MICROCHIP

AN3474

KSZ9477 High-Availability Seamless Redundancy

Author: *Brigham Steele*
Microchip Technology Inc.

INTRODUCTION

This application note introduces the concept of High-availability Seamless Redundancy (HSR), explains how it works, and aims to offer guidance and reference on how to implement it with the KSZ9477 Ethernet switch. This document is intended for users familiar with the HSR protocol and Microchip KSZ9477. It mainly focuses on a single-ring network scenario and does not cover connected-ring scenarios. The performance measurement data showed in this application note is based on EVB-KSZ9477 where an external processor, SAMA5D3, manages the switch.

SECTIONS

This application note covers the following sections:

- General Information
- HSR Support
- Hardware HSR
- HSR System Implementation
- KSZ9477 Chip Limitations
- Processor Selection

REFERENCES

Refer to following documents when using this application note:

- *KSZ9477S 7-Port Gigabit Ethernet Switch with Ring Redundancy, SGMII and RGMII/MII/RMII Interfaces Data Sheet*
- IEC 62439-3 Clause 5

GENERAL INFORMATION

Redundancy Options

To protect against communication failure, some Layer 2 redundancy protocols, such as STP/RSTP, Link Aggregation, and DLR, etc. have been developed. All these protocols are normally second/sub-second recovery and do not provide zero-packet loss during switchover.

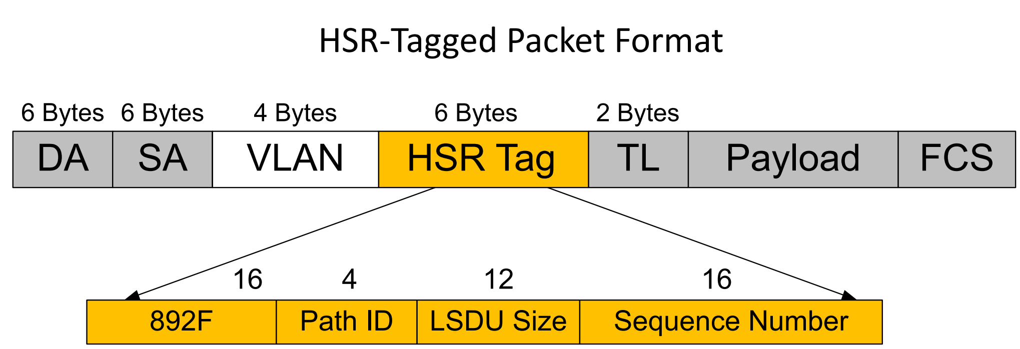
HSR Overview

High-availability Seamless Redundancy is a standard defined in IEC 62439-3 Clause 5 targeting applications that require short reaction time and high availability.

The typical HSR topology is a ring. The source node duplicates all the frames it wants to send. It sends the frames using two different paths to their destination. In case of one network component failure, such as a link or a node failure, the frames will still reach their destination.

Frames forwarding in the ring carry the HSR tag inserted by the source node, which contains a sequence number. The doublet of source MAC address and sequence number uniquely identifies copies of the same frame. See Figure 1.

FIGURE 1: HSR-TAGGED PACKET FORMAT



HSR Advantages

HSR is a low-cost, zero-recovery redundancy protocol. This feature makes it highly suitable for real-time and mission-critical applications. See Table 1.

TABLE 1: REDUNDANCY PROTOCOLS COMPARISON

Protocol	Switch-Over Time	Comments
Link Aggregation	Less than 50 ms	Only for link failure protection
Spanning Tree	Several seconds	Commonly used but has the largest fail-over time (**Note 1**)
Device Level Ring	Less than 50 ms	Use beacon packets to detect failure.
Parallel Redundancy Protocol	Zero	Double the switch, cable, and data. This is mostly used in star topology.
HSR	Zero	Double data. No single point of failure. This is mostly used in ring topology.

Note 1: Do not enable Spanning Tree Protocol (STP) on the HSR ring ports.

HSR SUPPORT

Software HSR Implementation

In the software-based HSR implementation, the HSR tag needs to be inserted in every frame sent out. These frames are also duplicated and sent to two ports. When the frame is received, the software needs to check duplicates and drop them. Because of this, the software has to maintain a database for received frames within a certain time.

Hardware HSR Implementation

In contrast with pure software implementation, hardware-based HSR implementation allows offloading of some CPU work, such as HSR frame duplication, hardware frame forwarding, and HSR sequence number checking, among others.

Switches that can Implement Hardware HSR

The Microchip KSZ9477 is a highly recommended Ethernet switch for developing the hardware-based HSR.

HARDWARE HSR

HSR Principle

Below are the two node types used in HSR:

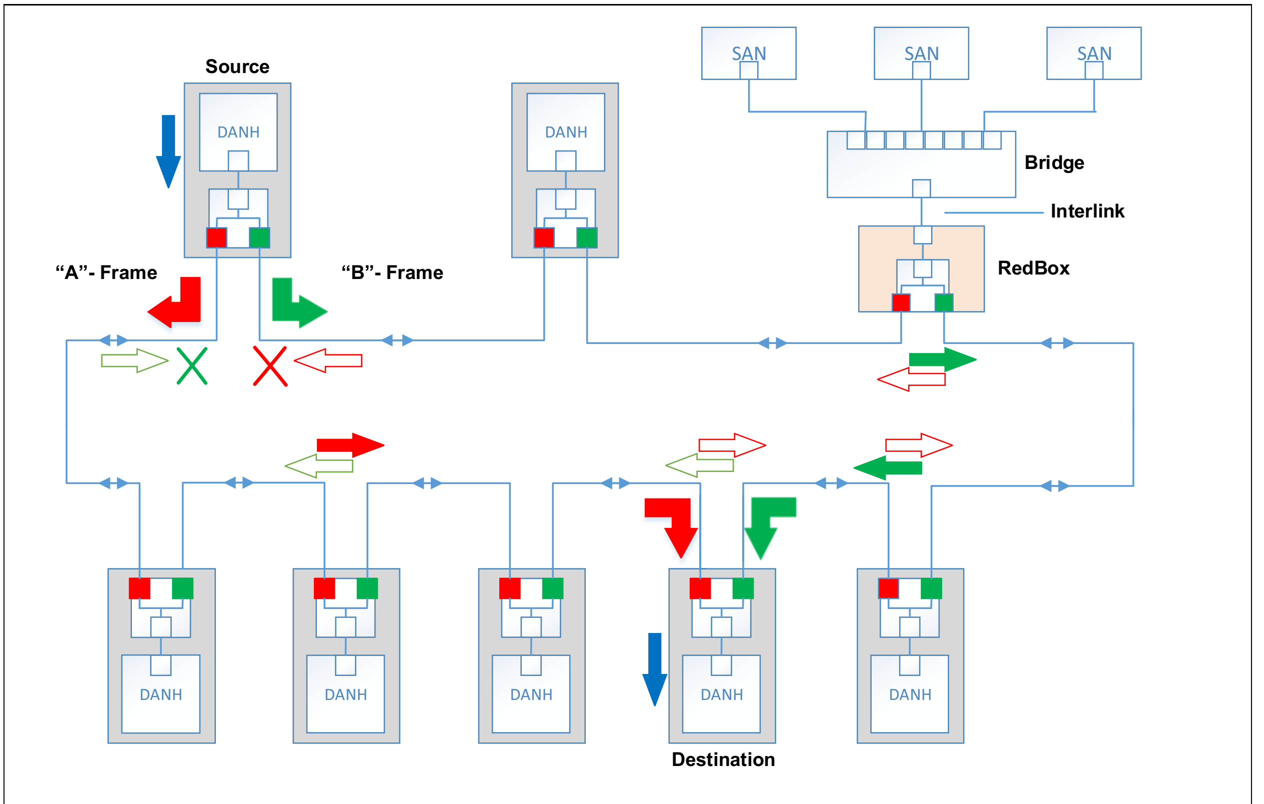
1. **Doubly Attached Node for HSR (DANH)**: A **DANH** has two Ethernet ports to connect to one HSR ring. When sending frames, DANH-node sends a duplicate of each frame into the network, one to each of the directions in the ring. When receiving, it accepts the first copy and discards the second, thus eliminating the duplicate.
2. **Redundancy Box (RedBox)**: RedBox is an entity that has three Ethernet ports. Two ports are connected to an HSR ring, and one port is a traditional Ethernet port. RedBoxes are used to connect non-HSR nodes and non-HSR network segments (Single Attached Nodes or SAN) to HSR networks. RedBoxes forward the frames over the ring-like DANH nodes and act as proxies for all SANs that access them.

Frame Forwarding In HSR Network

UNICAST FRAME FORWARDING DECISION

When a unicast packet reaches the destination node, it is consumed by the receiver and not forwarded. If a unicast frame does not match any node in the ring, it will be dropped and stopped being forwarded by one middle-node with the duplicate frame discard function enabled. In case it is forwarded back to the originating node, it is going to be dropped because the source MAC address matches the node address. See Figure 2.

FIGURE 2: UNICAST FRAME FORWARDING DECISION



Note 1: Solid arrows: red = "A" - Frame, green = "B"- Frame, and blue = Non-HSR Format Frame

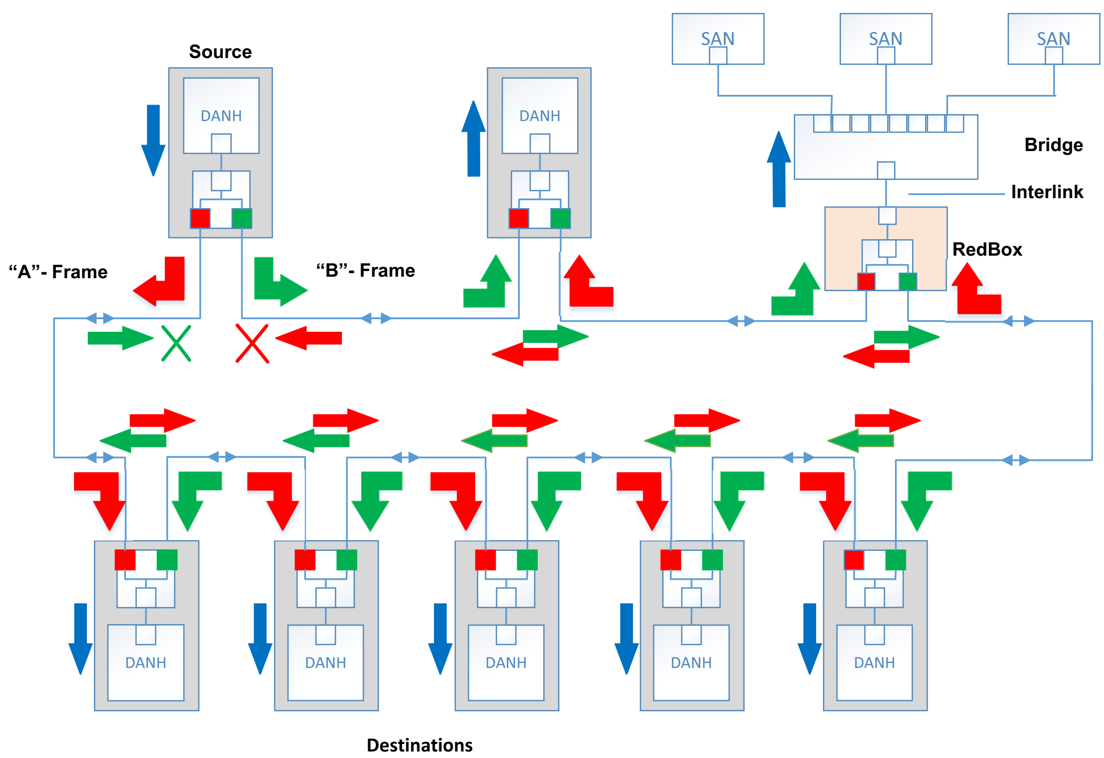
2: Void arrows = unicast frame not being received

3: Cross = Frame removed from the ring

MULTICAST/BROADCAST FRAME FORWARDING DECISION

A multicast/broadcast packet is forwarded by each node because there could be multiple potential receivers. Usually, a multicast/broadcast packet is dropped and stopped being forwarded by one middle-node with duplicate frame discard function enabled. If somehow a multicast frame is forwarded back to the originating node, it is dropped because the source MAC address matches the node address. See Figure 3.

FIGURE 3: MULTICAST/BROADCAST FRAME FORWARDING DECISION



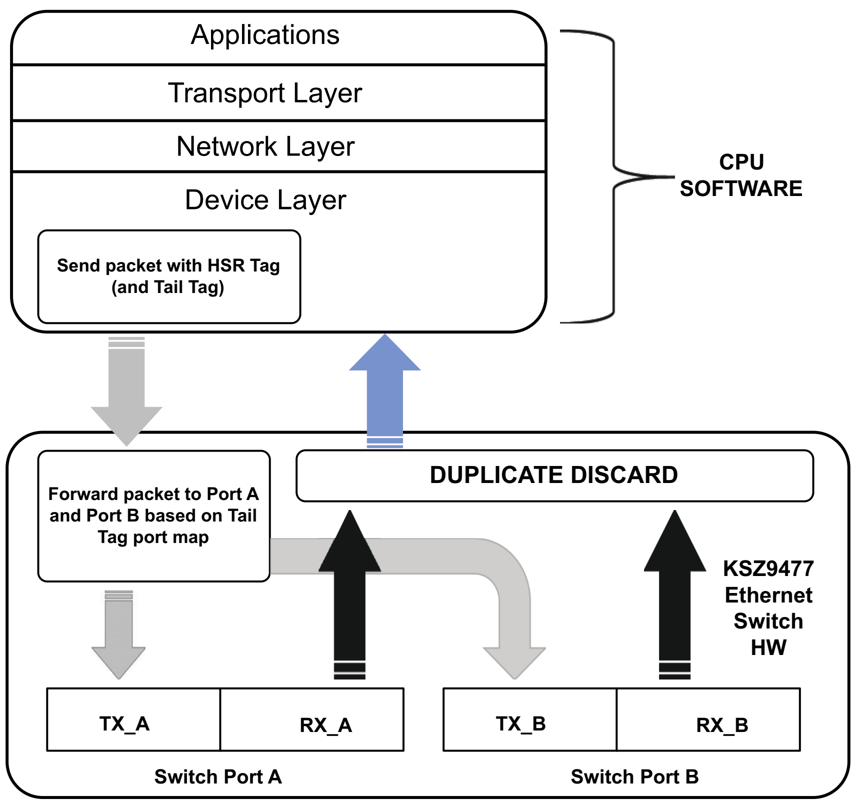
Note 1: Solid arrows: red = "A" - Frame, green = "B"- Frame, and blue = Non-HSR Format Frame

2: Cross = Frame removed from the ring

KSZ9477 Implementation Algorithm and Programming

Figure 4 shows how HSR is implemented with KSZ9477. In this process, the CPU software is required to attach the HSR tag to a packet and send it to both A and B ports with the tail tag mechanism. The KSZ9477 hardware forwards the packet to both port A and port B based on the egress tail tag port map (tail tag will be removed and will not be sent out on the wire). While receiving, KSZ9477 provides the hardware a duplicate discard functionality.

FIGURE 4: OVERVIEW OF THE KSZ9477 HSR IMPLEMENTATION



SELECTING TWO PORTS TO PARTICIPATE IN THE HSR RING

Any two ports of the KSZ9477 may be used as ports for participation in the HSR ring. In a related registers setting, HSR ports are selected by setting two bits, as appropriate, in the HSR Port Map register. For instance, to select port 1 and port 2, bit[6:0] should be set as 0x3.

Note:	The HSR Port Map register is required to be set once regardless of its default value.

TX PACKET DUPLICATION FROM HOST TO SWITCH

All frames in an HSR network are generated by the CPU software, including the HSR tag and sequence numbers. Tail tagging must be utilized for the CPU to indicate the two destination ports for each generated frame. It is a method that communicates ingress and egress port information between the CPU and the switch. Tail tagging is useful for spanning tree protocol, IGMP/MLD snooping, IEEE 1588, and other applications. The tail tag is inserted at the end of the packet, between the payload and the 4-byte CRC/FCS.

To enable tail tagging in a related registers setting, set the Tail Tag Enable bit in a Port Operation Control 0 register at address 0xN020 for port “N”. When this bit is set for one port, that port is referred to as the “host” port. Note that tail tagging applies only to the host port and never to any other ports of the switch. Then, in the two tail tag bytes (which refer to Transmit Tail Tag Format in the *KSZ9477S Data Sheet*), the host processor adds to each packet and determine the egress ports with bits [6:0]. The Lookup bit 10 should not be set.

RX PACKET DUPLICATION DISCARDING

Hardware-assisted duplicate frame discard is implemented in KSZ9477 by utilizing a two-way set-associative on-chip memory with 512 entries for storing and managing variables relating to received frames. Entries are indexed by a combination of the source and destination addresses as reduced by a hash function. Tracking is performed independently for each of the two ring ports. For each received frame, the HSR sequence number is extracted and compared to values in the table. If a matching frame has already been received on the other port, the frame is dropped. If not, then standard forwarding rules apply.

In a related registers setting, duplicate discard function is enabled by setting bit 7 in the Global HSR AME Control 0 register (0x0644). Meanwhile, bit 6 in the Global HSR AME Control 0 register needs to be turned off.

PREVENTING PACKET LOOP IN THE RING BY SELF-ADDRESS FILTERING

Self-address filtering is used to ensure that frames cannot traverse the ring more than once. When this feature is enabled, the source address of all received frames is compared to the node's own MAC address. If there is a match, the frame will not be forwarded and will be discarded.

In a related registers setting, self-address filtering can be enabled for all ports by setting bit 6 in the Switch Lookup Engine Control 1 register. Alternatively, it can be enabled on a per-port basis by setting bit 3 in the Port Control 2 register. Both this port enable bit and the global enable bit must be set to activate self-address filtering. The local MAC address is programmed in Global Switch MAC Address registers (Switch MAC Address 0 register through Switch MAC Address 5 register).

HSR SYSTEM IMPLEMENTATION

DANH-Node or RedBox

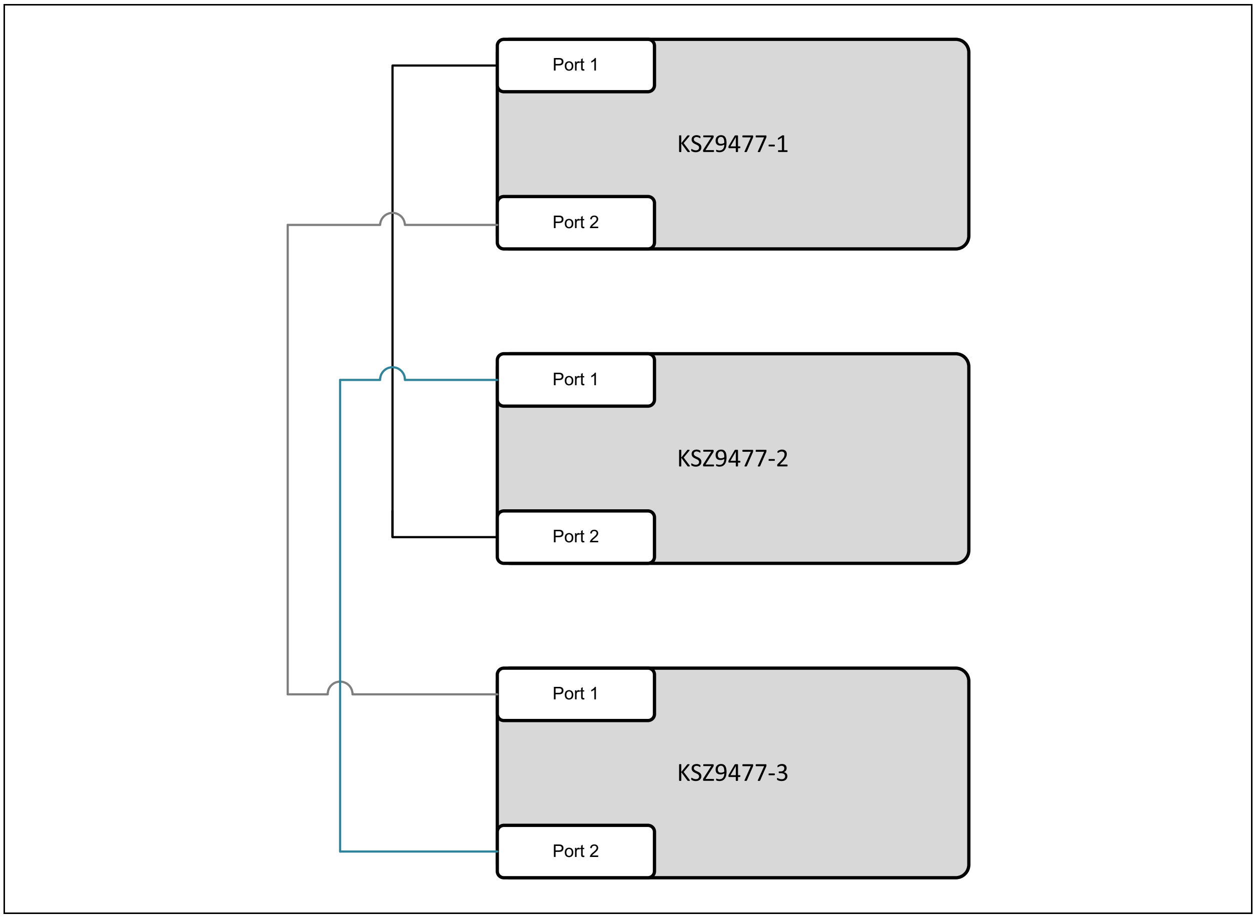
The EVB-KSZ9477 can operate as both DANH-node and RedBox in a typical HSR ring topology. Below is an HSR performance report of the EVB-KSZ9477.

THE EVB-KSZ9477 OPERATING AS DANH-NODE

Test Topology

The operation of EVB-KSZ9477 as a DANH-NODE in a ring topology is illustrated in Figure 5.

FIGURE 5: EVB-KSZ9477 AS DANH-NODE TEST TOPOLOGY



Device Configuration

All devices have almost the same configuration except for IP address/MAC address. On each device, run the commands in Figure 6 for the Uboot prompt:

FIGURE 6: COMMANDS FOR UBOOT PROMPT

```
setenv multi_dev 0
setenv eth1_ports 3
setenv eth1_proto hsr
setenv eth1_vlan 0x7e
saveenv
boot
```

Test Design

In this scenario, all frames originate from the SAMA5D3 processor. Use the Iperf3 tool to simulate TX/RX process of real application traffic. In destination node KSZ9477-3, run the `#iperf3 -s` command to operate as a server.

In source node KSZ9477-1 in Figure 5, iperf operates as a client, and UDP is used as a traffic type to measure the bandwidth. See Table 2.

TABLE 2: TEST RESULTS OF EVB-KSZ9477 OPERATING AS DANH-NODE

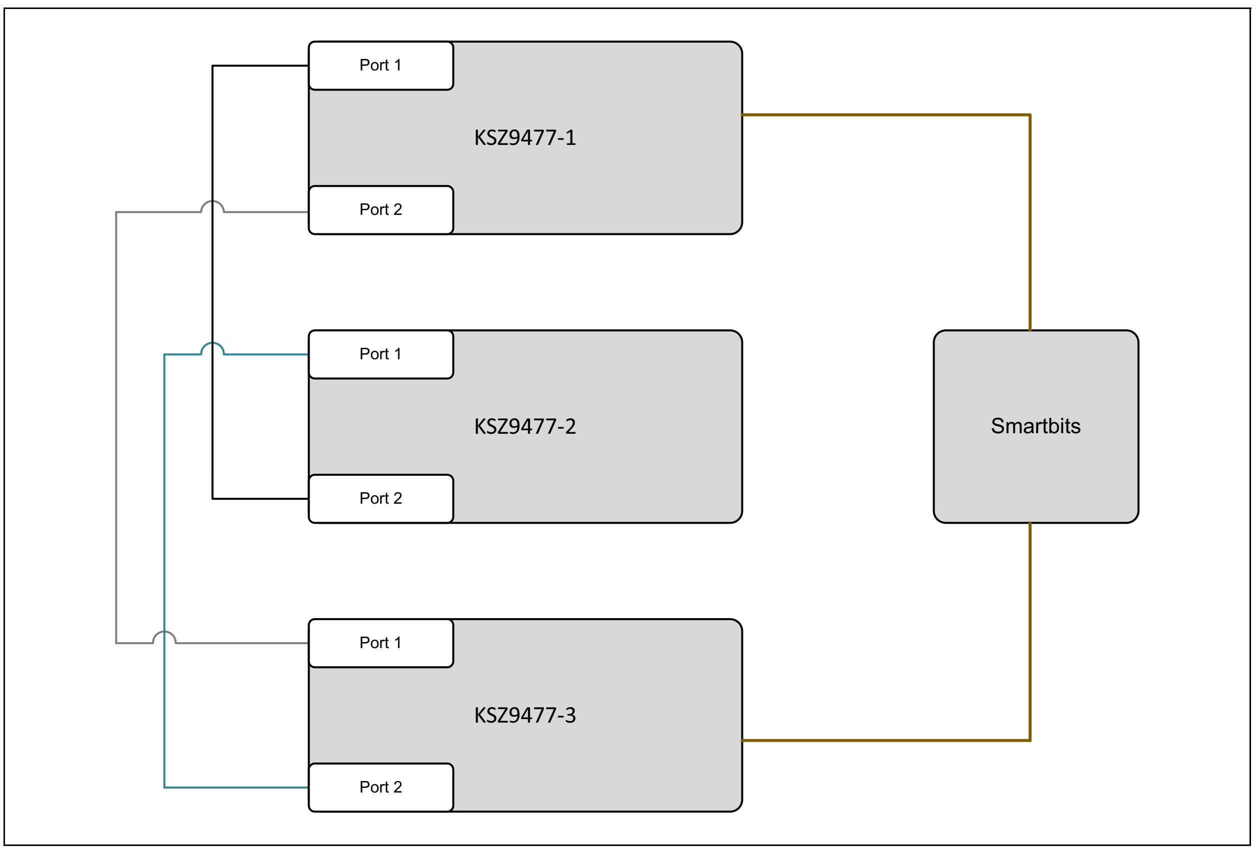
UDP Buffer Length	Ethernet Packet Size (including 6 bytes HSR tag, without FCS)	Maximum Bandwidth	Packet per Second	CPU Usage
16	64	776 Kbits/sec	6062 pps	Close to 100%
80	128	3.98 Mbits/sec	6221 pps	Close to 100%
208	256	9.96 Mbits/sec	5983 pps	Close to 100%
464	512	22.2 Mbits/sec	5971 pps	Close to 100%
976	1024	45.1 Mbits/sec	5778 pps	Close to 100%
1464	1512	67.1 Mbits/sec	5730 pps	Close to 100%
8192	8240	124 Mbits/sec	1885 pps	Close to 100%

THE EVB-KSZ9477 OPERATING AS REDBOX

Test Topology

The operation of EVB-KSZ9477 as a RedBox in a ring topology is shown in Figure 7.

FIGURE 7: EVB-KSZ9477 AS REDBOX TEST TOPOLOGY



Device Configuration

All devices have almost the same configuration except for IP address/MAC address. On each device, run the following commands on the Uboot prompt. See Figure 8.

FIGURE 8: DEVICE CONFIGURATION COMMANDS

```
setenv multi_dev 1
setenv eth1_ports 3
setenv eth1_proto hsr
setenv eth1_vlan 0x7e
setenv eth2_proto redbox
setnev eth2_vlan 0x7f
saveenv
boot
```

Test Design

Smartbits are used to simulate SANs (Singly Attached Nodes), which rely on RedBox to work with HSR networks. See Table 3.

TABLE 3: TEST RESULTS OF EVB-KSZ9477 OPERATING AS REDBOX

Packet Size	TX Speed	Maximum Throughput	CPU Usage
64 bytes	1000 Mbps	0.936%, 9.36Mbps	Close to 100%
128 bytes	1000 Mbps	1.55%, 15.5Mbps	Close to 100%
256 bytes	1000 Mbps	2.86%, 28.6Mbps	Close to 100%
512 bytes	1000 Mbps	5.25%, 52.5Mbps	Close to 100%
1024 bytes	1000 Mbps	9.58%, 95.8Mbps	Close to 100%
1500 bytes	1000 Mbps	12.99%, 129.9Mbps	Close to 100%

KSZ9477 CHIP LIMITATIONS

- **Software is required to implement the duplicate discard function when the number of nodes is even and the HW based duplicate discard is enabled in the switch.**

  The switch has to know that a packet has come before it can determine that a duplicate happened the next time the same packet arrives again. The arrival of the first packet is recognized at the end of its transmission, while the second packet is looked up just after the VLAN tag is received. For instance, the duplicate packet arrives before the first packet is completely received (so the recognition of the first packet has not happened yet). Given this situation, the switch does not have the record to remove the duplicated packet. This only happens when these two packets overlap inside the switch. Therefore, the difference between the arrival times of the two packets would probably be either in ns or µs.

  For example, a 64-byte packet has the packet receiving time of (8 + 64) x 80 ns = 5.76 µs for 100 Mbps port. The first packet arrived. If the first 24 bytes of the duplicate packet arrive before the previous packet is completely received, the first packet cannot be discarded. In this case, the second packet needs to arrive at least (8 + 64 – 24) x 80 ns = 3.84 µs later than the first packet's arrival. There is a shorter time required in Gigabit port (1/10 of it), but this depends on the length of the packet.

  This scenario happens when both the original packet and the duplicate packet enter the switch at about the same moment (with only ns-to-µs level difference), so the device cannot handle it. It should not happen frequently in network rings of more than 2 nodes. With the duplicate discard function enabled in the switch, it will remove most of the duplicate packets, and the software just needs to handle the rest of the packets.

- **Some MAC controllers do not understand the HSR tag, so hardware checksum generation may fail and the function has to be disabled. This affects the network performance for UDP and especially TCP packets when running on Linux®.**

PROCESSOR SELECTION

When a device such as EVB-KSZ9477 operates as DANH in the TX direction, all application traffic originates from the CPU. The CPU is also expected to add an HSR tag on all application traffic. In the RX direction, the CPU is responsible to strip the HSR tag before delivering traffic back to application layer.

When a device like EVB-KSZ9477 operates as RedBox, the CPU plays a traffic relay role between the Non-HSR network and HSR network. In particular, the CPU relays traffic in a software bridging manner. The CPU also handles the work of adding and removing the HSR tag.

Therefore, both conditions above show that the HSR bandwidth performance is limited to the capabilities of the CPU processor, so choosing the most suitable processor is crucial to achieving optimum results.

APPENDIX A: APPLICATION NOTE REVISION HISTORY

TABLE A-1: REVISION HISTORY

Revision Level & Date	Section/Figure/Entry	Correction
DS00003474A (05-20-2020)	Initial Release	

NOTES:

THE MICROCHIP WEBSITE

Microchip provides online support via our WWW site at www.microchip.com. This website is used as a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the website contains the following information:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQ), technical support requests, online discussion groups, Microchip consultant program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

CUSTOMER CHANGE NOTIFICATION SERVICE

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip web site at www.microchip.com. Under "Support", click on "Customer Change Notification" and follow the registration instructions.

CUSTOMER SUPPORT

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support

Customers should contact their distributor, representative or Field Application Engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document.

Technical support is available through the web site at: http://microchip.com/support

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Trademarks

ISBN: 978-1-5224-6123-4

For information regarding Microchip's Quality Management Systems, please visit www.microchip.com/quality.

Worldwide Sales and Service

AMERICAS

Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
Technical Support:
http://www.microchip.com/support
Web Address:
www.microchip.com

Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Austin, TX
Tel: 512-257-3370

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Novi, MI
Tel: 248-848-4000

Houston, TX
Tel: 281-894-5983

Indianapolis
Noblesville, IN
Tel: 317-773-8323
Fax: 317-773-5453
Tel: 317-536-2380

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608
Tel: 951-273-7800

Raleigh, NC
Tel: 919-844-7510

New York, NY
Tel: 631-435-6000

San Jose, CA
Tel: 408-735-9110
Tel: 408-436-4270

Canada - Toronto
Tel: 905-695-1980
Fax: 905-695-2078

ASIA/PACIFIC

Australia - Sydney
Tel: 61-2-9868-6733

China - Beijing
Tel: 86-10-8569-7000

China - Chengdu
Tel: 86-28-8665-5511

China - Chongqing
Tel: 86-23-8980-9588

China - Dongguan
Tel: 86-769-8702-9880

China - Guangzhou
Tel: 86-20-8755-8029

China - Hangzhou
Tel: 86-571-8792-8115

China - Hong Kong SAR
Tel: 852-2943-5100

China - Nanjing
Tel: 86-25-8473-2460

China - Qingdao
Tel: 86-532-8502-7355

China - Shanghai
Tel: 86-21-3326-8000

China - Shenyang
Tel: 86-24-2334-2829

China - Shenzhen
Tel: 86-755-8864-2200

China - Suzhou
Tel: 86-186-6233-1526

China - Wuhan
Tel: 86-27-5980-5300

China - Xian
Tel: 86-29-8833-7252

China - Xiamen
Tel: 86-592-2388138

China - Zhuhai
Tel: 86-756-3210040

ASIA/PACIFIC

India - Bangalore
Tel: 91-80-3090-4444

India - New Delhi
Tel: 91-11-4160-8631

India - Pune
Tel: 91-20-4121-0141

Japan - Osaka
Tel: 81-6-6152-7160

Japan - Tokyo
Tel: 81-3-6880- 3770

Korea - Daegu
Tel: 82-53-744-4301

Korea - Seoul
Tel: 82-2-554-7200

Malaysia - Kuala Lumpur
Tel: 60-3-7651-7906

Malaysia - Penang
Tel: 60-4-227-8870

Philippines - Manila
Tel: 63-2-634-9065

Singapore
Tel: 65-6334-8870

Taiwan - Hsin Chu
Tel: 886-3-577-8366

Taiwan - Kaohsiung
Tel: 886-7-213-7830

Taiwan - Taipei
Tel: 886-2-2508-8600

Thailand - Bangkok
Tel: 66-2-694-1351

Vietnam - Ho Chi Minh
Tel: 84-28-5448-2100

EUROPE

Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

Denmark - Copenhagen
Tel: 45-4485-5910
Fax: 45-4485-2829

Finland - Espoo
Tel: 358-9-4520-820

France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

Germany - Garching
Tel: 49-8931-9700

Germany - Haan
Tel: 49-2129-3766400

Germany - Heilbronn
Tel: 49-7131-72400

Germany - Karlsruhe
Tel: 49-721-625370

Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Germany - Rosenheim
Tel: 49-8031-354-560

Israel - Ra'anana
Tel: 972-9-744-7705

Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

Italy - Padova
Tel: 39-049-7625286

Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

Norway - Trondheim
Tel: 47-7288-4388

Poland - Warsaw
Tel: 48-22-3325737

Romania - Bucharest
Tel: 40-21-407-87-50

Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

Sweden - Gothenberg
Tel: 46-31-704-60-40

Sweden - Stockholm
Tel: 46-8-5090-4654

UK - Wokingham
Tel: 44-118-921-5800
Fax: 44-118-921-5820