An Analysis of Frame Replication and Elimination for Time-Sensitive Networking

Shifei Qian
Tongji University
No. 4800, Cao'an Road
Jiading District, Shanghai
+86 17721352664
qian_shifei@163.com

Feng Luo
Tongji University
No. 4800, Cao'an Road
Jiading District, Shanghai
+86 13817053378
luo_feng@tongji.edu.cn

Jinpeng Xu
Tongji University
No. 4800, Cao'an Road
Jiading District, Shanghai
+86 17302131360
1310790@tongji.edu.cn

ABSTRACT
Automotive Ethernet is becoming the backbone network of future vehicle because of its large data bandwidth and high communication speed. In order to satisfy the requirements of vehicle communication, the network needs the characteristic of seamless redundancy. IEEE P802.1 CB is one of the TSN (Time-Sensitive Networking) active projects, which provides seamless redundancy character for Automotive Ethernet by frame replication and elimination at flexible position to improve reliability. In this paper, the operating principle of IEEE P802.1 CB was analyzed, including network topology, redundancy tag, sequence generation function, sequence recovery function, timeout mechanism and latent error detection. A network complying IEEE P802.1 CB was simulated in Visual Studio and 7 situations of packet transmission were tested in CppUTest. The results show that when IEEE P802.1 CB is used over a network that are fixed to a specific topology, and that are protected against congestion loss, it can substantially increase seamless redundancy character for the network and is especially suitable for time-critical system.

CCS Concepts
• **Networks ➝ Network protocols ➝ Protocol correctness ➝ Protocol testing and verification**

• **Networks ➝ Network performance evaluation.**

Keywords
Automotive Ethernet; Seamless Redundancy; IEEE P802.1 CB.

Conference'10, Month 1–2, 2010, City, State, Country.

DOI: http://dx.doi.org/10.1145/12345.67890

1. INTRODUCTION
With the increasing complexity of automotive network, traditional automotive network cannot satisfy internal communication bandwidth requirements such as enhanced safety and entertainment solutions. Compared with CAN, LIN, FlesRay, MOST, Automotive Ethernet has obvious technical features, Table 1 has listed the comparison of the parameters of CAN, LIN, FlesRay, MOST and Automotive Ethernet [1-2], so Automotive Ethernet is becoming the backbone network of future vehicle because of its large data bandwidth and high communication speed [3-4].

Automotive Ethernet needs several switches to deliver frames between end stations, for traditional Ethernet, if one switch fails, the frames will be lost, and several seconds are needed to find a new way for frame delivery. However it is unacceptable for automotive communication because of its high importance for safety, so IEEE 802.1 Time-Sensitive Networking Task Group is founded to make Automotive Ethernet satisfy the harsh requirements of vehicle communication [5].

IEEE P802.1 CB is one of the TSN active projects, which provides seamless redundancy character for Ethernet network by frame replication and elimination to improve reliability. IEEE P802.1 CB, provides increased reliability (reduced packet loss rates) for a stream by sequence numbering and replicating every packet, in the source end system and relay systems of the network, and eliminating those replicates in the destination end system and in other relay systems [6].

This paper first introduces the operating principle of IEEE P802.1 CB, including network topology, redundancy tag, sequence generation function, sequence recovery function, timeout mechanism and latent error detection function. Then a redundant network complying IEEE P802.1 CB is simulated in visual studio and tested through CppUTest. Finally, a conclusion is given that when IEEE P802.1 CB is used over paths that are fixed to a specific topology, and that are protected against congestion loss, it can substantially increase seamless redundancy character for Ethernet network and is especially suitable for time-critical system.

2. IEEE P802.1 CB operating principle
2.1 Network topology
Highly seamless communication with fault tolerance is one of the key requirements for Ethernet-based, mission-critical and real-time systems, such as substation automation system (SAS) networks and other industrial Ethernet networks [7].

IEEE P802.1 CB provides flexible topology for Automotive

Table 1. Comparison of the parameters of different automotive network

network parameter	CAN	LIN	FlexRay	MOST	Automotive Ethernet
maximum transfer rate	1Mb/s	20Kb/s	10Mb/s	150Mb/s	100Mb/s
maximum node number	32	16	64	24	depends on port number of the switch
supported bus length	40m	40m	24m	1280m	15m/link
transfer mechanism	CSMA/CA	Time-Triggered	Time-Triggered	Time-Triggered	full duplex
Cost	low	very low	relatively high	high	relatively high
Topology	bus	bus	bus	ring	arbitrary
Support sleep mode	yes	yes	yes	yes	yes
Standard	ISO 11898	ISO 17987	ISO 17458	no official standard	802.3bw
Cable	UTP	single wire	UTP	optical fiber	UTP

Ethernet, which means this standard will work on all LAN topologies. Figure 1 illustrates an example: a sequence number is generated and encoded into each packet in the leftmost box. Sequence recovery functions eliminate duplicate packets, and the non-duplicate packets are copied as a new member stream (with sequence numbers unchanged) at two intermediate points. The final two member streams are brought together and the duplicates are eliminated at the destination at right. This configuration protects against all 7 possible one-link failures, and against 16 of 21 possible two-link failures.

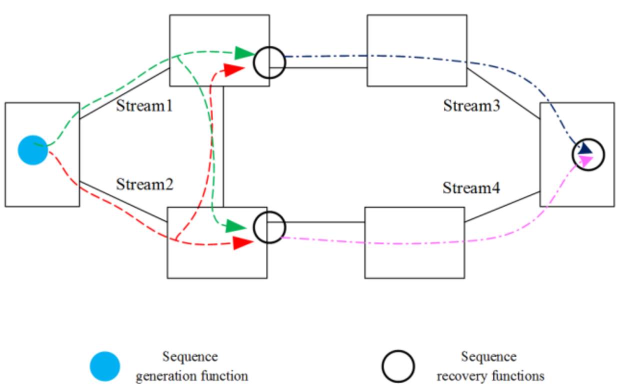

Figure 1. Topologies of IEEE P802.1 CB

2.2 Redundancy tag

In this paper we use raw Ethernet frame with VLAN, in order to realize frame replication and elimination for seamless redundancy, a redundancy tag needs to be added into the Ethernet frame to support IEEE P802.1 CB, as illustrated in Figure 2. A redundancy tag consists of a tag protocol identifier EtherType and a sequence number. EtherType is a constant, and sequence number identifies the order in which the packet was transmitted relative to other packets in the same compound stream.

2.3 Sequence generation function

Frame replication is realized by sequence generation function, it is called whenever the data request event occurs. It contains a counter, every time sequence generation function is called, the counter is incremented by one, and its value is copied to the sequence number of the redundancy tag of the packet, then the packet is passed to the next lower layer. If the counter exceeds its maximum value, it returns to zero.

For example, in Figure 3, the talker system T generates packets with consistent sequence number, and duplicates each packet to a packet pair, each packet pair have identical sequence number, they are sent separately through path A and path B to listener system L.

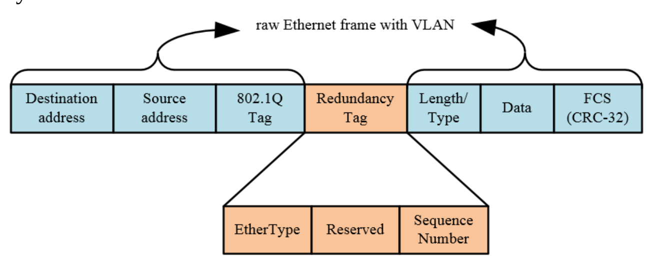

Figure 2. raw Ethernet frame with VLAN and redundancy tag

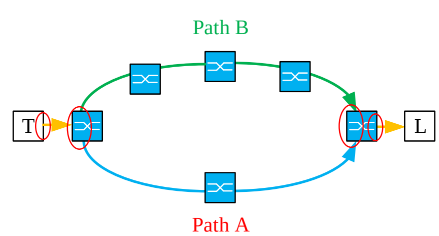

Figure 3. Example of IEEE P802.1 CB

2.4 Sequence recovery function

Frame elimination is realized by sequence recovery function, it is called whenever the data indication event occurs. It operates on a merged set of member streams originally marked with sequence number values from a single instance of the sequence generation function. It evaluates the sequence number of a packet of one or more member streams passed up from the lower layers, in order to decide which packets to pass and which to discard.

Sequence recovery function has many sequence recovery algorithms, in this paper we use vector recovery algorithm. Vector recovery algorithm is illustrated in Figure 4, MaxSeqNum holds the highest sequence number value received. SequenceHistory is shown in Table 2, it maintains the history of the sequence number of recently received packets, its length is SequenceHistoryLength. SequenceHistory is a bit vector, with one bit for each value from 0 to (SequenceHistoryLength – 1), corresponding to sequence number from MaxSeqNum to (MaxSeqNum–SequenceHistoryLength + 1). A 1 in the SequenceHistory indicates that the corresponding sequence number has been received, and a 0 that it has not.

Immediately after the sequence recovery function is reset, it accepts the first packet received as valid. After the first packet has been accepted, if the sequence number of the subsequent packet is outside the window MaxSeqNum± SequenceHistoryLength, it will be discarded. If the sequence number of the packet is inside the window MaxSeqNum- SequenceHistoryLength, and its sequence number value has been seen before, it will be discarded, else it will be passed and SequenceHistory will be updated. If the sequence number of the packet is inside the window MaxSeqNum+ SequenceHistoryLength, it will be passed, moreover MaxSeqNum and and SequenceHistory will be updated.

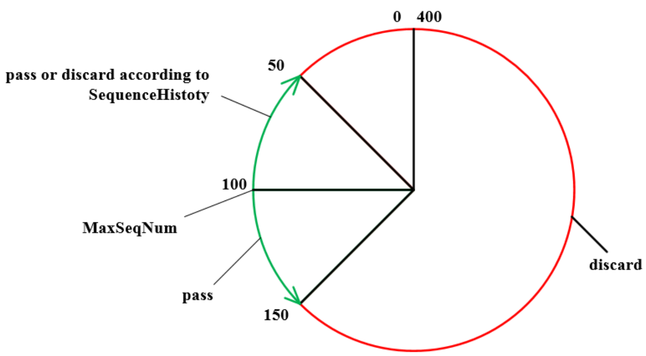

Figure 4. Vector recovery algorithm

Table 2. Sequence history

Sequence number	100	99	98	97	96	…	50
Has been received or not	1	0	1	0	0	···	0

2.5 Timeout mechanism

Each packet accepted and passed up the stack resets a timer, after reset it will tick down, when it turns to zero, meaning that no packet has been accepted during this time, the sequence recovery function will resets the algorithm, and the next packet received will be accepted. The timeout mechanism will resolve the following fault situations:

1) “Rogue” packets, meaning packets are discarded as invalid.

2) If a sequence recovery function somehow gets out of step with its corresponding sequence generation function, then after the specific time, the sequence recovery function will be reset and data will again be passed.

3) If a sequence generation function is reset, perhaps by rebooting a system, then sequence recovery functions that have not been reset are likely to discard packets until the timeout event has occurred.

4) If a talker or a relay system fails in such a way as to repeatedly transmit packets with the same sequence number, those packets will continue to be discarded, at least until the sequence number wraps around.

2.6 Latent error detection function

Latent error detection function monitors the managed objects associated with a single sequence recovery function, in order to detect the condition that relatively few packets are being discarded by that function. Latent error detection operates on the assumption that, in a properly functioning compound stream employing n paths into the current system, there will be n–1 packets discarded for every packet passed through the sequence recovery function. The latent error detection function issues a latent error signal when that assumption is violated.

2.7 Seamless redundancy character

In this section a network which does not complying IEEE P802.1 CB and a network complying IEEE P802.1 CB will be compared to show the advantages of using IEEE P802.1 CB.

In the network which does not complying IEEE P802.1 CB, 9 packages with consistent sequence number will be sent through only one path from the talker system T to the listener system L, as illustrated in Figure 5, among them packets with sequence number 3, 5 and 7 are lost for any reason, the result of the transmission is shown in Figure 6, there are 3 serials of packets in Figure 6, the first serial of packets represent the packets generated by the talker system T. The second serial of packets represent the packets received at the listener system L. The last serial of packets represent the packets passed up to the higher layer by the listener system L. The result shows that at the listener system L, the packets with sequence number 3, 5 and 7 are lost.

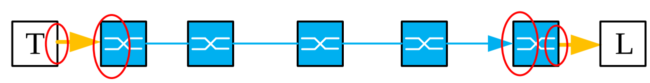

Figure 5. Network without IEEE P802.1 CB

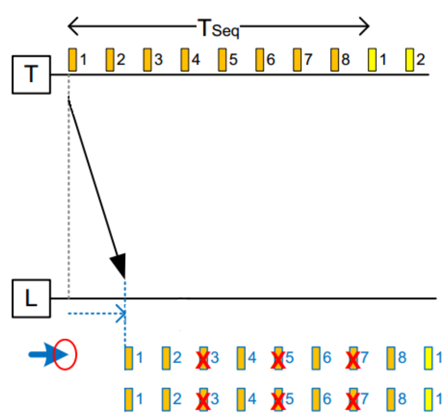

Figure 6. Result of network without IEEE P802.1 CB

In the network complying IEEE P802.1 CB, 9 packages with consistent sequence number will be sent through path A and path B from the talker system T to the listener system L, as illustrated in Figure 3, among them packets in path A with sequence number 3, 5 and 7 are lost for any reason, the result of the transmission is shown in Figure 7, there are 4 serials of packets in Figure 7, the first serial of packets represent the packets generated by the talker system T before they are split into path A and path B. The second and third serials of packets represent the packets of path A and path B received at the listener system L. Since there are more bridges in path B, the packets of path B arrive later than that of path A. The last serial of packets represent the packets passed up to the higher layer by the listener system L. The result shows that because of using IEEE P802.1CB, the packets with sequence number 3, 5 and 7 are not lost at the listener system L.

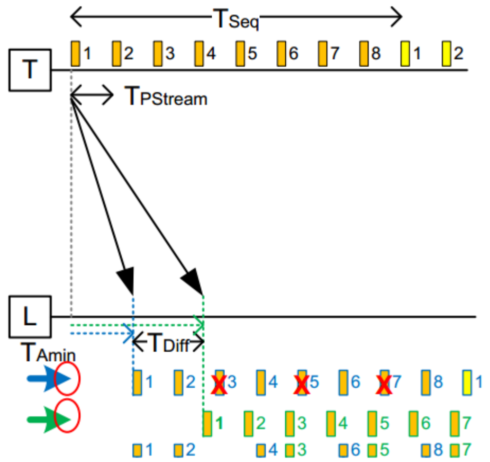

Figure 7. Seamless redundancy of IEEE P802.1 CB

With the comparison of the two networks, the seamless redundancy character of using IEEE P802.1 CB is demonstrated as follows:

1) For duplicated packets with the same sequence number of path A and path B, the listener system L passes the earlier one and discards the later one, so IEEE P802.1 CB achieves zero delay switchover making it suitable for the time-critical system;

2) Although the packets with sequence number of 3, 5 and 7 are lost in path A, since the packets with the same sequence number in path B are not lost, so there are no packets lost in the view of the listener system L, therefore the redundancy character of the network is enhanced;

3) IEEE P802.1 CB dose not rely on higher layer protocols, it is transparent to the application.

3. Simulation and test of IEEE P802.1 CB

In this paper, a network complying IEEE P802.1 CB is simulated and tested through visual studio and CppUTest, the test environment is illustrated in Figure 8. A network complying IEEE P802.1 CB is simulated in visual studio, then in CppUTest different packets are simulated and transmitted to the network, the test results are reported in CppUTest, including pass or discard the packets, reset or do not reset the algorithm, issue or do not issue the latent error.

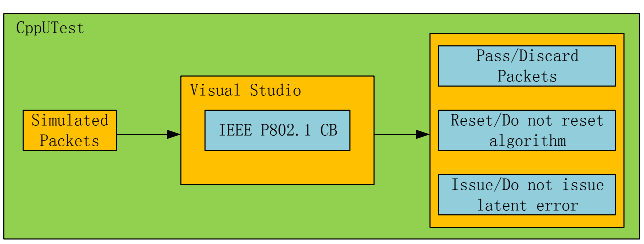

Figure 8. Test environment of IEEE P802.1 CB

Visual studio is a fully-featured integrated development environment for android, iOS, Windows, web and cloud. In this paper a network complying IEEE P802.1 CB as Figure 4 is simulated, the simulation parameters of the network are listed in Table 3. SequenceHistoryLength is used for sequence recovery function, it represents the size of sequence recovery window, its function has been discussed in 2.4, if this parameter is larger, more packets will be accepted by sequence recovery function. Timeout timer is used for timeout mechanism, it represents the timer which is reset after a packets is passed, if the network has not passed packets for some time, the timeout timer will tick down to zero, and the sequence recovery function will resets the algorithm, and the next packet received will be accepted. Latent error path is used for latent error detection function, it represents the path number of the network, if latent error path is n, then every packet will be duplicated for n times at the talker system T and sent separately through different path to listener system L. Maximum latent error difference is used for latent error detection function, it represents the maximum allowed difference of the expected and actual discarded packet number, if difference of the expected and actual discarded packet number exceeds this parameter, the latent error will be indicated.

Table 3. Simulation parameters of IEEE P802.1 CB network

SequenceHistoryLength	100
Timeout timer	10ms
Latent error path	2
Maximum latent error difference	10

The simulation parameters of the packets transmitted in the network are listed in Table 4, they are transmitted from the talker system T through path A or path B to the listener system L with different sequence number. For the following simulation they will be transmitted in different order and for different times to test the seamless redundancy character.

Table 4. Simulation parameters of packets

Packet	Source node	Destination node	Path	Sequence number
1	T	L	A	25
2	T	L	A	65450
3	T	L	A	130
4	T	L	A	15
5	T	L	B	25
6	T	L	B	15
7	T	L	A	30

CppUTest is a C /C++ based unit test framework for unit testing and for test-driving code, in this paper 7 situations of IEEE P802.1 CB are tested:

1) Test if the network can pass the first packet. Transmit Packet 1 from node T to node L, since Packet 1 is the first packet transmitted in the network, so the test result should show that packet 1 is passed.

2) Test if the network can discard the packet out of the sequence recovery window. Transmit Packet 1, Packet 2 and Packet 3 in turn from node T to node L, since the maximum received sequence number is 25 after Packet 1 has been received, and the difference of the sequence number of Packet 2 and 25 is larger than SequenceHistoryLength, so the test result should show that Packet 2 is discarded; Packet 3 should be discarded for the same reason.

3) Test if the network can pass the frame with sequence number in (MaxSeqNum- SequenceHistoryLength) window, and has not been seen before. Transmit Packet 1 and Packet 4 in turn from node T to node L, since the maximum received sequence number is 25 after Packet 1 has been

received, the difference of the sequence number of Packet 4 and 25 is within SequenceHistoryLength, the sequence number of Packet 4 is smaller than 25 and has not been seen before, so the test result should show that Packet 4 is passed.

4) Test if the network can discard the frame with sequence number in (MaxSeqNum- SequenceHistoryLength) window, but has been seen before. Transmit Packet 1, Packet 4, Packet 5 and Packet 6 in turn from node T to node L, since the maximum received sequence number is 25 after Packet 1 and Packet 4 have been received, the difference of the sequence number of Packet 5 and 25 is within SequenceHistoryLength, the sequence number of Packet 5 is smaller than 25 but has been seen before, so the test result should show that Packet 5 is discarded; Packet 6 should be discarded for the same reason.

5) Test if the network can pass the frame with sequence number in (MaxSeqNum+ SequenceHistoryLength) window. Transmit Packet 1 and Packet 7 in turn from node T to node L, since the maximum received sequence number is 25 after Packet 1 has been received, the difference of the sequence number of Packet 7 and Packet 1 is within SequenceHistoryLength, and the sequence number of Packet 7 is larger than 25, so the test result should show that Packet 7 is passed.

6) Test if the timeout mechanism can reset the sequence recovery algorithm and pass the packet again, when no packets has been passed, and the timeout timer ticks down to 0. First transmit Packet 1 from node T to node L, reset the timeout timer, then transmit Packet 2 from node T to node L until timeout timer ticks down to 0, check if Packet 2 can be passed when the timeout event occurs.

7) Test if the latent error signal is issued during packets transmission from node T to node L. Transmit packets from node T to node L, detect latent error every second: calculate the difference between (Passed Packets * (LatentErrorPaths - 1)) and (Discarded Packets), check if the difference is no more than (Latent error difference).

The test result is shown in Figure 9, which indicates that all test cases are passed, and IEEE P802.1 CB can realize seamless redundancy efficiently.

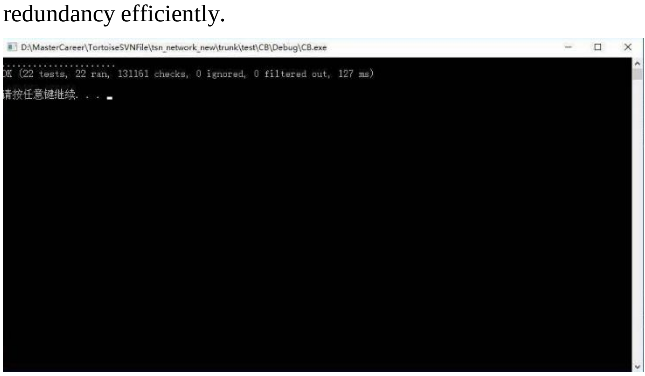

Figure 9. Test Result of IEEE P802.1 CB

4. Conclusions

This paper introduces the mechanism with which IEEE P802.1 CB realizes seamless redundancy for time-sensitive networking, including network topology, redundancy tag, sequence generation function, sequence recovery function, timeout mechanism and latent error detection. A network complying IEEE P802.1 CB is simulated and 7 situations are tested to demonstrate that IEEE P802.1 CB can provide seamless redundancy. In conclusion IEEE P802.1 CB is a redundancy method that:

1) Can be used with any topology;
2) Can realize seamless redundancy for the network;
3) Achieves zero delay switchover making it suitable for time-critical system;
4) Dose not rely on higher layer protocols;
5) Is transparent to the application.

5. Future work

This paper has analyzed the operating principle of IEEE P802.1 CB, a simple network complying IEEE P802.1 CB is simulated and tested to provide research basis for future work. In this paper only a network with 2 paths as illustrated in Figure 3 is simulated, and the simulation parameter is fixed, in the future networks with more path and cross links as illustrated in Figure 1 should be simulated and their redundancy character should be compared and analyzed, the simulation parameters should be adjusted to achieve the best redundancy character and the lowest cost.

6. ACKNOWLEDGMENTS

This work was financially supported by Shanghai Automotive Industry Science and Technology Development Foundation (1515).

7. REFERENCES

[1] Sridharan K, Goossens KK, Concer N, et al. Investigation of Time-Synchronization over Ethernet In-Vehicle Networks for automotive applications: [Master Thesis]. Eindhoven: Eindhoven University of Technology, 2015

[2] Zeng W, Khalid M, Chowdhury S. A qualitative comparison of FlexRay and Ethernet in vehicle networks. in: eds. Electrical and Computer Engineering (CCECE), 2015 IEEE 28th Canadian Conference on. IEEE, 2015. 571-576

[3] Steinbach T, Müller K, Korf F, et al. Demo: Real-time Ethernet in-car backbones: First insights into an automotive prototype. in: eds. 2014 IEEE Vehicular Networking Conference (VNC). IEEE, 2014. 133-134

[4] [8] Thiele D, Schlatow J, Axer P, et al. Formal timing analysis of CAN-to-Ethernet gateway strategies in automotive networks. Real-time systems, 2016, 52 (1): 88-112

[5] Alderisi G, Iannizzotto G, Bello LL. Towards IEEE 802.1 Ethernet AVB for advanced driver assistance systems: A preliminary assessment. in: eds. Proceedings of 2012 IEEE 17th International Conference on Emerging Technologies & Factory Automation (ETFA 2012). IEEE, 2012. 1-4

[6] Yoo I, Hwang M, Jung J, et al. Unidirectional ring ethernet for low-complexity in-vehicle control network. in: eds. Industrial Technology (ICIT), 2015 IEEE International Conference on. IEEE, 2015. 1951-1955

[7] Nsaif, S.A.; Rhee, J.-M. DVP: A Novel High-Availability Seamless Redundancy (HSR) Protocol Traffic-Reduction Algorithm for a Substation Automation System Network. Energies 2014, 7, 1792–1810.

Columns on Last Page Should Be Made As Close As Possible to Equal Length

Authors' background

Your Name	Title*	Research Field	Personal website
Shifei Qian	master student	automotive network	/
Feng Luo	full professor	automotive network	/
Jinpeng Xu	Phd candidate	automotive network	/

***This form helps us to understand your paper better, the form itself will not be published.**

***Title can be chosen from: master student, Phd candidate, assistant professor, lecture, senior lecture, associate professor, full professor**