDB - S1-S2-FM1) as it transmits large frame size.
- VL9 has the min end to end delay time as it is the shortest path (RDC1 - S4- ADRIU1) and smallest frame size.

Table 3: Fixed and variable parts of end to end delay of VL11, 7, 9

VL no.	*path*	*E-to-E delay (Max)* (µs)	*Fixed delay*		*Variable (Max)*
			$LD(F_v, P_x)$	$SD(F_v, P_x)$	$WD(F_v, P_x)$
11	ADRIU2 - S5- S1-S2- FM1	492	32	420	38
7	NDB - S1-S2-FM1	477	120	280	77

9	RDC1 - S4- ADRIU1	154	10.24	140	3.76

In order to study the effect of the switching time on the end to end delay time a network simulation is done again using faster switches with delay around 16 µs [13]. Table 4 shows the fixed and delay parts of the paths after changing the switch delay. It is noticed that:

- The end to end delays in all VLs are decreased compared the previous case with longer switch delay. VL7 became the max end to end delay instead of VL11 as the transmission delay becomes higher than switching delay.
- Variable delay part is changed in VLs. In VL 11 & 7, Variable delay decreased because using faster switches minimize the delay in the switches buffers. In VL 9, variable delay increase due to the change of the order of frames arrival time at switch 4 of the other VLs.

Table4: End to end delay fixed and variable parts of VL11, 7, 9 at switch delay =16 µs

VL no.	*path*	*E-to-E delay (Max)* (*µs*)	*Fixed delay*		*Variable (Max)*
			$LD(F_v, P_x)$	$SD(F_v, P_x)$	$WD(F_v, P_x)$
11	ADRIU2 - S5- S1-S2- FM1	92	32	48	12
7	NDB - S1-S2-FM1	194	120	32	42
9	RDC1 - S4- ADRIU1	33	10.24	16	6.76

IV. Conclusion

A proposed comprehensive Simulation model for Avionics Full-duplex switched Ethernet AFDX network using OPNET platform is developed. Performance analysis for (AFDX) network is carried out. The end to end delay distribution is estimated. The effect of frame size and switching delay is studied. It is found that using faster switches and small frame size reduces the fixed part of the end to end delay. Also, it affects the variable part of end to end delay due to changing of delay in switches buffers and frames transmission order.

References

[1] Safwat, N. E. D., El-Dakroury, M. A., & Zekry, A. (2014). The Evolution of Aircraft Data Networks. International Journal of Computer Applications, 94.

[2] ARINC Specification 429, Part 3-18. Annapolis, Maryland: Aeronautical Radio, Inc. 2001-10.

[3] ARINC Specification 429, Part 1-17. Annapolis, Maryland: Aeronautical Radio, Inc. 2004-05-17.

[4] Gabillon, A., & Gallon, L. (2006). Availability of ARINC 629 Avionic Data Bus.Journal of Networks, 1(6), 1-9.

[5] Enginering, C. Inc. AFDX/ARINC 664 Tutorial (1500—049). 2005—05. http://www. condoreng.com.

[6] AFDX® / ARINC 664 Tutorial, ID: 700008_TUT-AFDX-EN_1000, TechSAT GmbH, Poing, Release Date 29/08/2008.

[7] Part, A. D. N. (2005). 7: Avionics Full Duplex Switched Ethernet (AFDX) Network. ARINC Specification 664p7.

[8] Charara, H., & Fraboul, C. (2005, July). Modelling and simulation of an avionics full duplex switched ethernet.In Telecommunications,2005.advanced industrial conference on telecommunications/service assurance with partial and intermittent resources conference/e-learning on telecommunications workshop. aict/sapir/elete 2005. proceedings (pp. 207-212). IEEE.

[9] Dong, S., Xingxing, Z., Lina, D., & Qiong, H. (2010, October). The design and implementation of the AFDX network simulation system. In Multimedia Technology (ICMT), 2010 International Conference on (pp. 1-4). IEEE.

[10] Jiqiang, X., Weimin, Y., & Ronggang, B. (2011, July). Study on Real-Time Performance of AFDX Using OPNET. In Control, Automation and Systems Engineering (CASE), 2011 International Conference on (pp. 1-5). IEEE.

[11] Täubrich, J., & Von Hanxleden, R. (2007). Formal specification and analysis of AFDX redundancy management algorithms. In Computer Safety, Reliability, and Security (pp. 436-450). Springer Berlin Heidelberg.

[12] Lauer, M. (2012). Une méthode globale pour la vérification d'exigences temps réel: application à l'Avionique Modulaire Intégrée (Doctoral dissertation, Institut National Polytechnique de Toulouse-INPT).

[13] Scharbarg, J. L., & Fraboul, C. (2010). Methods and tools for the temporal analysis of avionic networks. New trends in technologies: control, management, computational intelligence and network systems, 413-438.