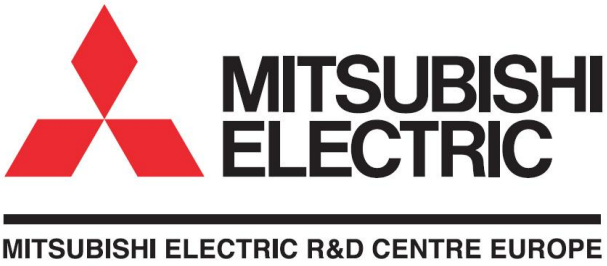

© 2015 IEEE

IEEE International Conference on High Performance Switching and Routing, Budapest, Hungary, July 2015

Experimental Study of a Low-Delay Ethernet Switch for Real Time Networks

Y. Hotta
A. Inoue
H. Bessho
C. Mangin
R. Kawate

Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.”

Experimental Study of a Low-Delay Ethernet Switch for Real Time Networks

Yoshifumi Hotta[1] , *Member, IEEE*, Ayako Inoue[1], Hiroshi Bessho[1], Christophe Mangin[2] and Ryusuke Kawate[1]
[1] Information Technology R&D Center, Mitsubishi Electric Corporation
[2] R&D Centre Europe, Mitsubishi Electric Corporation

***Abstract*— The past few decades have seen a large growth in the number and type of communication buses used in vehicle, train, and power plant. Recently, Ethernet is considered as a candidate for next generation network protocol for these networks because of its scalable bandwidth, variety of available devices and cost effectiveness. However, reliability and delay performance of conventional Ethernet switches are not sufficient for industrial application. To solve the delay performance problem, an express frame preemption method is currently being developed in the IEEE 802.3br task force. In this paper, the FPGA-based implementation of a four-port Ethernet switch featuring IEEE 802.3br MAC functions is presented and delay measurement is conducted to evaluate the latency experienced by the express frames. The measurement results confirm that the maximum latency of express frames could be significantly reduced to 2.46µs compared to the conventional switch delay of 27.57µs.**

I. INTRODUCTION

Ethernet is now the dominant, not only local area network technology in the home and office environment, but also in tele-communication systems [1]. Ethernet has been considered as a candidate for industrial network because of its high bandwidth, cost effectiveness and variety of the devices. However, conventional Ethernet is not capable to support the real-time communications required by the industrial applications, so that several investigations have been conducted to introduce Ethernet-based field buses to industrial network [2], [3], [4]. Since standard Ethernet is not able to meet the industrial network requirements, real-time protocols are defined on the top of TCP/IP or Ethernet as define in IEC 61158 and IEC 61784-2 [5].

On the other hand, the automotive industry is considering Ethernet as a candidate for next generation in-vehicle bus because in-vehicle networks have become complex and costly due to the growing number of automotive applications requiring communications. Conventional in-vehicle network has been implemented using different automotive network technologies such as Media Oriented System Transport (MOST), Controller Area Network (CAN), and Local Interconnect Network (LIN) which have been developed for specific applications (multimedia flow distribution, low-speed control) do not provide limited the transmission capabilities and scalability required by the emerging automotive applications. Therefore, solutions to replace these conventional in-vehicle busses by Ethernet are investigated [6]. It seems possible to replace MOST by Ethernet for audio and video applications, however there are difficulties to accommodate real-time signals, currently transported over CAN and LIN, with conventional Ethernet due to its delay performance and lack of reliability.

The real time problem is caused by the nature of the Ethernet MAC. When an express frame is transferred to an egress port just after the egress port starts transmitting a preceding frame, the express frame is stored in the buffer and must wait for the end of the preceding frame transmission. The frame length which is currently supported in IEEE 802.3 ranges from 64 bytes to 2000 bytes [7], therefore if the port speed is 1 Gb/s, the express frame can be delayed in the switch for approximately 512 ns (64 byte time) to 16 µs (2000 byte time). This delay and the correlated delay variation can adversely affect to the real-time application which delay requirements can be less than 3µs [8] and delay jitter requirement, less than 1µs [3]. To address this problem, the IEEE 802.3br [9], Interspersing Express Traffic (IET) task force is currently developing MAC mechanisms for frame preemption. In parallel, the IEEE 802.3bp [10] 1000 Base-T1 task force (1 Gb/s over one twisted copper pair) carries out PHY technology investigations to reduce the number of twisted pairs from conventional four pairs down to one pair to achieve 1Gb/s bandwidth, with wire harness weight reduction and simple wiring.

In order to evaluate the technical feasibility of those technologies, it is important to perform an experimental study. In a first step, we focus on low-delay frame transmission. The aim of this work is to implement an IEEE 802.3br-based MAC logic in an FPGA and to experimentally evaluate the delay performance of express frames in case of transmission conflict with non-express frames.

The paper is organized as follows. Section II introduces the delay performance problem of the conventional Ethernet switch and the frame preemption mechanisms that are discussed in the IEEE 802.3br task force. Section III details the implementation of the IEEE 802.3br-compliant four-1Gb/s-ports Ethernet switch which is implemented in the FPGA evaluation board. Section IV shows the evaluation results both for express frame delay and jitter in the condition where normal and express frames conflict on the egress port. The measurement is conducted for both a conventional Ethernet switch with strict priority frame transmission selection on the egress port and the newly designed Ethernet switch. Finally, Section V concludes the paper.

II. Delay Performance Problem of Ethernet Switch and IEEE 802.3br Solution

In this section, first, the delay performance problem is described when the frames with different delay requirement are multiplexed in Ethernet switch. Then, the solution developed in IEEE 802.3br to overcome that problem, is introduced.

A. *Delay performance problem of the Ethernet switch*

In a converged architecture, the network will have to multiplex both time-critical control frames used by actuators or sensors and non-time-critical frames such as those carrying HTTP, email and FTP. Basically, the real-time data for actuators and sensors fit into short datagram; however they need to be delivered with low delay in order to meet their tight timing requirements (time- or event-triggered signals) [4]. On the other hand, typical non-time-critical applications do not need real-time transmission; but may be transported in larger datagram up to 2000 byte.

If the conventional Ethernet multiplexing is applied to the future network mentioned above, the time-critical frame will be adversely affected by the non-time-critical frames. As shown in Fig. 1. When the time-critical (Express in Fig.1.) arrives in the egress port just after a non-time-critical frame (Normal in Fig. 1.) transmission is started, the express frame must wait for the normal frame transmission to complete. In the worst case, when the line rate is 1 Gb/s, the express frame experiences a delay of up to 16 us per hop even if the strict priority transmission

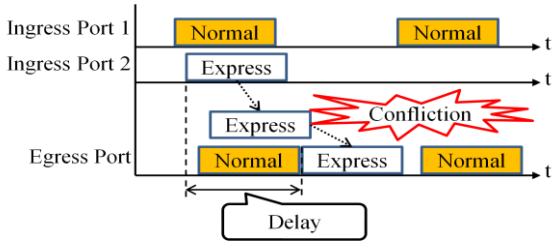

Fig. 1. Conventional Ethernet switch delay problem on the frame confliction

selection is applied. This problem introduces unpredictable delay and frame jitter to the express frames. When we consider a converged network which accommodates both real-time and non-real-time applications with simple network configuration, this kind of problem makes network design difficult.

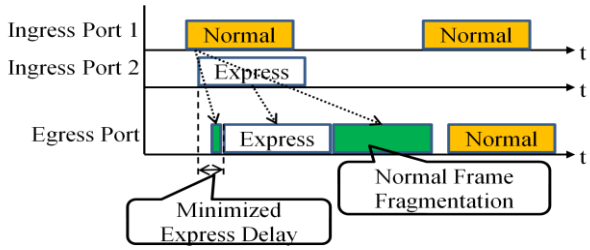

Fig. 2. Frame preemption discussed in IEEE 802.3br task force

B. *IEEE 802.3br frame preemption*

To overcome the problem, a frame preemption media access control (MAC) mechanism is currently developed by the IEEE 802.3br task force [9]. This frame preemption mechanism is illustrated in Fig. 2.

With newly designed MAC, when an express frame arrives at the egress port during the transmission of a normal frame, the express frame is able to preempt the normal frame transmission. The standardized mechanism has been carefully designed to

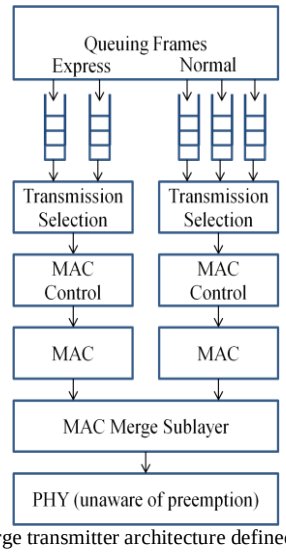

Fig. 3. MAC merge transmitter architecture defined in IEEE 802.3br

preserve compatibility with the conventional Ethernet frame format and minimum and maximum MAC frame sizes. Using frame preemption, the latency to initiate transmission of an express frame shall be less than two times the minimum packet size plus inter-packet gap.

To realize frame preemption, a MAC merge transmitter architecture is defined as shown in Fig. 3.

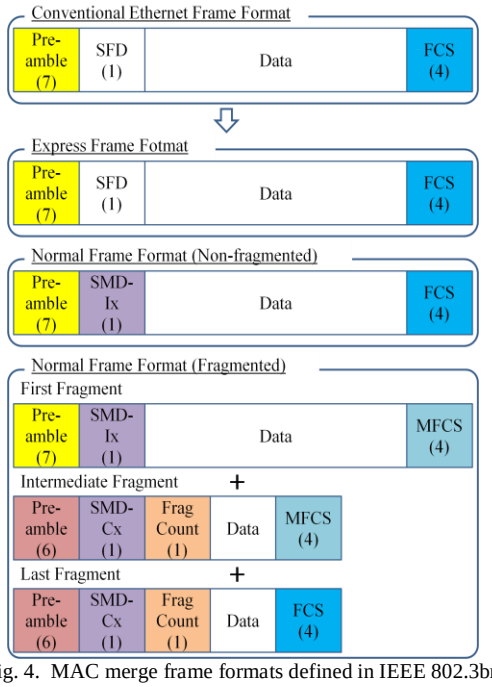

Fig. 4. MAC merge frame formats defined in IEEE 802.3br

According to the proposed MAC merge architecture, both express and normal data paths have exclusive queues, transmission selection, MAC control and MAC functions. The MAC merge serves both express and normal paths and operates so as to transmit express frame immediately.

The MAC merge frame formats are also defined to distinguish among express frames, fragmented normal frames and non-fragmented normal frames as shown in Fig. 4.

The conventional Ethernet frame format has seven bytes preamble and one byte SFD (Start Frame Delimiter). The data length of Ethernet frame ranges from 64 bytes to 2000 bytes of which the last four bytes contain the FCS (Frame Check Sequence) field. The express frame has same format as the conventional Ethernet. To identify the non-fragment and the first fragment normal frame, the SMD (Start MAC merge frame Delimiter) field is introduced. The SMD-Ix indicates the beginning of a normal frame, whereas the SMD-Cx and FragCount fields are defined to signal frame fragments. FragCount relies on a circular numbering from 0 to 3 for the

TABLE I
ENCODE VALUE FOR SFD, SMD FRAGMENT COUNT

Frame type	SFD/SMD	Frame count	Encoded value
Express	SFD	N/A	0xD5
Beginning of the Normal frame	SMD-Ix	0	0xE6
		1	0x4C
		2	0x7F
		3	0xB3
Remnant of the Normal frame	SMD-Cx	0	0x61
		1	0x52
		2	0x9E
		3	0xAD
	FragCount	0	0xE6
		1	0x4C
		2	0x7F
		3	0xB3

protection against reassembly errors upon frame fragment loss. The MFCS field is defined to mark the end of non-final fragments. The calculation of MFCS is done per fragment by exclusive-ORing the fragment FCS with 0xFFFF0000. On the receiver Merge MAC sublayer side, the preamble information is used to distinguish whether the received frame is an express frame or a normal frame. If the received frame is a fragmented normal frame, then the receiver Merge MAC sublayer waits for the remnant frame fragments to reassemble the frame. When the receiver Merge MAC sublayer receives a normal frame with no FCS or reassembly errors, then the normal frame is transferred to the MAC.

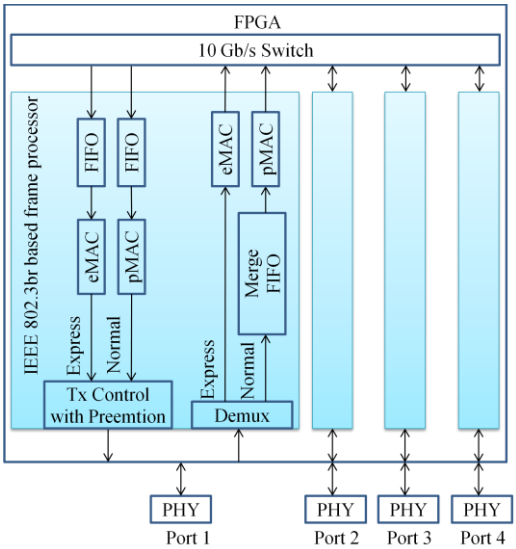

Fig. 5. Designed IEEE 802.3br based Ethernet switch architecture

III. IEEE 802.3BR-BASED ETHERNET SWITCH DESIGN

The architecture of the IEEE-802.3br-based four-port Ethernet switch is illustrated in the Fig. 5.

On the transmitter side, two paths are connected to the 10 Gb/s capable Ethernet switch each of which being respectively dedicated to the express and the normal frames. Both paths have dedicated FIFOs, eMAC and pMAC, and a transmission control with preemption function, denoted by "Tx control with preemption", that selects the frame to be transmitted, based on a transmission algorithm. The preamble modification according to IEEE 802.3br and MFCS/MFCS calculation are also conducted in this block.

On the receiver side, each port has a demultiplexer, denoted by "Demux", that parses the last byte of the preamble to determine if the received frame/fragment is to be forwarded to the express or normal frame path. The express path is directly connected to the express MAC, denoted by "eMAC", and the normal path is connected to the merge FIFO that is used for the frame reassembly if the frame is fragmented. If the normal frame is not fragmented or the fragmented frames are correctly reassembled, the assembled frames are transferred to the preemptable MAC, denoted by "pMAC". Otherwise upon detection of a reassembly error, by checking SMD-Ix, SMD-Cx and FragCount or MFCS/FCS, the errored frame is discarded in this block.

To realize this switch, a Xilinx FPGA XC7Z020 and conventional Ethernet PHY (Marvell 88E1111) are used.

A. *Transmission control with preemption*

Designed transmission control algorithm is illustrated in Fig. 6. To verify the minimum MAC frame size imposed by the IEEE 802.3 standard, normal frames shorter than 128 bytes are not fragmented. Therefore, when an express frame transmit indication is received from eMAC while transmitting a normal

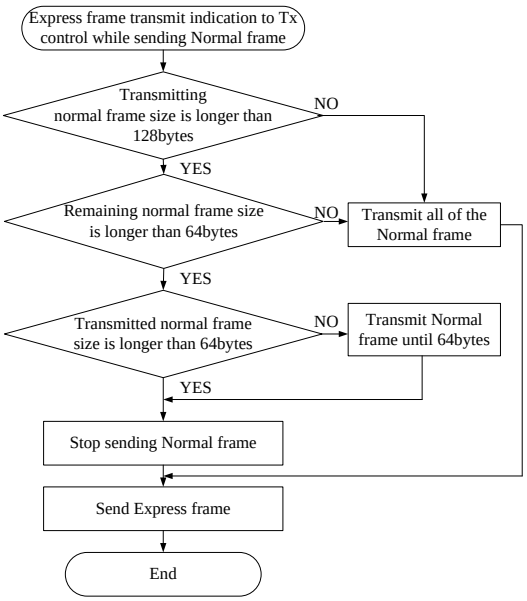

Fig. 6. Tx Control with preemption block flowchart

frame, the Tx control checks whether the total size of the concurrently transmitted normal frame is longer than 128 bytes. If the size is less than 128 bytes, the Tx control continues transmitting the normal frame. Otherwise if the normal frame size is longer than 128 bytes, the Tx control checks whether remaining frame size is longer than 64 bytes. If the remaining size is less than 64 bytes, then the Tx control continues transmitting the normal frame. Otherwise, if the remaining size is longer than 64 bytes, the Tx controller terminates transmitting the normal frame from pMAC and adds the MFCS to the end of the preempted normal frame. Then after inter-packet gap is secured, the express frame is transmitted to the PHY.

The preamble modification is conducted on the normal frame transmission. The SMD-Ix is inserted as eighth byte of initial fragment or non-fragmented normal frame preamble only, and the frame count for this value is increased upon initial fragment or non-fragment normal frame transmission. The SMD-Cx is inserted as seventh byte in the preamble of the intermediate and the last frame fragment to indicate that they belong to the same original MAC frame. Therefore, the SMD-Cx value is increased only when a different MAC frame is fragmented. The FragCount is inserted as eighth byte of the non-initial fragmented normal frame to indicate the fragment transmission order. Therefore the FragCount value is incremented upon each fragmented transmission.

B. Receiver demultipexer, merge FIFO control and 10 Gb/s switch

On the receiver side, the demultiplexer distinguishes the express frames from the normal frames. This function is simply implemented by checking the eighth preamble byte. If this byte matches SFD then the frame is recognized as an express frame and transferred to the eMAC after FCS check. Both the eMAC and the 10 Gb/s switch transfer the express frames through the cut-through path to the destination port to reduce the internal delay as much as possible.

On the other hand, if a received frame preamble is followed by SMD-Ix or SMD-Cx, the frame is transferred to the merge FIFO control. The merge FIFO control evaluates the preamble and FCS/MFCS value consistency. The merge FIFO control transfers reassembled MAC frames only when the normal frame is reassembled with no preamble and FCS/MFCS error, otherwise it is discarded. The merge FIFO control concurrently evaluates FCS and MFCS so that the end of fragmented frame is detected when the difference between CRC 32 calculation and FCS field value of the frame is 0xFFFFFFFF. Else if the difference between CRC 32 and FCS field value of the frame is 0x0000FFFF, the frame is treated as non-final fragment frame.

IV. EVALUATED SYSTEM CONFIGURATION AND RESULTS

The experimental system configuration is illustrated in Fig. 7(a). In the experimental system, two switches are connected to each other via port 1 of both switch#1 and switch#2. Also, ports 3 and 4 for both the switch#1, 2 are connected to a LAN analyzer. Both normal and express frames are sent to switch#1 and port 3 and 4 receive each of them and transfer frames to port 1.

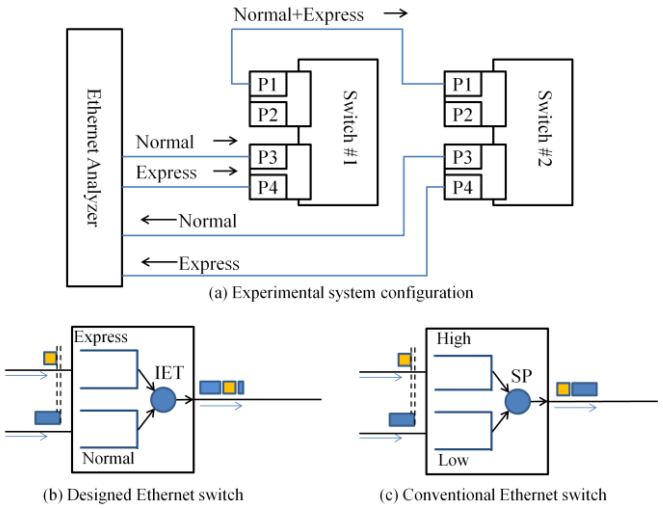

Fig. 7. Evaluated system configuration

Switch#2 receives both normal and express frames from port 1, which are respectively transferred to ports 3 and 4 to measure delay performance.

To compare delay performance, both a conventional switch and the designed switch are evaluated. As explained above, the designed switch supports a normal FIFO and an express FIFO per port and the IEEE-802.3br-based frame preemption egress control is implemented as shown in Fig. 7(b). The conventional switch evaluation is conducted by using strict priority egress transmission selection in switch#1 as shown in Fig. 7(c).

TABLE II
TRAFFIC CONDITIONS

Case	Data rate	Frame length
1	Express 70 Mb/s Normal 920Mb/s	Express 256 byte Normal 1500 byte
2		Express 256 byte Normal 1024byte
3		Express 256 byte Normal 512 byte
4		Express 256 byte Normal 256byte

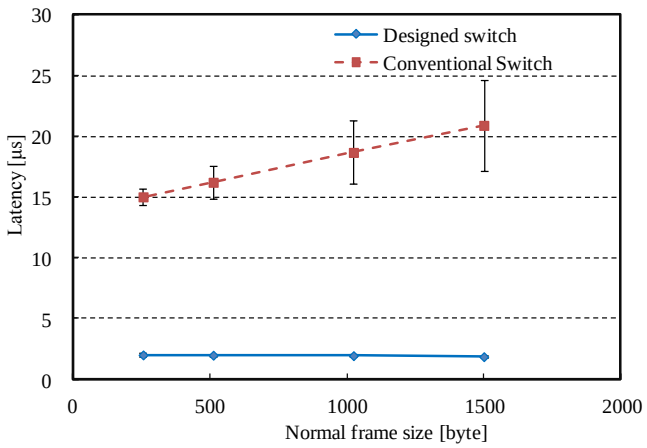

Fig. 8. Relationship between Normal frame length and express frame latency

Traffic conditions which is generated by LAN analyzer is shown in Table II. The evaluations are conducted using four cases. In all cases, the data rate for both express and normal traffic is fixed to 70 Mb/s and 920 Mb/s, respectively. The express frame length in each case is fixed to 256 byte; however the normal frame length is varied from 256 byte to 1500 byte in each case to evaluate the influence on the express frame delay performance. The results of the measured delay experienced by the express frames are provided in Fig. 8. The solid line shows

TABLE III
DELAY MEASUREMENT RESULTS FOR DESIGNED SWITCH

Case	Express frame delay (μs)			
	maximum	minimum	Average	Standard deviation
1	2.45	1.74	1.94	0.08
2	2.42	1.84	1.95	0.11
3	2.46	1.75	1.99	0.15
4	2.46	1.74	2.02	0.17

TABLE IV
DELAY MEASUREMENT RESULTS FOR CONVENTIONAL SWITCH

Case	Express frame delay (μs)			
	maximum	minimum	Average	Standard deviation
1	27.57	14.27	20.91	3.75
2	23.19	14.30	18.68	2.60
3	18.69	13.97	16.22	1.33
4	16.49	13.96	15.03	0.68

measured latency of the express frame in the designed switch and the dashed line shows that experienced in the conventional switch. The range bars represent standard deviations. Additionally, Table III and IV summarize the delay measurement results for the designed and the conventional switches, respectively. The delays of the express frames were less than 2.46 μs and frame jitter was less than 0.72 μs in the designed switch. In contrast, the express frames maximum delay measurements obtained with the conventional switch range from 16.49 μs to 27.57 μs, while frame jitter varies from 2.53 μs to 13.30 μs depending on the normal frame length.

The latency distribution observed for 10,000 express frames with the designed switch in each are provided in Fig. 9. In case 1, approximately 93.7% of the express frames are transferred within 2.0 μs, and the percentages of the express frames which have latency below 2.0 μs are reduced to 90.5%, 77.61%, 63.4% with decreasing frame length as shown in cases 2, 3, 4. The apparent latency increases of the express frames are attributable to the increasing probability of express frames buffering during the transmission of a normal non-fragmentable frame. The latency distributions observed for 10,000 express frames with the conventional switch in each case are provided in Fig. 10. The express frame delay varies from approximately 14.3 μs to 20.9 μs and the delay variations have a broad distribution with the increase of the interfering normal frame length.

To summarize these results, two main observations in this experimental study are as follows. The first is that the delay of the express frames in the designed switch is less than 2.46 μs and, compared to the conventional switch, the maximum

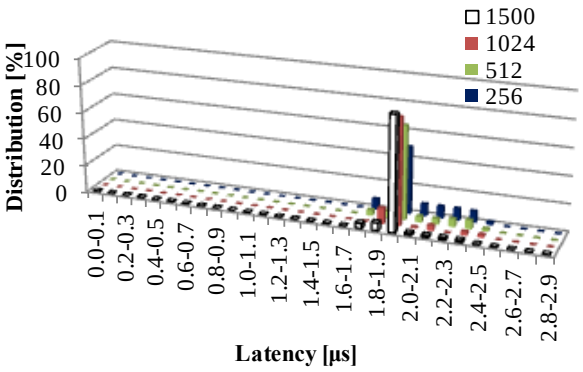

Fig. 9. Designed switch delay distribution of the express frame

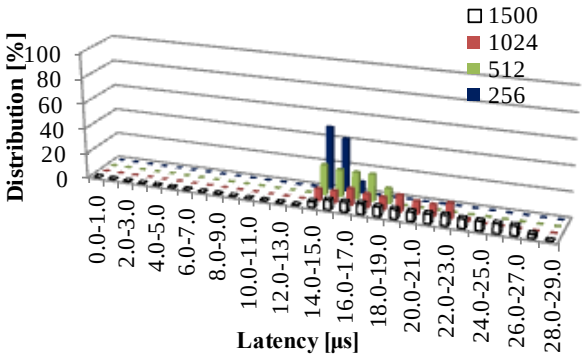

Fig. 10. Conventional switch delay distribution of the express frame

latency is reduced to approximately 1/10 when the normal frames length is 1500 byte. The second is that the delay variation of the express frames in the designed switch is drastically reduced to less than 0.72 μs, compared to the 13.30 μs measured in the conventional switch. It should be noted that more than 90% of the express frames are transferred with a latency comprised between 1.7 μs and 2.0 μs so that the express frames jitter is also significantly reduced when the normal frame length is larger than 1024 bytes.

Comparing latency and delay variation results to the requirements (latency < 3 μs/hop@ 1 Gb/s [8], jitter < 1 μs [3]), it may be concluded that Ethernet switch including the frame preemption egress function is suitable for real time network such as industrial and in-vehicle networks.

V. CONCLUSION

In order to evaluate technical feasibility of IEEE-802.3br-based real-time Ethernet multiplexing technologies, a four-port Ethernet switch with IEEE 802.3br MAC is implemented in an FPGA-based board and delay measurements are conducted to evaluate the latency experience by the express frames. Consequently, it is confirmed that the maximum latency of express frames could be significantly reduced to 2.46μs compared to the conventional switch delay of 27.57μs. The delay variation performance of the express frames with the designed switch is greatly improved: less than 0.72 μs,

compared to the 13.30 μs with the conventional switch. It should be noted that more than 90% of the express frames are transferred with a jitter within 0.3 μs, therefore jitter for express frame is also significantly reduced when the normal frame length is larger than 1024 byte.

Comparing latency and delay variation results to the requirements, it can be concluded Ethernet switch with frame preemption egress function is suitable for real-time network such as industrial and in-vehicle networks.

REFERENCES

[1] K. Fouli and M. Maier, "The road to carrier-grade Ethernet," *IEEE Commun. Mag*., vol. 47, no. 3, Mar. 2009, pp. S30–S38.

[2] H. Kopetz, A. Ademaj, P. Grillinger, and K. Steinhammer. " The Time-Triggered Ethernet (TTE) Design," *Proc. 8th IEEE Int. Symp. Object-oriented Realtime distributed Computing (ISORC)*, Seattle, Washington, May 2005.

[3] M. Felser, "Real-Time Ethernet – Industrial Perspective," *Proc. IEEE*, vol. 93, no. 6, pp. 1118 – 1129, June 2005.

[4] J. D. Decotignie, "Ethernet-Based Real-Time and Industrial Communications," *Proc. IEEE*, vol. 93, no.6, pp. 1102-1117, June 2005.

[5] L. Winkel, A.G. Siemens, Karlsruhe, "Real-time Ethernet in IEC 61784-2 and IEC 61158 Series", *Proc. IEEE, Int. Conf. Ind. Informat.*, Singapore, pp.246–250, 2006.

[6] M. Rahmani, K. Tappayuthpijarn, B. Krebs, E. Steinbach, and R. Bogenberger "Traffic shaping for resource-efficient in-vehicle communication," *IEEE Trans. Ind. Informat.* vol. 5, no. 4, pp. 414-428, Nov 2009.

[7] *IEEE Standards for Local Area Networks: Part 1: Carrier Sense Multiple Access With Collision Detect on (CSMA/CD) Access Method and Physical Layer Specifications*, IEEE Std. 802.3, 2012

[8] L. Winkel, "Distinguished minimum latency traffic in a converged traffic environment" [Online].
Available:
http://www.ieee802.org/802_tutorials/2013-07/Winkel_00_0713_DMLT_SG_Tutorial_v04.pdf

[9] *IEEE 802.3br Interspersing Express Traffic task force baseline*, ver.1.0, rev. 3, 2014. [Online].
Available:
http://www.ieee802.org/3/br/Baseline/8023-IET-TF-1405_Winkel-iet-Baseline-r3.pdf

[10] *IEEE 802.3bp 1000 BASE-T1 task force*, [Online].
Available: http://ieee802.org/3/bp/