KSZ9897 Switch Reference Guide

Rev 1.2

July 26, 2017

2180 Fortune Dr., San Jose, CA 95131, USA • (408) 944-0480 • http://www.microchip.com

Table of Contents

1 Revision History

Revision	Date	Summary of Changes
1.0	10/28/14	Initial revision.
1.1	04/06/16	Updated for S2 revision.
1,2	07/26/17	Updated for more information.

2 Introduction

This document describes how to use Microchip KSZ9897 family switch in software. The switch is used in the KSZ9567, KSZ9566, and KSZ9477 chips. The software implementation is done in a Linux driver. The driver uses sysfs to provide an interface so that users can access the switch without using any application.

Each section describes a feature in the switch. Inside the section is **Register Description** where registers related to the feature are displayed. They are copied from the KSZ9567/KSZ9566/KSZ9477 datasheet. Refer to that datasheet for more information.

After **Register Description** is **Sysfs Variables** where variables related to the feature are described. The sysfs variables are attached to the device created by the driver. For a network devices they are located in the directory `/sys/class/net/eth?`. For a SPI devices they are located in the directory `/sys/bus/spi/devices/spi?`.

The main global variables are located under the subdirectory `sw`. Variables from each switch port is located under the subdirectory `sw[port]` where `[port]` is 0, 1, 2, and up to 6. Because of the way the sysfs variables is implemented in the driver, all variables under a port subdirectory is preceded with `[#]_` where `[#]` is the same as the port number.

For those subdirectories with an index value only the maximum index is shown. In the case of switch ports it implies some variables, and so some functions, are not available in port 7, the host port. Note the last and host port of KSZ9566 is port 6.

The sysfs file is read by using the "`cat`" command. It is written with the "`echo "" >`" command. Most the files just return on/off status with "0" meaning off and "1" meaning on. For other commands do an "`echo 1 > sw/info`" command first to turn on verbose mode so that more detailed information are provided.

2.1 Register Description

Chip ID Register (0x0000 – 0x0004)

Bit	Default	R/W	Description
23-16	0x95, 0x98	RO	**Family ID**
15-8	0x67, 0x66	RO	**Chip ID**
7-4	0x0	RO	**Revision ID**
0	0	RW	**Software Reset**

The register range is 0x0000 – 0xFFFF. The 0xN000 – 0xNFFF range is used by each port where N is 1 for port 1, 2 for port 2, and so on.

Chip Global Options Register (0x000F)

Bit	Default	R/W	Description
6	0	RO	**Gigabit Capable**
5	0	RO	**Redundancy Capable (DLR, HSR)**
4	0	RO	**AVB Capable (1588 PTP)**

This register is used to indicate the capabilities of the switch. Depending of models the capabilities are different.

2.2 Sysfs Variables

`sw/reg`

The `reg` variable is used to access registers in the switch. Writing a "reg" value in hexadecimal returns the value in the register. Writing "reg=val" in hexadecimal writes the value to the register. It is used primarily to verify the registers are programmed correctly.

This is just a quick and easy way to read a register to verify something. For regular use it is recommended to use the `regs_bin` utility to read/write registers.

3 Tables

The switch has some internal tables that can only be accessed indirectly using a set of fixed registers. Those tables are Dynamic MAC Address Table, Static MAC Address Table, VLAN Table, and MIB Counters.

3.1 Register Description

MIB Counters

Switch MAC Control Register 6 (0x0336)

Bit	Default	R/W	Description
7	0	SC	**Flush MIB Counters**

6	0	RW	**Freeze MIB Counters**

Port MIB Control Register (0xN500 – 0xN5003)

Bit	Default	R/W	Description
31	0	RO	**Counter Overflow**
25	0	RW	**Read Enable or Counter Valid**
24	0	RW	**Flush and Freeze Enable**
23-16	0x00	RW	**MIB Index**
3-0	0x0	RO	**Counter Value [35:32]**

Port MIB Data Register (0xN504 – 0xN507)

Bit	Default	R/W	Description
31-0	0x00000000	RO	**Counter Value [31:0]**

The MIB counters are cleared after read, so software needs to keep track of them.

Static MAC Address Table (0x0 – 0xF)

Dynamic MAC Address Table (0x000 – 0x3FF)

Switch ALU Index Register 0 (0x0410 – 0x0413)

Bit	Default	R/W	Description
31-16	0	RW	**FID Index**
15-0	0	RW	**MAC Index [47:32]**

Switch ALU Index Register 1 (0x0414 – 0x0417)

Bit	Default	R/W	Description
31-0	0	RW	**MAC Index [31:0]**
11-0	0	RW	**Direct Address**

Switch ALU Control Register (0x0418 – 0x041B)

Bit	Default	R/W	Description
29-16	0	RO	**Valid Count**

7	0	SC	**Start**
6	0	RW	**Valid**
2	0	RW	**Direct Access**
1-0	0	RW	**Action**

Switch Static ALU Control Register (0x041C – 0x041F)

Bit	Default	R/W	Description
31-16	0	RW	**ALU Index**
7	0	RW	**Start**
1	0	RW	**Access Reserved Multicast Map**
0	0	RW	**Action**

Switch ALU Value A Register (0x0420 – 0x0423)

Bit	Default	R/W	Description
31	0	RO	**Static**
31	0	RW	**Valid**
30	0	RW	**Source Filtering**
29	0	RW	**Destination Filtering**
28-26	0	RW	**Priority or Age Count**
2-0	0	RW	**Multiple Spanning Tree Number**

Switch ALU Value B Register (0x0424 – 0x0427)

Bit	Default	R/W	Description
31	0	RW	**Override**
30	0	RW	**Use Filter ID**
23-0	0	RW	**Forward Ports**

Switch ALU Value C Register (0x0428 – 0x042B)

Bit	Default	R/W	Description
31-16	0	RW	**Filter ID**
15-0	0	RW	**MAC Address [47:32]**

Switch ALU Value D Register (0x042C – 0x042F)

Bit	Default	R/W	Description
31-0	0	RW	**MAC Address [31:0]**

Static MAC table and dynamic MAC table use the same ALU Value Registers to pass information.

There are 16 entries in the static MAC table. In addition there is a reserved multicast MAC table so that some commonly used multicast addresses need not be defined in the static MAC table. There are 32 entries but they are grouped in 6 groups.

The MAC Address specifies the MAC address for lookup.

The Forward Ports indicates where to forward the frames. Bit 0 is port 1, bit 1 is port 2, and so on. Setting 0 means no forwarding, while setting 0x7f means broadcasting.

The Valid bit is used to indicate the entry is valid. Setting 0 is like deleting the entry.

The Override bit is used to override the port transmit and receive disable settings.

The Use FID bit is used to specify FID+MAC for lookup.

The FID specifies the Filter ID if FID+MAC is used.

The Static ALU Control Register is used to access the static MAC table.

The ALU Index specifies the index.

The Access Reserved Multicast Map bit specifies accessing the reserved multicast MAC table.

The Action code is 0 for write and 1 for read.

The Start bit executes the command.

There can be up to 1048 entries in the dynamic MAC table.

The Static bit indicates the entry is static rather than dynamic.

The Age Count is a counter for internal aging.

The Forward Port indicates which port the FID+MAC information is learned.

The FID specifies the Filter ID.

The MAC Address specifies the MAC address learned.

The ALU Control Register is used to access the dynamic MAC table.

The ALU Index Registers are used to specify the MAC address and FID to get the entry. A direct address can also be used if the Direct Access bit is on.

The Action code is 1 for write, 2, for read, and 3 for search.

The Start bit executes the command.

The Valid bit indicates an entry is found and valid.

The Valid Count indicates the number of valid entries found.

The dynamic MAC table can be used to set a static entry just like in static MAC table.

VLAN Table (0x0 – 0xFFF)

Switch LUE VLAN Entry Register 0 (0x0400 – 0x0403)

Bit	Default	R/W	Description
31	0	RW	**Valid**
27	0	RW	**Forward Option**
26-24	0x0	RW	**Priority**
14-12	0x0	RW	**Multiple Spanning Tree Number**
6-0	0x00	RW	**Filter ID**

Switch LUE VLAN Entry Register 4 (0x0404 – 0x0407)

Bit	Default	R/W	Description
6-0	0x7f	RW	**Untag Ports**

Switch LUE VLAN Entry Register 8 (0x0408 – 0x040B)

Bit	Default	R/W	Description
6-0	0x7f	RW	**Receive Ports**

Switch LUE VLAN Index Register (0x040C – 0x040D)

Bit	Default	R/W	Description
11-0	0x000	RW	**VLAN Index**
7	0	RW	**Start**
1-0	0	RW	**Action**

There are 4096 entries in the VLAN table. Each entry associates its VLAN ID (VID) to a Filter ID (FID). The Receive Ports indicates which ports are forwarded when the lookup fails. The

Untag Ports indicates whether the tag is removed when the frame is forwarded to that port. A tag is added to untagged frame only when VLAN function is enabled. The Valid bit is used to indicate the entry is valid. The Action code is 1 for write, 2 for read, and 3 for clear.

3.2 Sysfs Variables

```
sw/dynamic_table
sw/static_table
sw/alu_index
sw/alu_type
sw/alu_info
sw/alu_fid
sw/alu_use_fid
sw/alu_override
sw/alu_src
sw/alu_dst
sw/alu_mstp
sw/alu_ports
sw/alu_prio
sw/alu_addr
sw/alu_valid
```

The `dynamic_table` variable displays the entries in the dynamic MAC table. The format is "<entry>: <MAC address> <port> <mstp> <age> <source> <destination> <override> <fid>:<use fid>."

It can be written to flush the dynamic MAC table.

The `static_table` variable displays only valid entries in the static MAC table. The format is "<entry>: <MAC address> <ports> <mstp> <priority> <source> <destination> <override> <fid>:<use fid>."

It can be written to clear the static MAC table.

The `alu_index` variable is used to specify the index.

The `alu_type` variable is used to specify the type of entry. It is 0 for static MAC table, 1 for static reserved multicast table, and 2 for dynamic MAC table.

The `alu_info` variable is used to show all the information of the entry. The last number in bracket indicates the state of the entry: 0 for invalid, 1 for valid, and 2 for changes that do not reflect the actual contents in the table. Reading this variable after change to `alu_index` or `alu_type` reads the entry from the table if there are no changes in other variables.

All the other variables allow the user to change the contents of the entry.

Writing to the valid variable will write the entry to the MAC table. Therefore, it should the last one to write.

Examples:

```
echo 0 > sw/alu_type
echo 1 > sw/alu_index
echo 01:00:5e:00:00:00 > sw/alu_addr
echo 2 > sw/alu_fid
echo 1 > sw/alu_use_fid
echo 0x40 > sw/alu_ports
echo 1 > sw/alu_override
echo 1 > sw/alu_valid


echo 1 > sw/alu_type
echo 01:80:C2:00:00:00 > sw/alu_addr
echo 0x40 > sw/alu_ports
echo 1 > sw/alu_override
echo 1 > sw/alu_valid


echo 2 > sw/alu_type
echo 01:00:5e:00:00:00 > sw/alu_addr
echo 0 > sw/alu_fid
echo 0x40 > sw/alu_ports
echo 1 > sw/alu_valid
```

```
sw/vlan_table
sw/vlan_index
sw/vlan_info
sw/vlan_fid
sw/vlan_mstp
sw/vlan_option
sw/vlan_prio
sw/vlan_ports
sw/vlan_untag
sw/vlan_valid
```

The vlan_table variable displays only valid entries in the VLAN table. The format is "<VID>: <FID> <mstp> <priority> <option> <untag> <ports>."

The vlan_index variable is used to specify the VID.

The vlan_info variable is used to show all the information of the entry. The last number in bracket indicates the state of the entry: 0 for invalid, 1 for valid, and 2 for changes that do not

reflect the actual contents in the table. Reading this variable after change to `vlan_index` reads the entry from the table if there are no changes in other variables..

All the other variables allow the user to change the contents of the entry.

Writing to the `valid` variable will write the entry to the VLAN table. Therefore, it should be the last one to write.

Examples:

```
echo 1 > sw/vlan_index
echo 0x7f > sw/vlan_ports
echo 0x0 > sw/vlan_untag
echo 2 > sw/vlan_fid
echo 1 > sw/vlan_valid
```

```
sw/mib
sw6/6_mib
```

The `mib` variable displays the MIB counters. The counters in each port can be displayed individually. Writing to it clears the MIB counters.

4 MII/RMII/GMII/RGMII Interface

4.1 Register Description

Port xMII Control Register 0 (0xN300)

Bit	Default	R/W	Description
7	0	RW	**SGMII Select**
6	1	RW	**Switch Full Duplex**
4	1	RW	**Switch 100MBit**
0	0	RW	**GRXC Enable**

Port xMII Control Register 1 (0xN301)

Bit	Default	R/W	Description
7	1	RW	**MII 100Mbps**

6	0	RW	**MII Edge Select**
4	0	RW	**RGMII Ingress Delay**
3	1	RW	**RGMII Egress Delay**
2	1	RW	**MAC Mode**
1-0	0b00	RW	**MII Select**

These registers are not available for ports with PHY.

The MII Select is 3 for MII, 2 for GMII, 1 or RMII, and 0 for RGMII.

These registers should be read first to make sure the MII connection is setup correctly so that the host can communicate through the switch.

5 Flow Control

5.1 Register Description

Switch MAC Control Register 0 (0x0330)

Bit	Default	R/W	Description
1	0	RW	**Aggressive Back-Off Enable**

It is advised to turn on Aggressive Back-Off in half-duplex mode to enhance performance.

Switch MAC Control Register 1 (0x0331)

Bit	Default	R/W	Description
5	1	RW	**Back Pressure Mode**
4	1	RW	**Flow Control and Back Pressure Fair Mode**
3	0	RW	**No Excessive Collision Drop**

The default Back Pressure Mode is carrier sense-based rather than collision-based.

Flow Control and Back Pressure Fair Mode is the default.

It is advised to turn on No Excessive Collision Drop in half-duplex mode.

Port MAC Control Register 1 (0xN401)

Bit	Default	R/W	Description
3	0	RW	**Back Pressure Enable**

Port Control Register 0 (0xN020)

Bit	Default	R/W	Description
4	0	RW	**Force Transmit Flow Control**
3	0	RW	**Force Receive Flow Control**

It is advised to enable back pressure when in half-duplex mode. Flow control can be forced.

Port Status Register 0 (0xN030)

Bit	Default	R/W	Description
1	0	RO	**Transmit Flow Control Active**
0	0	RO	**Receive Flow Control Active**

The receive and transmit flow control status can be retrieved.

5.2 Sysfs Variables

```
sw/aggr_backoff
sw/back_pressure
sw/no_exc_drop
sw/fair_flow_ctrl
sw6/6_rx_flow_ctrl
sw6/6_tx_flow_ctrl
sw6/6_back_pressure
sw6/6_force_flow_ctrl
```

6 MAC Address

The switch MAC address is used to send PAUSE frames. It can be programmed the same as the host MAC address. Furthermore it is used by the self-address filtering function.

6.1 Register Description

MAC Address Register 1 (0x0302)

Bit	Default	R/W	Description
7-0	0x00	RW	**MAC Address [47:40]**

MAC Address Register 2 (0x0303)

Bit	Default	R/W	Description
7-0	0x10	RW	**MAC Address [39:32]**

MAC Address Register 3 (0x0304)

Bit	Default	R/W	Description
7-0	0xA1	RW	**MAC Address [31:24]**

MAC Address Register 4 (0x0305)

Bit	Default	R/W	Description
7-0	0xFF	RW	**MAC Address [23:16]**

MAC Address Register 5 (0x0306)

Bit	Default	R/W	Description
7-0	0xFF	RW	**MAC Address [15:8]**

MAC Address Register 6 (0x0307)

Bit	Default	R/W	Description
7-0	0xFF	RW	**MAC Address [7:0]**

6.2 Sysfs Variables

`sw/macaddr`

The format of `macaddr` is xx:xx:xx:xx:xx:xx.

7 Link

7.1 Register Description

Port PHY Control Register (0xN100 – 0xN101)

Bit	Default	R/W	Description
15	0	SC	**PHY Soft Reset**
14	0	RW	**PHY Loopback**
13	0	RW	**Speed Select [0]**
12	1	RW	**Auto-Negotiation Enable**
11	0	RW	**Power Down**
10	0	RW	**Isolate**
9	0	RW	**Restart Auto-Negotiation**
8	0	RW	**Force Full Duplex**
7	0	RW	**Collision Test**
6	1	RW	**Speed Select [1]**
5	1	RO	**HP MDI-X Mode**
4	0	RO	**Force MDI-X**
3	0	RO	**Auto MDI/MDI-X Disable**
2	0	RO	**Far-End Fault Disable**
1	0	RO	**Transmit Disable**
0	0	RO	**LED Disable**

Port PHY Status Register (0xN102 – 0xN103)

Bit	Default	R/W	Description
15	0	RO	**T4 Capable**
14	1	RO	**100BT Full-Duplex Capable**
13	1	RO	**100BT Half-Duplex Capable**
12	1	RO	**10BT Full-Duplex Capable**
11	1	RO	**10BT Half-Duplex Capable**
10-9	0x0	RO	**Reserved**
8	1	RO	**Extended Status**

6	0	RO	**Preamble Suppressed**
5	0	RO	**Auto-Negotiation Done**
4	0	RO	**Far-End Fault**
3	1	RO	**Auto-Negotiation Capable**
2	0	RO	**Link Good**
1	0	RO	**Jabber Test**
0	0	RO	**Extended Register Capable**

Port PHY ID 1 Register (0xN104 – 0xN105)

Bit	Default	R/W	Description
15-0	0x0022	RO	**PHY ID High Word**

Port PHY ID 2 Register (0xN106 – 0xN107)

Bit	Default	R/W	Description
15-0	0x1470	RO	**PHY ID Low Word**

Port PHY Auto-Negotiation Advertisement Register (0xN108 – 0xN109)

Bit	Default	R/W	Description
15	0	RO	**Next Page**
14	0	RO	**Reserved**
13	0	RO	**Remote Fault**
12	0x0	RO	**Reserved**
11-10	1	RW	**Advertise Flow Control Capability**
9	0	RW	**Advertise T4 Capability**
8	1	RW	**Advertise 100BT Full-Duplex Capability**
7	1	RW	**Advertise 100BT Half-Duplex Capability**
6	1	RW	**Advertise 10BT Full-Duplex Capability**
5	1	RW	**Advertise 10BT Half-Duplex Capability**
4-0	0x01	RO	**Selector Field**

Port PHY Auto-Negotiation Link Partner Capability Register (0xN10A – 0xN10B)

Bit	Default	R/W	Description
15	0	RO	**Next Page**
14	0	RO	**Partner Acknowledge**

13	0	R(	**Partner Remote Fault**
12	0	RO	**Reserved**
11-10	0	RO	**Partner Flow Control Capable**
9	0	RO	**Reserved**
8	0	RO	**Partner 100BT Full-Duplex Capable**
7	0	RO	**Partner 100BT Half-Duplex Capable**
6	0	RO	**Partner 10BT Full-Duplex Capable**
5	0	RO	**Partner 10BT Half-Duplex Capable**
4-0	0x01	RO	**Reserved**

The definitions of these registers are the same as the standard MDIO registers.

Port PHY 1000 Control Register (0xN112 – 0xN113)

Bit	Default	R/W	Description
15-13	0	RW	**Test Mode**
12	0	RW	**Manual Configuration**
11	0	RW	**Force Master**
10	0	RW	**Master Preferred**
9	1	RW	**Advertise 1000BT Full-Duplex Capability**
8	0	RW	**Advertise 1000BT Half-Duplex Capability**
7-0	0	RO	**Reserved**

Port PHY 1000BT Status Register (0xN114 – 0xN115)

Bit	Default	R/W	Description
15	0	RO	**Master-Slave Fault**
14	0	RO	**Master-Slave Resolution**
13	0	RO	**Local Receive Status**
12	0	RO	**Remote Receive Status**
11	0	RO	**Partner 1000BT Full-Duplex Capable**
10	0	RO	**Partner 1000BT Half-Duplex Capable**
7-0	0	RO	**Idle Error Count**

The definitions of these registers are the same as the standard MDIO registers.

Port PHY MMD Setup Register (0xN11A – 0xN11B)

Bit	Default	R/W	Description
15-14	0	RW	**MMD Operation Mode**
13-5	0	RO	**Reserved**
4-0	0	RW	**MMD Device ID**

Port PHY MMD Data Register (0xN11C – 0xN11D)

Bit	Default	R/W	Description
15-0	0	RW	**MMD Index or Data**

Port PHY Extended Status Register (0xN11E – 0xN11F)

Bit	Default	R/W	Description
15	0	RO	**1000BX Full-Duplex Capable**
14	0	RO	**1000BX Half-Duplex Capable**
13	1	RO	**1000BT Full-Duplex Capable**
12	1	RO	**1000BT Half-Duplex Capable**

Port Status Register 0 (0xN030)

Bit	Default	R/W	Description
4-3	0b10	RO	**Interface Speed**
2	1	RO	**Interface Full Duplex**
1	0	RO	**Transmit Flow Control Active**
0	0	RO	**Receive Flow Control Active**

This register provides the speed and duplex of the interface when a link is established. The Interface Speed is 0 for 10BT, 1 for 100BT, and 2 for 1000BT. It is normally 3 when there is no link.

7.2 Sysfs Variables

```
sw/duplex
sw/speed
sw/force
sw/flow_ctrl
```

```
sw6/6_duplex
sw6/6_speed
sw4/4_power
sw4/4_mac_loopback
sw4/4_phy_loopback
```

The duplex variable displays the duplex mode as either full-duplex or half-duplex if there is a link; otherwise it is unlinked. Writing 0 means auto, normally selecting full-duplex as the best operating mode. Writing 1 specifies half-duplex mode. Writing 2 specifies full-duplex mode.

The speed variable displays the speed as either 10, 100, or 1000 Mbps if there is a link; otherwise it is unlinked. Writing 0 means auto, normally selecting 1000 Mbps as the best operating speed. Writing 10 specifies 10 Mbps. Writing 100 specifies 100 Mbps only. Writing 1000 specifies 1000 Mbps only.

The force variable specifies that auto-negotiation not to be used. Writing to it updates the registers for the new link mode.

The flow_ctrl variable enables flow control advertisement.

The general duplex and speed variables indicate at least one of the ports of the switch is connected. To get the status of each individual port the port duplex and speed variables can be read.

The power variable can be used to turn off the power of the PHY.

The mac_loopback variable is used to program the registers to setup MAC loopback.

The phy_loopback variable is used to program the registers to setup PHY loopback.

8 LinkMD

8.1 Register Description

Port 1 PHY Special Control Register 1 (0x07C– 0x07D)
Port 2 PHY Special Control Register 1 (0x094– 0x095)

Bit	Default	R/W	Description
15	0	RO	**Less Than 10m**
14-13	0x0	RO	**LinkMD Result**
12	0	SC	**LinkMD Test Enable**
11	0	RW	**Force Link**

9	0	RW	**Remote Loopback**
8-0	0x000	RW	**LinkMD Fault Count**

Port 1 Control Register 4 (0x07E – 0x07F)
Port 2 Control Register 4 (0x096 – 0x097)

Bit	Default	R/W	Description
15	0	RW	**LED Off**
14	0	RW	**Transmit Disable**
13	0	RW	**Restart Auto Negotiation**
12	0	RW	**Far-End Fault Disable**
11	0	RW	**Power Down**
10	0	RW	**Auto MDI/MDI-X Disable**
9	0	RW	**Force MDI-X**
8	0	RW	**Far-End Loopback**
7	0	RW	**Auto Negotiation Enable**
6	0	RW	**Force Speed**
5	0	RW	**Force Duplex**
4	0	RW	**Advertise Flow Control Capability**
3	0	RW	**Advertise 100BT Full-Duplex Capability**
2	0	RW	**Advertise 100BT Half-Duplex Capability**
1	0	RW	**Advertise 10BT Full-Duplex Capability**
0	0	RW	**Advertise 10BT Half-Duplex Capability**

Port 1 Status Register (0x080 – 0x081)
Port 2 Status Register (0x098 – 0x099)

Bit	Default	R/W	Description
15	1	RW	**HP Auto MDI-X Mode**
14	0	RO	**Reserved**
13	0	RO	**Polarity Reverse**
12	0	RO	**Receive Flow Control Active**
11	0	RO	**Transmit Flow Control Active**
10	0	RO	**Link Speed**
9	0	RO	**Link Duplex**

8	0	RO	**Far-End Fault**
7	0	RO	**MDI-X Not Active**
6	0	RO	**Auto-Negotiation Done**
5	0	RO	**Link Good**
4	0	RO	**Partner Flow Control Capable**
3	0	RO	**Partner 100BT Full-Duplex Capable**
2	0	RO	**Partner 100BT Half-Duplex Capable**
1	0	RO	**Partner 10BT Full-Duplex Capable**
0	0	RO	**Partner 10BT Half-Duplex Capable**

8.2 Sysfs Variables

`sw4/4_linkmd`

Writing to the `linkmd` variable starts the LinkMD test. Reading from the variable displays a string in the format "[overall length:overall status] [pair 1 length: pair 1 status] [pair 2 length:pair 2 status]." The meaning of the status is 0 for unknown, 1 for good,

`sw4/4_sqi`

Writing to the `sqi` variable starts the SQI test. Reading from the variable displays a number ranging from 0 to 15, with 15 being best. A value of 0 is returned if the test is not run yet.

9 Jumbo Frame

The switch supports transmitting and receiving frames with length longer than the standard 1518 bytes. The maximum length supported is 9000 bytes.

9.1 Register Description

Switch Maximum Transmit Unit Register (0x0308 – 0x0309)

Bit	Default	R/W	Description
13-0	2000	RW	**Maximum Transmit Frame Size**

Switch MAC Control Register 0 (0x0330)

Bit	Default	R/W	Description
3	0	RW	**Frame Length Check**

Switch MAC Control Register 2 (0x0331)

Bit	Default	R/W	Description
2	0	RW	**Jumbo Packet Support**
1	0	RW	**Legal Maximum Packet Size Check Disable**

The Jumbo Packet Support allows packet size up to 9000 bytes.

The Legal Maximum Packet Size Check allows packet size of 1522 bytes for tagged packets and 1518 bytes for untagged packets. Otherwise, the maximum packet size is 1536 bytes.

9.2 Sysfs Variables

```
sw/mtu
sw/jumbo_packet
sw/legal_packet
sw/length_check
```

10 Aging

Aging allows the switch lookup engine to discard records that are not updated for a certain time. This frees up space in the dynamic MAC table and minimizes lookup time. The normal aging period is about 75 seconds.

10.1 Register Description

Switch LUE Control Register 1 (0x311)

Bit	Default	R/W	Description
2	1	RW	**Aging Enable**

1	0	RW	**Fast Aging Enable**
0	0	RW	**Link Change Aging Enable**

Switch LUE Control Register 3 (0x313)

Bit	Default	R/W	Description
7-0	75	RW	**Age Period**

Aging is enabled by default. Fast Aging reduces the aging period to microseconds. It is used normally in support to Spanning Tree Protocol. Link Change Aging allows the switch to flush the MAC table when the cable is disconnected. It is normally good practice to enable it.

Switch LUE Control Register 2 (0x312)

Bit	Default	R/W	Description
5	0	RW	**Flush STP MAC Table**
4	0	RW	**Flush MSTP MAC Table**

Switch LUE Control Register 2 (0x0312)

Bit	Default	R/W	Description
3-2	0	RW	**MAC Table Flush Option**

The dynamic MAC address table can be flushed on command. However, it requires the port learning to be disabled first. Basically this command removes all entries of the port that stops learning.

10.2 Sysfs Variables

```
sw/aging
sw/fast_aging
sw/link_aging
```

11 Broadcast Storm Protection

Broadcast Storm Protection protects the switch system from receiving too many broadcast packets. A rate can be set to specify how much broadcast packets are allowed to forward. There

is an option to include multicast packets in this protection.

11.1 Register Description

Switch MAC Control Register 1 (0x0331)

Bit	Default	R/W	Description
6	1	RW	**Multicast Storm Protection Disable**

Multicast Storm Protection is normally disabled.

Switch MAC Control Register 2 (0x0332)

Bit	Default	R/W	Description
2-0	0x0	RW	**Broadcast Storm Protection Rate [10:8]**

Switch MAC Control Register 3 (0x0333)

Bit	Default	R/W	Description
7-0	0x63	RW	**Broadcast Storm Protection Rate [7:0]**

The default storm protection rate is 1%. The rate value is calculated as `(148800 * 67 * 1 / 100 / 1000 = 0x63)`. It is broken in two parts when programmed to the register.

Port MAC Control Register 1 (0xN400)

Bit	Default	R/W	Description
1	0	RW	**Broadcast Storm Protection Enable**

The Broadcast Storm Protection can be enabled in each port individually.

11.2 Sysfs Variables

```
sw/mcast_storm
sw/bcast_per
sw6/6_bcast_storm
```

The bcast_per variable displays the storm protection rate in percentage. The maxim value is about 21%.

12 Port Source Address Filtering

The switch can filter out packets sent by its own ports in a ring topology. For that to work a MAC address has to be assigned. Normally it will be the same as the host MAC address.

12.1 Register Description

Switch LUE Control Register 1 (0x0311)

Bit	Default	R/W	Description
6	0	RW	**Source Address Filtering**

Port LUE Control Register (0xNB00)

Bit	Default	R/W	Description
3	0	RW	**Source Address Filtering**

The Source Address Filtering can be enabled in each port to filter address.

12.2 Sysfs Variables

```
sw6/6_src_addr_filter
sw/macaddr
```

The format of macaddr is xx:xx:xx:xx:xx:xx.

13 Spanning Tree Support

13.1 Register Description

Port LUE MSTP Index Register (0xNB01)

Bit	Default	R/W	Description
7-0	0	RW	**MSTP Index**

Port LUE MSTP State Register (0xNB04)

Bit	Default	R/W	Description
2	1	RW	**Transmit Enable**
1	1	RW	**Receive Enable**
0	0	RW	**Learning Disable**

The port can be shut off by disabling transmit and receive. The learning can be turned off depending on STP state.

Switch LUE Control Register 1 (0x0311)

Bit	Default	R/W	Description
5	0	SC	**Flush STP MAC Table**
4	0	SC	**Flush MSTP MAC Table**

Switch LUE Control Register 2 (0x0312)

Bit	Default	R/W	Description
3-2	0	RW	**MAC Table Flush Option**

The learning of the ports needs to be disabled first before the whole dynamic MAC table can be flushed with the Flush Dynamic MAC Table command.

Mode 1 is flush dynamic MAC table, mode 2 is flush static MAC table, and mode 3 is flush both.

13.2 Sysfs Variables

```
sw6/6_rx
sw6/6_tx
sw6/6_learn
```

13.3 RSTP Support

Following variables are used for running RSTP inside the switch driver.

```
sw/stp_br_fwd_delay
sw/stp_br_hello_time
sw/stp_br_info
sw/stp_br_max_age
sw/stp_br_on
sw/stp_br_prio
sw/sttp_br_tx_hold
sw/stp_version
sw6/6_stp_admin_edge
sw6/6_stp_admin_p2p
sw6/6_stp_admin_path_cost
sw6/6_stp_auto_edge
sw6/6_stp_info
sw6/6_stp_mcheck
sw6/6_stp_on
sw6/6_stp_path_coast
sw6/6_stp_prio
```

They are used to configure various RSTP parameters. The `stp_br_on` variable is used to turn on and off RSTP function of the switch. The `#_stp_on` variable is used to turn on and off the individual port for participating in RSTP.

14 Tail Tagging

Tail Tagging is a feature in which the switch attaches a tag at the end of packet to indicate the source port when forwarded to the host port. The host then has a way to determine the source of the packet. The host can attach the tag when transmitting to specify the destination ports. PTP operation requires tail tag.

14.1 Register Description

Port Control Register 1 (0xN020)

Bit	Default	R/W	Description

2	0	RW	**Tail Tag Enable**

After tail tagging is enabled the host needs to make sure the tag at the end of transmit packet is set correctly to ensure proper operation. It requires at least 60 bytes to send a frame through the switch.

The incoming tag shows either 0 for port 1, 1 for port 2, and so on. A special bit is 0x80, which indicates a PTP message. In this case there are 4 additional bytes before the tag to hold the receive timestamp. The outgoing tag can be 0x400 for normal lookup, 1 for port 1, 2 for port 2, 4 for port 3, and so on. A bitmap 0x7f can be used to broadcast to all the ports. A special bit 0x200 can be added to override sending through a closed port. (The bits 7 and 8 can store the frame priority 0 to 3.)

14.2 Sysfs Variables

```
sw6/6_tail_tag
```

15 IGMP Snooping

IGMP Snooping allows Internet Group Management Protocol (IGMP) packets to be forwarded to the host port so that software can setup the Static MAC Address Table to limit the broadcasting of multicast packets. Tail Tagging mode needs to be enabled so that software knows which port the IGMP packet is received. Note that enabling these modes is actually counter-productive unless the software does not want these multicast packets to be forwarded to other switches.

15.1 Register Description

Switch MRI Control Register 0 (0x0370)

Bit	Default	R/W	Description
6	0	RW	**IGMP Snooping Enable**

15.2 Sysfs Variables

```
sw/igmp_snoop
```

16 IPv6 MLD Snooping

IPv6 MLD Snooping is similar to IGMP Snooping but works for IPv6 packets. It allows Multicast Listener Discovery (MLD) packets to be forwarded to the host port so that software can setup the Static MAC Address Table to limit the broadcasting of multicast packets.

16.1 Register Description

Switch MRI Control Register 0 (0x0370)

Bit	Default	R/W	Description
3	0	RW	**IPv6 MLD Snooping Option**
2	0	RW	**IPv6 MLD Snooping Enable**

IPv6 MLD Snooping allows IPv6 packets with next header equals to 1 or 58 to be forwarded to the host port.

IPv6 MLD Snooping Option allows IPv6 packets with next header equals to 43, 44, 50, 51, or 60 to be forwarded to the host port.

16.2 Sysfs Variables

```
sw/ipv6_mld_snoop
sw/ipv6_mld_option
```

17 Port Mirroring

Port Mirroring allows network traffic among ports to be routed to a specific port for network monitoring and debugging purpose.

17.1 Register Description

Switch MRI Control Register 0 (0x0370)

Bit	Default	R/W	Description
0	0	RW	**Sniff Mode Select**

The default sniff mode is RX or TX, meaning either the source port or destination port needs to match. This is the mode used to implement RX only sniff. Setting 1 selects RX and TX sniff mode, meaning both the source port and destination need to match. This can narrow down the mirrored traffic to a specific path.

Port MRI Mirror Control Register (0xN800)

Bit	Default	R/W	Description
6	0	RW	**Receive Sniff**
5	0	RW	**Transmit Sniff**
1	0	RW	**Sniffer Port**

Each port can turn on receive or transmit mirroring. At least one needs to be a sniffer port.

Port VLAN Membership Register (0xNA04-0xNA07)

Bit	Default	R/W	Description
7-0	0x7F	RW	**Port VLAN Membership**

It is required that the sniffer port is in the port membership of the mirror rx or tx port.

17.2 Sysfs Variables

```
sw/mirror_mode
sw6/6_mirror_rx
sw6/6_mirror_tx
sw6/6_mirror_port
sw6/6_member
```

18 VLAN

18.1 Register Description

Switch LUE Control Register 1 (0x0310)

Bit	Default	R/W	Description
7	0	RW	**802.1Q VLAN Enable**
6	1	RW	**Drop Invalid VID**

Switch MAC Control Register 2 (0x0332)

Bit	Default	R/W	Description
3	0	RW	**Null VID Replacement**

Ingress packet with zero VID will be replaced with the port VID value.

Switch QM Control Register 1 (0x0390)

Bit	Default	R/W	Description
1	1	RW	**Unicast Port-VLAN Mismatch Discard**

VLAN table needs to be setup first before turning on VLAN. Unicast Port-VLAN Mismatch Discard makes sure no packets can cross the VLAN boundary.

Port MTI Queue Control Register 3 (0xN904-0xN907)

Bit	Default	R/W	Description
0	0	RW	**Port VID Replacement**

Egress packet with zero VID will be replaced with the port VID value.

Switch LUE Unknown Control Register 8 (0x0328 – 0x032B)

Bit	Default	R/W	Description
31	0	RW	**Enable Unknown VID Packet Forward**
6-0	0	RW	**Forward Ports**

Frames with invalid VID can be forwarded to certain ports.

Port MRI MAC Control Register (0xN802)

Bit	Default	R/W	Description
7	0	RW	**User Priority Ceiling**
4	0	RW	**Drop Non-VLAN Packet**
3	0	RW	**Drop Tagged Packet**

Port LUE Control Register (0xNB00)

Bit	Default	R/W	Description
7	0	RW	**VID 0 Lookup**
6	0	RW	**Ingress VLAN Filtering**
5	0	RW	**Discard Non PVID Packets**

Ingress VLAN filtering discards VLAN packets from port that is not in the VLAN membership of the VLAN table.

Discard Non PVID Packets discards VLAN packets that do not match the port VID.

User Priority Ceiling can remap 802.1p priority in the VLAN packets with the one defined in the port.

Port VLAN Membership Register (0xNA04-0xNA07)

Bit	Default	R/W	Description
7-0	0x7F	RW	**Port VLAN Membership**

Port VLAN Membership defines the membership of the port.

Port VID Register (0xN000 – 0xN001)

Bit	Default	R/W	Description
15-13	0	RW	**User Priority Field of VLAN Tag**
12	0	RW	**CFI of VLAN Tag**
11-0	0x001	RW	**VID of VLAN Tag**

The VID defined for each port is used in several switch functions.

The User Priority Field can be used for 802.1p priority remapping.

Port Custom VID Register (0xN002 – 0xN003)

Bit	Default	R/W	Description
15-13	0	RW	**User Priority Field of VLAN Tag**
12	0	RW	**CFI of VLAN Tag**
11-0	0x001	RW	**VID of VLAN Tag**

Port AVB SR 1 VID Register (0xN004 – 0xN005)

Bit	Default	R/W	Description
15-13	3	RW	**User Priority Field of VLAN Tag**
12	0	RW	**CFI of VLAN Tag**
11-0	0x002	RW	**VID of VLAN Tag**

Port AVB SR 2 VID Register (0xN006 – 0xN007)

Bit	Default	R/W	Description
15-13	3	RW	**User Priority Field of VLAN Tag**
12	0	RW	**CFI of VLAN Tag**
11-0	0x002	RW	**VID of VLAN Tag**

Port AVB SR 1 Type Register (0xN008 – 0xN009)

Bit	Default	R/W	Description
15-0	0x88b5	RW	**SR Type 1**

Port AVB SR 2 Type Register (0xN00A – 0xN00B)

Bit	Default	R/W	Description
15-0	0x88b5	RW	**SR Type 2**

18.2 Sysfs Variables

```
sw/vlan
sw/drop_inv_vid
sw/null_vid
sw/vlan_bound
sw/unk_vid_fwd
sw/unk_vid_ports
sw6/6_member
```

```
sw6/6_vid
sw6/6_cust_vid
sw6/6_sr_1_vid
sw6/6_sr_2_vid
sw6/6_sr_1_type
sw6/6_sr_2_type
sw6/6_drop_non_vlan
sw6/6_drop_tagged
sw6/6_ingress
sw6/6_non_vid
sw6/6_replace_prio
sw6/6_replace_vid
```

18.3 Tag Insertion

VLAN tag insertion allows two VLANs with tagged and untagged packets coexist.

18.3.1 Register Description

Switch LUE VLAN Entry Register 4 (0x0404 – 0x0407)

Bit	Default	R/W	Description
6-0	0x00	RW	**Untag Ports**

VLAN tag is always inserted when the untag ports of the VID entry are not defined and VLAN is enabled.

18.3.2 Sysfs Variables

```
sw/vlan_untag
```

18.4 Tag Removal

VLAN tag removal allows two VLANs with tagged and untagged packets coexist.

18.4.1 Register Description

Switch LUE VLAN Entry Register 4 (0x0404 – 0x0407)

Bit	Default	R/W	Description
6-0	0x7f	RW	**Untag Ports**

VLAN tag is removed if untag ports are defined.

18.4.2 Sysfs Variables

```
sw/vlan_untag
```

18.5 Double Tag

18.5.1 Register Description

Switch Operation Control Register (0x0300)

Bit	Default	R/W	Description
7	0	RW	**Double Tagging Enable**
0	1	RW	**Start Switch**

Switch ISP TPID Register (0x030A – 0x030B)

Bit	Default	R/W	Description
15-0	0x9100	RW	**Tag for Untagged Frame or ISP Tag**

18.5.2 Sysfs Variables

```
sw/double_tag
sw/isp
```

18.6 Drop Tagged Packet

18.6.1 Register Description

Port MRI MAC Control Register (0xN802)

Bit	Default	R/W	Description
3	0	RW	**Drop Tagged Packet**

18.6.2 Sysfs Variables

`sw6/6_drop_tagged`

19 QoS

19.1 Priority Queues

There are 8 priorities for incoming frames, but at most 4 transmit queues, so there are several ways to map these priorities to different queues.

19.1.1 Register Description

Port Control Register 0 (0xN020)

Bit	Default	R/W	Description
1-0	0	RW	**TX Queues Select Enable**

Multiple priority queues can be turned on in each port. They are needed for QoS priority. The select is 0 for single queue, 1 for 2 queues, and 2 for 4 queues.

Port MRI TC Map Register (0xN808 – 0xN80B)

Bit	Default	R/W	Description
31-28	0x3	RW	**Regenerated Priority for Priority 7**
27-24	0x3	RW	**Regenerated Priority for Priority 6**
23-20	0x2	RW	**Regenerated Priority for Priority 5**
19-16	0x2	RW	**Regenerated Priority for Priority 4**
15-12	0x1	RW	**Regenerated Priority for Priority 3**
11-8	0x1	RW	**Regenerated Priority for Priority 2**
7-4	0x0	RW	**Regenerated Priority for Priority 1**
3-0	0x0	RW	**Regenerated Priority for Priority 0**

The default setting is 0x33221100 so it is easy to understand the mapping.

Port MRI Priority Control Register (0xN801)

Bit	Default	R/W	Description
7	0	RW	**Highest Priority Selected**
6	0	RW	**OR All Priorities**
4	0	RW	**MAC Address Priority Enable**
3	0	RW	**VLAN Priority Enable**
2	0	RW	**802.1p Priority Enable**
1	0	RW	**DiffServ Priority Enable**
0	0	RW	**ACL Priority Enable**

The MAC address priority is set through the static or dynamic MAC table.

The VLAN priority is set through the VLAN table.

The ACL priority is set through the ACL table.

Port MTI Queue Index Register (0xN900)

Bit	Default	R/W	Description
1-0	0	RW	**Queue Index**

The queue index is used whenever one of the 4 queues needs to be accessed.

Port MTI Queue Control Register 0 (0xN914)

Bit	Default	R/W	Description

7-6	0b10	RW	**Schedule Mode**
5-4	0	RW	**Shaping Enable**

The Schedule Mode is 0 for strict priority, or 2 for WRR.

The Shaping Enable is 0 for disabled, or 1 for SRP enabled.

Port MTI Queue Control Register 1 (0xN915)

Bit	Default	R/W	Description
6-0	1	RW	**Queue Ratio**

The transmit priority in each queue can be controlled individually. Setting 0 to Schedule Mode means all packets in this priority queue are transmitted before transmitting packets in lower priority queues. The Queue Ratio indicates the number of packets allowed to transmit within a certain time. Normally higher priority queues can transmit more packets than lower priority queues. If those numbers are changed in an illogical way, the queue priority can be switched.

The default value is 1 for all 4 queues. For WWR scheduling it is better to program 1 for queue 1, 2 for queue 2, 4 for queue 3, and 8 for queue 4 as this is the default setting from the other KSZ switches.

19.1.2 Sysfs Variables

```
sw6/6_prio_queue
sw6/6_q_index
sw6/6_q_scheduling
sw6/6_q_tx_ratio
sw6/6_prio_highest
sw6/6_prio_or
sw6/6_prio_acl
sw6/6_prio_mac
sw6/6_prio_vlan
```

19.2 Port-Based Priority

19.2.1 Register Description

Port MRI MAC Control Register (0xN802)

Bit	Default	R/W	Description
2-0	0x0	RW	**Port-Based Priority Select**

If DiffServ or 802.1p priority is not enabled then port-based priority is used. The queue priorities are 0 to 7.

19.2.2 Sysfs Variables

```
sw6/6_port_prio
```

19.3 802.1p Priority

19.3.1 Register Description

Switch MAC Priority Mapping Register 0 (0x0338)

Bit	Default	R/W	Description
6-4	0x1	RW	**IEEE 802.1p mapping of frame's priority with tag 1**
2-0	0x0	RW	**IEEE 802.1p mapping of frame's priority with tag 0**

Switch MAC Priority Mapping Register 1 (0x0339)

Bit	Default	R/W	Description
6-4	0x3	RW	**IEEE 802.1p mapping of frame's priority with tag 3**
2-0	0x2	RW	**IEEE 802.1p mapping of frame's priority with tag 2**

Switch MAC Priority Mapping Register 2 (0x033A)

Bit	Default	R/W	Description
6-4	0x5	RW	**IEEE 802.1p mapping of frame's priority with tag 5**
2-0	0x4	RW	**IEEE 802.1p mapping of frame's priority with tag 4**

Switch MAC Priority Mapping Register 3 (0x033B)

Bit	Default	R/W	Description

6-4	0x7	RW	**IEEE 802.1p mapping of frame's priority with tag 7**
2-0	0x6	RW	**IEEE 802.1p mapping of frame's priority with tag 6**

The 802.1p priority has 8 tags, 0 to 7. Each tag can be mapped to a queue priority from 0 to 7.

Port MRI Priority Control Register (0xN801)

Bit	Default	R/W	Description
2	0	RW	**802.1p Priority Enable**

The 802.1p priority can be turned on in each port individually.

19.3.2 Sysfs Variables

```
sw/p_802_1p_map
sw6/6_p_802_1p
```

The `p_802_1p_map` variable displays the priority queue mapping of the 8 tags. To change the mapping the users need to pass the information in the format of "<tag>=<queue>." The tag value should be less than 8, and the queue value should be less than 8. To enter the information quickly the value can be more than 7 to indicate the value is for more than 1 tag. For a value 7 or less it is meant for 1 tag; and 0x77 or less, 2 tags.

19.4 DiffServ Priority

19.4.1 Register Description

Switch MAC TOS Priority Register 0 (0x0340)

Bit	Default	R/W	Description
6-4	0	RW	**DiffServ mapping of frame's priority with value 0x04 [5:3]**
2-0	0	RW	**DiffServ mapping of frame's priority with value 0x00 [2:0]**

Switch MAC TOS Priority Register 1 (0x0341)

Bit	Default	R/W	Description

6-4	0	RW	**DiffServ mapping of frame's priority with value 0x0c [11:9]**
2-0	0	RW	**DiffServ mapping of frame's priority with value 0x08 [8:6]**

Switch MAC TOS Priority Register 2 (0x0342)

Bit	Default	R/W	Description
6-4	0	RW	**DiffServ mapping of frame's priority with value 0x14 [17:15]**
2-0	0	RW	**DiffServ mapping of frame's priority with value 0x10 [14:12]**

Switch MAC TOS Priority Register 3 (0x0343)

Bit	Default	R/W	Description
6-4	0	RW	**DiffServ mapping of frame's priority with value 0x1c [23:21]**
2-0	0	RW	**DiffServ mapping of frame's priority with value 0x18 [20:18]**

Switch MAC TOS Priority Register 4 (0x0344)

Bit	Default	R/W	Description
6-4	0	RW	**DiffServ mapping of frame's priority with value 0x24 [29:27]**
2-0	0	RW	**DiffServ mapping of frame's priority with value 0x20 [26:24]**

Switch MAC TOS Priority Register 5 (0x0345)

Bit	Default	R/W	Description
6-4	0	RW	**DiffServ mapping of frame's priority with value 0x2c [35:33]**
2-0	0	RW	**DiffServ mapping of frame's priority with value 0x28 [32:30]**

Switch MAC TOS Priority Register 6 (0x0346)

Bit	Default	R/W	Description
6-4	0	RW	**DiffServ mapping of frame's priority with value 0x34 [41:39]**
2-0	0	RW	**DiffServ mapping of frame's priority with value 0x30 [38:36]**

Switch MAC TOS Priority Register 7 (0x0347)

Bit	Default	R/W	Description
6-4	0	RW	**DiffServ mapping of frame's priority with value 0x3c [47:45]**
2-0	0	RW	**DiffServ mapping of frame's priority with value 0x38 [44:42]**

Switch MAC TOS Priority Register 8 (0x0348)

Bit	Default	R/W	Description
6-4	0	RW	**DiffServ mapping of frame's priority with value 0x44 [52:51]**
2-0	0	RW	**DiffServ mapping of frame's priority with value 0x40 [50:48]**

Switch MAC TOS Priority Register 9 (0x0349)

Bit	Default	R/W	Description
6-4	0	RW	**DiffServ mapping of frame's priority with value 0x4c [59:57]**
2-0	0	RW	**DiffServ mapping of frame's priority with value 0x48 [56:45]**

Switch MAC TOS Priority Register 10 (0x034A)

Bit	Default	R/W	Description
6-4	0	RW	**DiffServ mapping of frame's priority with value 0x54 [65:63]**
2-0	0	RW	**DiffServ mapping of frame's priority with value 0x50 [62:60]**

Switch MAC TOS Priority Register 11 (0x034B)

Bit	Default	R/W	Description
6-4	0	RW	**DiffServ mapping of frame's priority with value 0x5c [71:69]**
2-0	0	RW	**DiffServ mapping of frame's priority with value 0x58 [68:66]**

Switch MAC TOS Priority Register 12 (0x034C)

Bit	Default	R/W	Description
6-4	0	RW	**DiffServ mapping of frame's priority with value 0x64 [77:75]**
2-0	0	RW	**DiffServ mapping of frame's priority with value 0x60 [74:72]**

Switch MAC TOS Priority Register 13 (0x034D)

Bit	Default	R/W	Description
6-4	0	RW	**DiffServ mapping of frame's priority with value 0x6c [83:81]**
2-0	0	RW	**DiffServ mapping of frame's priority with value 0x68 [80:78]**

Switch MAC TOS Priority Register 14 (0x034E)

Bit	Default	R/W	Description

6-4	0	RW	**DiffServ mapping of frame's priority with value 0x74 [89:87]**
2-0	0	RW	**DiffServ mapping of frame's priority with value 0x70 [86:84]**

Switch MAC TOS Priority Register 15 (0x034F)

Bit	Default	R/W	Description
6-4	0	RW	**DiffServ mapping of frame's priority with value 0x7c [95:93]**
2-0	0	RW	**DiffServ mapping of frame's priority with value 0x78 [92:90]**

Switch MAC TOS Priority Register 16 (0x0350)

Bit	Default	R/W	Description
6-4	0	RW	**DiffServ mapping of frame's priority with value 0x84 [101:99]**
2-0	0	RW	**DiffServ mapping of frame's priority with value 0x80 [98:96]**

Switch MAC TOS Priority Register 17 (0x0351)

Bit	Default	R/W	Description
6-4	0	RW	**DiffServ mapping of frame's priority with value 0x8c [107:105]**
2-0	0	RW	**DiffServ mapping of frame's priority with value 0x88 [104:102]**

Switch MAC TOS Priority Register 18 (0x0352)

Bit	Default	R/W	Description
6-4	0	RW	**DiffServ mapping of frame's priority with value 0x94 [113:111]**
2-0	0	RW	**DiffServ mapping of frame's priority with value 0x90 [110:108]**

Switch MAC TOS Priority Register 19 (0x0353)

Bit	Default	R/W	Description
6-4	0	RW	**DiffServ mapping of frame's priority with value 0x9c [119:117]**
2-0	0	RW	**DiffServ mapping of frame's priority with value 0x98 [116:114]**

Switch MAC TOS Priority Register 20 (0x0354)

Bit	Default	R/W	Description
6-4	0	RW	**DiffServ mapping of frame's priority with value 0xa4 [125:123]**
2-0	0	RW	**DiffServ mapping of frame's priority with value 0xa0 [122:120]**

Switch MAC TOS Priority Register 21 (0x0355)

Bit	Default	R/W	Description
6-4	0	RW	**DiffServ mapping of frame's priority with value 0xac [131:129]**
2-0	0	RW	**DiffServ mapping of frame's priority with value 0xa8 [128:126]**

Switch MAC TOS Priority Register 22 (0x0356)

Bit	Default	R/W	Description
6-4	0	RW	**DiffServ mapping of frame's priority with value 0xb4 [137:135]**
2-0	0	RW	**DiffServ mapping of frame's priority with value 0xb0 [134:132]**

Switch MAC TOS Priority Register 23 (0x0357)

Bit	Default	R/W	Description
6-4	0	RW	**DiffServ mapping of frame's priority with value 0xbc [143:141]**
2-0	0	RW	**DiffServ mapping of frame's priority with value 0xb8 [140:138]**

Switch MAC TOS Priority Register 24 (0x0358)

Bit	Default	R/W	Description
6-4	0	RW	**DiffServ mapping of frame's priority with value 0xc4 [149:147]**
2-0	0	RW	**DiffServ mapping of frame's priority with value 0xc0 [146:144]**

Switch MAC TOS Priority Register 25 (0x0359)

Bit	Default	R/W	Description
6-4	0	RW	**DiffServ mapping of frame's priority with value 0xcc [155:153]**
2-0	0	RW	**DiffServ mapping of frame's priority with value 0xc8 [152:150]**

Switch MAC TOS Priority Register 26 (0x035A)

Bit	Default	R/W	Description
6-4	0	RW	**DiffServ mapping of frame's priority with value 0xd4 [161:159]**
2-0	0	RW	**DiffServ mapping of frame's priority with value 0xd0 [158:156]**

Switch MAC TOS Priority Register 27 (0x035B)

Bit	Default	R/W	Description

6-4	0	RW	**DiffServ mapping of frame's priority with value 0xdc [167:165]**
2-0	0	RW	**DiffServ mapping of frame's priority with value 0xd8 [164:162]**

Switch MAC TOS Priority Register 28 (0x035C)

Bit	Default	R/W	Description
6-4	0	RW	**DiffServ mapping of frame's priority with value 0xe4 [173:171]**
2-0	0	RW	**DiffServ mapping of frame's priority with value 0xe0 [170:168]**

Switch MAC TOS Priority Register 29 (0x035D)

Bit	Default	R/W	Description
6-4	0	RW	**DiffServ mapping of frame's priority with value 0xec [179:177]**
2-0	0	RW	**DiffServ mapping of frame's priority with value 0xe8 [176:174]**

Switch MAC TOS Priority Register 30 (0x035E)

Bit	Default	R/W	Description
6-4	0	RW	**DiffServ mapping of frame's priority with value 0xf4 [185:183]**
2-0	0	RW	**DiffServ mapping of frame's priority with value 0xf0 [182:180]**

Switch MAC TOS Priority Register 31 (0x035F)

Bit	Default	R/W	Description
6-4	0	RW	**DiffServ mapping of frame's priority with value 0xfc [191:189]**
2-0	0	RW	**DiffServ mapping of frame's priority with value 0xf8 [188:186]**

The DiffServ priority has 64 TOS/DiffServ/Class values. Each value can be mapped to a queue priority from 0 to 7.

Port MRI Priority Control Register (0xN801)

Bit	Default	R/W	Description
1	0	RW	**DiffServ Priority Enable**

The DiffServ priority can be turned on in each port individually.

19.4.2 Sysfs Variables

```
sw/diffserv_map
sw6/6_diffserv
```

The `diffserv_map` variable displays the priority queue mapping of the 64 DiffServ classes. To change the mapping the users need to pass the information in the format of "<class>=<queue>." The class value should be less than 64, and the queue value should be less than 8. To enter the information quickly the value can be more than 7 to indicate the value is for more than 1 class. For a value 7 or less it is meant for 1 class; and 0x77 or less, 2 classes.

19.5 Rate Limiting

19.5.1 Register Description

Port MAC Ingress Rate Limit Control Register (0xN403)

Bit	Default	R/W	Description
6	0	RW	**Ingress Rate Limit Is Port Based**
5	0	RW	**Rate Limit is Packet Based**
4	0	RW	**Ingress Rate Limit Flow Control**
3-2	0x0	RW	**Ingress Limit Mode**
1	0	RW	**Count Inter Frame Gap**
0	0	RW	**Count Preamble**

The Ingress Limit Mode selects what frames are counted and limited. Mode 0 means all frames; mode 1 means broadcast, multicast, and flooded unicast frames; mode 2 means broadcast and multicast frames; and mode 3 means broadcast frames only.

Count Inter Frame Gap also counts IFG bytes, which are 12 per frame.

Count Preamble also counts preamble bytes, which are 8 per frame.

Switch MAC Control Register 5 (0x0335)

Bit	Default	R/W	Description
7-4	0	RO	**Reserved**
3	0	RW	**Egress Rate Limit Is Queue Based**

The default is egress rate limit is port based. So when multiple queues function is enabled it is better to program this register to enable queue based egress rate limiting. Otherwise only the first queue will take effect to limit the transmission.

Port Ingress Rate Register 0 (0xN410)

Bit	Default	R/W	Description
7	0	RO	**Reserved**
6-0	0x00	RW	**Ingress Data Rate Limit for Priority 0 Frames**

Port Ingress Rate Register 1 (0xN411)

Bit	Default	R/W	Description
7	0	RO	**Reserved**
6-0	0x00	RW	**Ingress Data Rate Limit for Priority 1 Frames**

Port Ingress Rate Register 2 (0xN412)

Bit	Default	R/W	Description
7	0	RO	**Reserved**
6-0	0x00	RW	**Ingress Data Rate Limit for Priority 2 Frames**

Port Ingress Rate Register 3 (0xN413)

Bit	Default	R/W	Description
7	0	RO	**Reserved**
6-0	0x00	RW	**Ingress Data Rate Limit for Priority 3 Frames**

Port Ingress Rate Register 4 (0xN414)

Bit	Default	R/W	Description
7	0	RO	**Reserved**
6-0	0x00	RW	**Ingress Data Rate Limit for Priority 4 Frames**

Port Ingress Rate Register 5 (0xN415)

Bit	Default	R/W	Description
7	0	RO	**Reserved**
6-0	0x00	RW	**Ingress Data Rate Limit for Priority 5 Frames**

Port Ingress Rate Register 6 (0xN416)

Bit	Default	R/W	Description
7	0	RO	**Reserved**
6-0	0x00	RW	**Ingress Data Rate Limit for Priority 6 Frames**

Port Ingress Rate Register 7 (0xN417)

Bit	Default	R/W	Description
7	0	RO	**Reserved**
6-0	0x00	RW	**Ingress Data Rate Limit for Priority 7 Frames**

There is a procedure to program the ingress rate limit of each priority. The Ingress Rate Register 7 has to be programmed last for each rate limit to take effect. Programming a zero value can turn off the rate limit of the priority, but programming a new value will cause the hardware to use the old one until Ingress Rate Register 7 is programmed.

Port Egress Rate Register 0 (0xN420)

Bit	Default	R/W	Description
7	0	RO	**Reserved**
6-0	0x00	RW	**Egress Data Rate Limit for Queue 0 Frames**

Port Egress Rate Register 1 (0xN421)

Bit	Default	R/W	Description
7	0	RO	**Reserved**
6-0	0x00	RW	**Egress Data Rate Limit for Queue 1 Frames**

Port Egress Rate Register 2 (0xN422)

Bit	Default	R/W	Description
7	0	RO	**Reserved**
6-0	0x00	RW	**Egress Data Rate Limit for Queue 2 Frames**

Port Egress Rate Register 3 (0xN423)

Bit	Default	R/W	Description
7	0	RO	**Reserved**

6-0	0x00	RW	**Egress Data Rate Limit for Queue 3 Frames**

The ingress and egress rate limits can be controlled in each port individually.

The data rate limit value is calculated using the following table. A value of zero means full rate with no limit. A value of 1 to 100 means from 1 Mbps to 100 Mbps. After that a value of 0x65 means 64 Kbps. Each value increment means 64 Kbps increase. For 1000 Mbps connection everything is increased by a factor of 10.

Data Rate	Value
1 Mbps	0x01
10 Mbps	0x0A
100 Mbps	0x64
64 Kbps	0x65
128 Kbps	0x66
192 Kbps	0x67
256 Kbps	0x68
320 Kbps	0x69
384 Kbps	0x6A
448 Kbps	0x6B
512 Kbps	0x6C
576 Kbps	0x6D
640 Kbps	0x6E
704 Kbps	0x6F
768 Kbps	0x70
832 Kbps	0x71
896 Kbps	0x72
960 Kbps	0x73

19.5.2 Sysfs Variables

```
sw/tx_queue_based
sw6/6_rx_prio_rate
sw6/6_tx_prio_rate
sw6/6_limit
```

```
sw6/6_limit_port_based
sw6/6_limit_packet_based
sw6/6_limit_flow_ctrl
sw6/6_limit_cnt_ifg
sw6/6_limit_cnt_pre
sw6/6_rx_p0_rate
sw6/6_rx_p1_rate
sw6/6_rx_p2_rate
sw6/6_rx_p3_rate
sw6/6_rx_p4_rate
sw6/6_rx_p5_rate
sw6/6_rx_p6_rate
sw6/6_rx_p7_rate
sw6/6_tx_q0_rate
sw6/6_tx_q1_rate
sw6/6_tx_q2_rate
sw6/6_tx_q3_rate
```

The `rx_prio_rate` variable is used to enable or disable ingress rate limit function of the switch. Writing 0 means programming ingress rate control registers to zero. Writing 1 means programming whatever values stored in the priority queue rate control variables.

As a result, writing to a priority queue rate control variable with a value other than zero means enabling rate limit unless there is a specific bit for enabling this function.

The `tx_prio_rate` variable is used to enable or disable egress rate limit function of the switch. Writing 0 means programming egress rate control registers to zero. Writing 1 means programming whatever values stored in the priority queue rate control variables.

The legitimate values for queue rate control variables are 0, values in multiple of 64 through 960, and values in multiple of 1000 through 100000.

Note the actual register value will mean different thing when packet is used. For user convenience the driver will keep track of 2 sets of numbers, one for bit and the other for packet.

20 MAC Address Filtering

When a unicast MAC address is not learned in the switch, it is termed as an unknown destination address and the packet is broadcasted to all other ports. To disable this broadcasting the switch can be setup to only forward those unknown destination packets to certain ports.

20.1 Register Description

Switch LUE Unknown Control Register 0 (0x0320 – 0x0323)

Bit	Default	R/W	Description
31	0	RW	**Enable Unknown Unicast Packet Forward**
6-0	0	RW	**Forward Ports**

Switch LUE Unknown Control Register 4 (0x0324 – 0x0327)

Bit	Default	R/W	Description
31	0	RW	**Enable Unknown Multicast Packet Forward**
6-0	0	RW	**Forward Ports**

20.2 Sysfs Variables

```
sw/unk_ucast_fwd
sw/unk_ucast_ports
sw/unk_mcast_fwd
sw/unk_mcast_ports
```

21 ACL

21.1 Register Description

Port ACL Register 0 (0xN600)

Bit	Default	R/W	Description
3-0	0x0	RW	**First Rule Number**

This specifies the entry whose rule is used. It is used together with the Rule Set.

Port ACL Register 1 (0xN601)

Bit	Default	R/W	Description
5-4	0x0	RW	**Mode**
3-2	0x0	RW	**Enable**
1	0	RW	**Source**
0	0	RW	**Equal**

The Mode has 4 configurations: disabled, layer 2, layer 3, or layer 4.

Depending on the Mode the Enable has different types:

Layer 2: Count packets or receive time, match EtherType, match MAC address, or match both.

Layer 3: Match IP address or match source and destination addresses.

Layer 4: Match protocol, match TCP ports, match UDP ports, or match TCP sequence number.

The Source bit indicates the source is used rather than destination in some cases.

The Equal bit indicates the match should be equal.

Port ACL Register 2 (0xN602)

Bit	Default	R/W	Description
7-0	0x00	RW	**MAC Address [47:40]**
7-0	0x00	RW	**IP Address [31:24]**
7-0	0x00	RW	**Maximum Port [15:8]**
7-0	0x00	RW	**TCP Sequence Number [31:24]**

Registers 2, 3, 4, 5, 6, and 7 are used to store the MAC address in layer 2 mode.

Registers 2, 3, 4, and 5 are used to store the IP address in layer 3 mode.

Registers 2, 3, 4, and 5 are used to store the TCP sequence number in layer 3 mode.

Registers 2 and 3 are used to store the maximum port number in layer 3 mode.

Port ACL Register 3 (0xN603)

Bit	Default	R/W	Description
7-0	0x00	RW	**MAC Address [39:32]**
7-0	0x00	RW	**IP Address [23:16]**
7-0	0x00	RW	**Maximum Port [7:0]**
7-0	0x00	RW	**TCP Sequence Number [23:16]**

Port ACL Register 4 (0xN604)

Bit	Default	R/W	Description
7-0	0x00	RW	**MAC Address [31:24]**
7-0	0x00	RW	**IP Address [15:8]**
7-0	0x00	RW	**Minimum Port [15:8]**
7-0	0x00	RW	**TCP Sequence Number [15:8]**

Registers 4 and 5 are used to store the minimum port number in layer 3 mode.

Port ACL Register 5 (0xN605)

Bit	Default	R/W	Description
7-0	0x00	RW	**MAC Address [23:16]**
7-0	0x00	RW	**IP Address [7:0]**
7-0	0x00	RW	**Minimum Port [7:0]**
7-0	0x00	RW	**TCP Sequence Number [7:0]**

Port ACL Register 6 (0xN606)

Bit	Default	R/W	Description
7-0	0x00	RW	**MAC Address [15:8]**
7-0	0x00	RW	**IP Address Mask [31:24]**
2-1	0x0	RW	**Port Mode**
0	0	RW	**Protocol [7]**

Registers 6, 7, 8, and 9 are used to store the IP address mask in layer 3 mode.

The Port Mode in layer 4 mode has 4 configurations: disabled, match either port, match ports in range, and match ports out of range.

Port ACL Register 7 (0xN607)

Bit	Default	R/W	Description
7-0	0x00	RW	**MAC Address [7:0]**
7-0	0x00	RW	**IP Address Mask [23:16]**
7-1	0x00	RW	**Protocol [6:0]**
0	0	RW	**TCP Flag Enable**

The Protocol and TCP Flag Enable bit are used in layer 4 mode.

Port ACL Register 8 (0xN608)

Bit	Default	R/W	Description
7-0	0x00	RW	**EtherType [15:8]**
7-0	0x00	RW	**IP Address Mask [15:8]**
7-0	0x00	RW	**TCP Flag Mask**

Registers 8 and 9 are used to store the EtherType in layer 2 mode.

The TCP Flag Mask is used in layer 4 mode.

Port ACL Register 9 (0xN609)

Bit	Default	R/W	Description
7-0	0x00	RW	**EtherType [7:0]**
7-0	0x00	RW	**IP Address Mask [7:0]**
7-0	0x00	RW	**TCP Flag**

The TCP Flag is used in layer 4 mode.

Port ACL Register A (0xN60A)

Bit	Default	R/W	Description
7-0	0x0	RW	**Count [10:3]**
7-6	0x0	RW	**Priority Mode**
5-3	0x0	RW	**Priority**
2	0	RW	**Remark Priority Enable**
1-0	0x0	RW	**Remark Priority [2:1]**

The Count is an 11-bit number which can mean number or time.

The Priority Mode has 4 configurations: disabled, match higher, match lower, or replace.

The Priority is used by the Priority Mode.

The Remark Priority Enable bit allows VLAN priority to be changed by Remark Priority.

Port ACL Register B (0xN60B)

Bit	Default	R/W	Description

7-5	0x0	RW	**Count [2:0]**
7	0x0	RW	**Remark Priority [0]**
6-5	0x0	RW	**Map Mode**

The Map Mode has 4 configurations: disabled, OR operation, AND operation, and replace.

Port ACL Register C (0xN60C)

Bit	Default	R/W	Description
7-0	0	RW	**Reserved**

Port ACL Register D (0xN60D)

Bit	Default	R/W	Description
6	0	RW	**Msec Unit**
5	0	RW	**Interrupt Mode**
6-0	0x00	RW	**Port Map**

The Count when used as time can be in microsecond or millisecond unit.

The Interrupt Mode specifies whether the Count is interpreted as number of packets received or time passed after initial packet receive.

The Port Map is used by the Map Mode.

Port ACL Register E (0xN60E)

Bit	Default	R/W	Description
7-0	0x00	RW	**Rule Set [15:8]**

The Rule Set can group several ACL entries together. Each rule has to be matched before they are all considered matched so that the first rule takes effect.

Port ACL Register F (0xN60F)

Bit	Default	R/W	Description
7-0	0x00	RW	**Rule Set [7:0]**

Port ACL Byte Enable MSB (0xN610)

Bit	Default	R/W	Description

7-0	0x00	RW	**Byte Enable [15:8]**

Port ACL Byte Enable LSB (0xN611)

Bit	Default	R/W	Description
7-0	0x00	RW	**Byte Enable [7:0]**

As the ACL entry has three distinct programmed functions it is efficient to use byte enables to write only the programmed part. However in current implementation the bits do not always match the full byte. Observation is:

First byte is not controllable by byte enable. Maximum value is 0x0F.

Maximum value of byte 2 is 0x3F.

Maximum value of byte 12 is 0xE0.

Byte 13 is completely ignored. What is written is read back.

Maximum value of byte 14 is 0x7F.

For user convenience a value of 0x003C for ACL_ACTION_ENABLE is defined for programming action rule, a value of 0x8003 for ACL_RULESET_ENABLE is defined for programming ruleset, and a value of 0x7FC0 for ACL_MATCH_ENABLE is defined for programming matching rule.

Port ACL Control Register 0 (0xN612)

Bit	Default	R/W	Description
6	0	RO	**Write Completed**
5	0	RO	**Read Completed**
4	0	RW	**Write**
3-0	0x0	RW	**ACL Index**

Port ACL Control Register 1 (0xN613)

Bit	Default	R/W	Description
0	0	RW	**Force DLR Miss**

Port MRI Authentication Control Register (0xN803)

Bit	Default	R/W	Description
2	0	RW	**ACL Enable**

When ACL is disabled the ACL table is not accessible. For convenience the driver automatically enables ACL when users change or look up the ACL rules.

21.2 Sysfs Variables

```
sw6/6_acl_table
sw6/6_acl_act_index
sw6/6_acl_prio
sw6/6_acl_prio_mode
sw6/6_acl_map_mode
sw6/6_acl_vlan_prio
sw6/6_acl_vlan_prio_replace
sw6/6_acl_act
sw6/6_acl_rule_index
sw6/6_acl_first_rule
sw6/6_acl_ruleset
sw6/6_acl_index
sw6/6_acl_info
sw6/6_acl_addr
sw6/6_acl_cnt
sw6/6_acl_enable
sw6/6_acl_equal
sw6/6_acl_intr_mode
sw6/6_acl_ip_addr
sw6/6_acl_ip_mask
sw6/6_acl_max_port
sw6/6_acl_min_port
sw6/6_acl_msec
sw6/6_acl_port_mode
sw6/6_acl_ports
sw6/6_acl_protocol
sw6/6_acl_seqnum
sw6/6_acl_src
sw6/6_acl_tcp_flag
sw6/6_acl_tcp_flag_enable
sw6/6_acl_tcp_flag_mask
sw6/6_acl_type
sw6/6_acl_mode
sw6/6_acl
```

The `acl_table` variable displays only valid entries in the port ACL table. The format is varied depending on the mode.

"<entry>: <MAC address> <EtherType> <count> <intr> <msec> <enable> <source> <equal> <1>."

"<entry>: <IP address> <IP address mask> <enable> <source> <equal> <2>."

"<entry>: <port mode> <protocol> <sequence number> <TCP flag> <enable> <source> <equal> <3>."

The `acl_index` variable is used to specify the index for programming matching rule.

The `acl_rule_index` variable is used to specify the index for programming ruleset.

The `acl_act_index` variable is used to specify the index for programming action rule.

The `acl_info` variable is used to show all the information of the entry. The last number in bracket indicates the state of the entry: 0, 1, 2, and 3 for the actual mode, or 8 for changes that do not reflect the actual contents in the table. It can be written to read the entry directly.

All the other variables allow the user to change the contents of the entry.

Writing to the `acl_mode` variable will write the entry to the ACL table to update matching rule. Therefore, it should be the last one to write.

Writing to the `acl_ruleset` variable will write the entry to the ACL table to update ruleset. Therefore, it should be the last one to write to modify that.

Writing to the `acl_act` variable with value 1 will write the entry to the ACL table to update the action rule. Therefore, it should be the last one to write. Writing 0 will read the entry directly.

21.3 ACL Usage

ACL provides a filtering mechanism so that packets with certain formats can be matched and forwarded to specific ports or dropped. The feature can be enabled individually in each port. Although the host port can use ACL, the function is typically not used as the host can control the forwarding itself.

The ACL programming in the switch is broken into 3 fields: the comparison, the action, and the ruleset. Each port has 16 entries to hold this information. As the comparison and the action fields can be shared among these 16 entries, it is the number of rulesets that actually limits the number of ACL rules in operation.

Because the information in the entry can be set separately, it is not necessary to always write the whole entry to the hardware to change the operation. Byte enable is used so that only specific bytes are written. So there are certain byte enable formats to change the comparison field, the action field, or the ruleset.

The switch driver has provided the necessary functions to program the switch to operate ACL. The developers just fill in the rules for comparison and set the action. For users the sysfs system in the Linux kernel can be used to program ACL entry in the ACL table.

The ACL function can be controlled by the sysfs file `#_acl`:

```
echo 1 > 0_acl
echo 0 > 0_acl
```

Normally ACL is turned on as it is required to access the ACL table. When it is manually disabled by users the switch driver will enable it temporarily when the ACL table is being accessed. The driver will show a warning when the ACL table is read to remind the users that ACL is disabled.

21.3.1 ACL Ruleset

As the comparison field is complicated, the easiest field is discussed first.

The ruleset field selects which ACL rules to compare and which action to execute when all the rules match. The `ruleset` parameter itself is a 16-bit number containing the bits of the rules to be compared. For a simple rule the number of bits is exactly 1, namely in the format of (1 << n), where n is the ACL ruleset index, starting at 0. For many rules that need to be match, the ruleset contains the combination of the bits of those rules. A ruleset with zero means the ruleset is not used.

The `first_rule` parameter indicates the action to execute.

`#_acl_ruleset` – view or set the ruleset
`#_acl_first_rule` – view or change the first rule
`#_acl_rule_index` – view or change the ruleset index

21.3.2 ACL Action

The action field contains 3 actions that can affect how the packet is forwarded inside the switch. The first one is the port mapping mode, which has 4 options: do nothing, OR the defined ports, AND the defined ports, and replace with defined ports. The `ports` parameter defines the port bitmap that is acted by the mapping mode. Typically the replace operation is used to either forward the frame to specific port like the host port or drop the frame.

The second action is to adjust the priority. There are 4 options: do nothing, change if higher priority, change if lower priority, and replace priority. The `priority` parameter defines the value that is acted by the priority change mode.

The last action is to replace the VLAN priority. It is a simple operation where the `VLAN priority` parameter will just be used and put in the VLAN tag.

`#_acl_map_mode` – view or change the port mapping mode
`#_acl_ports` – view or change the port bitmap
`#_acl_prio_mode` – view or change the priority adjustment
`#_acl_prio` – view or change the priority
`#_acl_vlan_prio_replace` – view or change the VLAN priority set function
`#_acl_vlan_prio` – view or change the VLAN priority
`#_acl_acl_act` – view or set the action
`#_acl_acl_act_index` – view or change the action index

21.3.3 ACL Comparison

The comparison field is broken into 4 categories which act on different network layers. The `mode` parameter specifies these layers: 0 means disabled, 1 for layer 2, 2 for layer 3, and 3 for layer 4.

21.3.3.1 Layer 2

The destination and source address can be matched. The EtherType in the frame can also be matched. The `src` parameter is used to specify whether destination or source address is matched. The `equal` parameter is used to specify matching or not. There is also a special function to count up or down the number of received frames. After a condition is met an interrupt is triggered.

Counting up means the switch keeps track of received frames until the number matches. A count of zero means the interrupt is triggered every time the frame is received. Counting down means the switch first receives a frame and will trigger an interrupt if the same frame is not received within the time set. The default time unit is in microsecond. There is a `millisecond` parameter to change the time unit to millisecond. As the interrupt does not tell which rule triggers it this function can only be used in one rule. Actually there is only one counter so this function cannot be used multiple times.

The `count` parameter has 11-bit resolution, so the maximum time allowed is 2047 milliseconds. Note the `count` parameter overlaps the space for action field, so an action cannot be set in the same entry.

The `enable` parameter specifies the match operation: 0 for count, 1 for EtherType, 2 for MAC address, and 3 for both EtherType and MAC address.

The single rule can only match one MAC address. To match both destination and source addresses two rules have to be used.

`#_acl_addr` – view or change the MAC address. The source or destination type is by `src` parameter
`#_acl_type` – view or change the EtherType
`#_acl_equal` – view or change the `equal` parameter.
`#_acl_src` – view or change the source address parameter
`#_acl_cnt` – view or change the `count` parameter
`#_acl_intr_mode` – view or change the count down/up interrupt mode
`#_acl_msec` – view or change the `millisecond` parameter which is used in count down operation
`#_acl_enable` – view or change the `enable` parameter
`#_acl_mode` – view or set the `mode`. This should be set last to program the rule

21.3.3.2 Layer 3

Only IPv4 addresses can be matched. There are `ip_addr` and `ip_mask` parameters to provide the necessary information. The hardware uses the algorithm ip_addr OR ip_mask == rx_ip_addr OR ip_mask for comparison, so it is quite unintuitive to set the `ip_mask` to 0.0.0.0 for exact match. The `src` parameter is used to specify whether destination or source IP address is matched. The `equal` parameter is used to specify matching or not.

There is a function to match any destination and source IP address to make sure they are not the same.

The `enable` parameter specifies these match operations: 1 for IP address and 2 for source and destination addresses.

`#_acl_ip_addr` – view or change the IP address
`#_acl_ip_mask` – view or change the IP address mask
`#_acl_equal` – view or change the `equal` parameter
`#_acl_src` – view or change the source address parameter
`#_acl_enable` – view or change the `enable` parameter
`#_acl_mode` – view or set the `mode`. This should be set last to program the rule

21.3.3.3 Layer 4

Only IPv4 is supported for now. This mode can be used to match IP protocol, TCP/UDP ports, TCP sequence number and flags.

The `IP protocol` is a simple match with the protocol number. The port matching mode can be 0 for disabled, 1 for either, 2 for in range, and 3 for out of range. The minimum and maximum ports specify the range to be matched. The `src` parameter is used to specify whether destination

or source port is matched. The TCP sequence number is a simple match with a number. The `TCP flag enable` parameter can be enabled to also match TCP flag which is indicated by the `tcp_flag` and `tcp_flag_mask` parameters.

`#_acl_protocol` – view or change the protocol number
`#_acl_port_mode` – view or change the port mode
`#_acl_min_port` – view or change the minimum port parameter
`#_acl_max_port` – view or change the maximum port parameter
`#_acl_equal` – view or change the `equal` parameter
`#_acl_src` – view or change the `src` parameter
`#_acl_seqnum` – view or change the TCP sequence number parameter
`#_acl_tcp_flag` – view or change the `tcp_flag` parameter
`#_acl_tcp_flag_mask` – view or change the `tcp_flag_mask` parameter
`#_acl_tcp_flag_enable` – view or change the `tcp_flag_enable` parameter
`#_acl_enable` – view or change the `enable` parameter
`#_acl_mode` – view or set the `mode`. This should be set last to program the rule

21.3.4 ACL Sysfs Usage Examples

There are some common sysfs files that apply to all the rules.

`#_acl_index` – view or change the index
`#_acl_info` – view the current entry
`#_acl_table` – view the ACL table
`#_acl` – view or enable ACL

As the ACL information is changed depending on the `mode` and `enable` parameters, the driver tries to show only information that are relevant. If the mode is not really set then the information shown does not mean much.

Typically the line for comparison rule looks like this:

```
0: 01:80:C2:00:00:03-888e c:0.0 E:3 S:0 Q:1 [1]
```

The first number is the index. Depending on the mode the information after that will be different. For mode 1, which is shown at the end, the `MAC address` is shown and then `EtherType`. Count up/down is indicated next. If this function is not set the "c" character will be in lowercase. Otherwise it is in uppercase. The number indicates it is count up if it is 1. The next number is the `count`. If count down is used "ms" will be shown to indicate millisecond is the time unit, or "us" to indicate microsecond. Next is the `enable` parameter. Showing in uppercase means it is being used. The `src` parameter is next, and then the `equal` parameter. The number inside the brackets indicates the `mode`. When the number is 8 it indicates the rule has been modified but not yet written to hardware. As it requires updating the `mode` parameter

to program the rule, the number will not be 8 when displayed in the ACT table.

The line for comparison rule in layer 3 looks like this:

```
3: 192.168.0.0:0.0.0.255 E:1 S:0 Q:1 [2]
```

The first number is the index. Next are the `IP address` and `mask`. Next is the `enable` parameter. The `src` parameter is next, and then the `equal` parameter. The number inside the brackets indicates the `mode`. When the number is 8 it indicates the rule has been modified but not yet written to hardware.

The line for comparison rule in layer 4 looks like this:

```
1: 1= 66d- 714 0x0 s:00000000 f:0=0:0 E:2 S:0 q:0 [3]
```

The first number is the index. Next are the `port mode`, `minimum` and `maximum ports`. The `IP protocol` number is next. Then the TCP sequence number and `flag enable`, `flag value` and `flag mask`. Next is the `enable` parameter. The `src` parameter is next, and then the `equal` parameter. The number inside the brackets indicates the `mode`. When the number is 8 it indicates the rule has been modified but not yet written to hardware.

The line for ruleset looks like this:

```
0: 0:0001 [1]
```

The first number is the index. After that it is the `first rule`, which indicates which action rule to use. The `ruleset` bitmap indicating which rules should be matched is next. The number 1 inside the brackets indicates the rule is valid. Otherwise a number 8 is used to indicate the rule has been modified but not yet written to hardware.

The line for action rule looks like this:

```
0: p:0=0 v:0=0 3=0000 [1]
```

The first number is the index. Next are the `priority adjustment mode` and the `priority`. After that it is `VLAN priority replacement` and the `VLAN priority`. After that it is the `port forwarding mode` and the `port bitmap`. The number 1 inside the brackets indicates the rule is valid. Otherwise a number 8 is used to indicate the rule has been modified but not yet written to hardware.

The ACL table shows all the entries which mode is not disabled, ruleset not empty, and action doing something or used by a ruleset.

An example of handling beacons in DLR:

```
rules:
```

```
c: 00:10:A1:94:77:05-80e1 c:0.0 E:3 S:1 Q:1 [1]
d: 01:21:6C:00:00:01-80e1 c:0.0 E:3 S:0 Q:1 [1]
e: 01:21:6C:00:00:01-80e1 C:0.1960 us E:0 S:0 Q:1 [1]

rulesets:
c: c:3000 [1]
d: f:4000 [1]

actions:
c: p:0=0 v:0=0 3=0000 [1]
f: p:0=0 v:0=0 0=0000 [1]
```

The device drops all beacons sent by supervisor 00:10:A1:94:77:05. It also asks the hardware to trigger an interrupt if the beacon is not received within 1960 microseconds. Note the action for that ruleset is not doing anything.

An example of using 802.1X Authentication:

```
rules:
0: 01:80:C2:00:00:03-888e c:0.0 E:3 S:0 Q:1 [1]
1: 1= 66d- 714 0x0 s:00000000 f:0=0:0 E:2 S:0 q:0 [3]
2: 00:00:00:00:00:00-0806 c:00 E:1 s:0 Q:1 [1]

rulesets:
0: 0:0001 [1]
1: 1:0002 [1]
2: 1:0004 [1]

actions:
0: p:0=0 v:0=0 0=0000 [1]
1: p:0=0 v:0=0 3=0020 [1]
```

The device allows PAE frames to go to the host when the port is blocked. Note the multicast address 01:80:C2:00:00:03 is already set in the reserved multicast table. RADIUS messages using UDP packets can also go to host. ARP frames are passed to the host so that ARP response from the RADIUS server can be processed.

21.3.4.1 Filter 802.3 frame

```
# find an unused index
echo 0 > #_acl_index
echo 01:80:C2:00:00:03 > #_acl_addr
# EtherType is always interpreted as a hexadecimal number
echo 888e > #_acl_type
# indicate destination address
echo 0 > #_acl_src
# indicate matching
echo 1 > #_acl_equal
# select the proper enable parameter to match both
echo 3 > #_acl_enable
# select the mode and program the entry to ACL table
echo 1 > #_acl_mode

# find an unused index
echo 0 > 0_acl_act_index
echo 3 > 0_acl_map_mode
# forward to last port; use zero to drop
echo 0x40 > 0_acl_ports
# program the entry to ACL table
echo 1 > 0_acl_act

# basic action to drop frame
echo 1 > 0_acl_act_index
echo 3 > 0_acl_map_mode
echo 0 > 0_acl_ports
echo 1 > 0_acl_act

# find an unused index
echo 0 > 0_acl_rule_index
# select which action to execute
echo 0 > 0_acl_first_rule
# set the ruleset and program the entry to ACL table
echo 0x1 > 0_acl_ruleset

# view the current rule
cat #_acl_info
# view the current action
cat #_acl_act
# view the current ruleset
cat #_acl_ruleset
# view the ACL table
cat #_acl_table
```

21.3.4.2 Filter IP packets

```
# find an unused index
echo 1 > #_acl_index
# match IP address in subnet 255.255.255.0
echo 192.168.0.1 > #_acl_ip_addr
echo 0.0.0.255 > #_acl_ip_mask
# indicate source address
echo 1 > #_acl_src
# indicate matching
echo 1 > #_acl_equal
# select the proper enable parameter to match IP address
echo 1 > #_acl_enable
# select the mode and program the entry to ACL table
echo 2 > #_acl_mode

# find an unused index
echo 1 > 0_acl_rule_index
# select which action to execute
echo 0 > 0_acl_first_rule
# set the ruleset and program the entry to ACL table
echo 0x2 > 0_acl_ruleset
```

21.3.4.3 Filter UDP packets

```
# find an unused index
echo 2 > #_acl_index
# match UDP packets
echo 17 > #_acl_protocol
# indicate matching
echo 1 > #_acl_equal
# select the proper enable parameter to match IP protocol
echo 0 > #_acl_enable
# select the mode and program the entry to ACL table
echo 3 > #_acl_mode

# find an unused index
echo 2 > 0_acl_rule_index
# select which action to execute
```

```
echo 0 > 0_acl_first_rule
# set the ruleset and program the entry to ACL table
echo 0x4 > 0_acl_ruleset
```

21.3.4.4 Filter RADIUS messages

```
# find an unused index
echo 3 > #_acl_index
# match UDP RADIUS messages
echo 1645 > #_acl_min_port
echo 1812 > #_acl_max_port
echo 1 > 0_acl_port_mode
# indicate destination port
echo 0 > #_acl_src
# select the proper enable parameter to match UDP ports
echo 1 > #_acl_enable
# select the mode and program the entry to ACL table
echo 3 > #_acl_mode

# find an unused index
echo 3 > 0_acl_rule_index
# select which action to execute
echo 0 > 0_acl_first_rule
# set the ruleset and program the entry to ACL table
echo 0x8 > 0_acl_ruleset
```

21.3.4.5 Combine rules to filter RADIUS messages from certain IP address

```
# find an unused index
echo 4 > 0_acl_rule_index
# select which action to execute
echo 1 > 0_acl_first_rule
# set the ruleset and program the entry to ACL table
echo 0xA > 0_acl_ruleset
```

21.3.4.6 Trigger interrupt if condition is met

```
# find an unused index
```

```
echo 0xe > 0_acl_rule_index
echo 01:80:C2:00:00:03 > #_acl_addr
# EtherType is always interpreted as a hexadecimal number
echo 888e > #_acl_type
# indicate destination address
echo 0 > #_acl_src
# indicate matching
echo 1 > #_acl_equal
# trigger interrupt if frame is not received within 2 seconds
echo 2000 > 0_acl_cnt
echo 1 > 0_acl_msec
echo 0 > 0_acl_intr_mode
# select the proper enable parameter to match both
echo 0 > #_acl_enable
# select the mode and program the entry to ACL table
echo 1 > #_acl_mode
# select which action to execute
echo 0xf > 0_acl_first_rule
# set the ruleset and program the entry to ACL table
echo 0x4000 > 0_acl_ruleset
```

Note if this function is set the action rule should not be used in the same index.

This special function is only intended to be used internally. For verification purpose user can set this up and observe if the interrupt is triggered. To do that the driver has to be told to report the ACL interrupt. First read the `overrides` file. Note the value and add 0x00020000 to it and write back the value.

```
cat sw/overrides
echo 0x00020004 > sw/overrides
```

The driver will print out a statement showing the interrupt from the port.

22 802.1X Authentication

22.1 Register Description

Port MRI Authentication Control Register (0xN803)

Bit	Default	R/W	Description

1-0	0x0	RW	**Authentication Mode**

The Authentication Mode can be pass, block, or trap. When port based 802.1X Authentication is enabled the port is put into block mode while ACL rules are configured as shown above so the Authenticator can pass the user credentials to the RADIUS server to be verified. When that is successful the port is put back into pass mode.

Port LUE Control Register (0xNB00)

Bit	Default	R/W	Description
4	0	RW	**MAC Based 802.1X Enable**

The MAC based 802.1X Authentication uses a different method to allow traffic into a switch. Port based 802.1X Authentication is not secure enough such that another switch can be put between the authenticated user and the switch so that another unauthenticated user can gain access to the switch. MAC based 802.1X Authentication allows frames with MAC addresses that are known in the switch to enter the switch. The MAC addresses are stored in the look up tables. As such learning is not allowed and should be disabled for all ports.

22.2 Sysfs Variables

```
sw6/6_authen_mode
sw6/6_mac_802_1x
```

23 Policing and WRED

23.1 Register Description

Switch MRI Control Register 8 (0x0378)

Bit	Default	R/W	Description
7-6	0b00	RW	**No Color**
5-4	0b11	RW	**DSCP Value of Red Color**
3-2	0b10	RW	**DSCP Value of Yellow Color**
1-0	0b01	RW	**DSCP Value of Green Color**

This register defines the color mapping of DSCP value from the packet.

Port MRI Index Register (0xN804 – 0xN807)

Bit	Default	R/W	Description
18-16	0x0	RW	**MRI Port Index**
1-0	0x0	RW	**MRI Queue Index**

Port MRI TC Map Register (0xN808 – 0xN80B)

Bit	Default	R/W	Description
31-28	0x3	RW	**Regenerated Priority for Priority 7**
27-24	0x3	RW	**Regenerated Priority for Priority 6**
23-20	0x2	RW	**Regenerated Priority for Priority 5**
19-16	0x2	RW	**Regenerated Priority for Priority 4**
15-12	0x1	RW	**Regenerated Priority for Priority 3**
11-8	0x1	RW	**Regenerated Priority for Priority 2**
7-4	0x0	RW	**Regenerated Priority for Priority 1**
3-0	0x0	RW	**Regenerated Priority for Priority 0**

For AVB this map is changed to 0x11113200 as traffic Class A uses priority 3 and Class B uses priority 2.

Port MRI Police Control Register (0xN80C – 0xN80F)

Bit	Default	R/W	Description
10	0	RW	**Police Drop All**
9-8	0x0	RW	**Police Packet Type**
7	0	RW	**Port Based Policing**
6-5	0x1	RW	**Color for Non-DSCP Frame**
4	0	RW	**DSCP Color Mark Enable**
3	0	RW	**DSCP Color Remap Enable**
2	0	RW	**Allow SRP Drop**
1	0	RW	**Police Color Not Aware**
0	0	RW	**Police Enable**

The Police Packet Type is used with WRED_PMON to read the specific type of packets. It is 0

for dropped packets, 1 for green packets, 2 for yellow packets, or 3 for red packets.

Port MRI Police Color Remap Register 0 (0xN810 – 0xN813)

Bit	Default	R/W	Description
31-30	0b11	RW	**DSCP Color 63**
29-28	0b11	RW	**DSCP Color 62**
27-26	0b10	RW	**DSCP Color 61**
25-24	0b10	RW	**DSCP Color 60**
23-22	0b01	RW	**DSCP Color 59**
21-20	0b01	RW	**DSCP Color 58**
19-18	0b00	RW	**DSCP Color 57**
17-16	0b00	RW	**DSCP Color 56**
15-14	0b11	RW	**DSCP Color 55**
13-12	0b11	RW	**DSCP Color 54**
11-10	0b10	RW	**DSCP Color 53**
9-8	0b10	RW	**DSCP Color 52**
7-6	0b01	RW	**DSCP Color 51**
5-4	0b01	RW	**DSCP Color 50**
3-2	0b00	RW	**DSCP Color 49**
1-0	0b00	RW	**DSCP Color 48**

Port MRI Police Color Remap Register 1 (0xN814 – 0xN817)

Bit	Default	R/W	Description
31-30	0b11	RW	**DSCP Color 47**
29-28	0b11	RW	**DSCP Color 46**
27-26	0b10	RW	**DSCP Color 45**
25-24	0b10	RW	**DSCP Color 44**
23-22	0b01	RW	**DSCP Color 43**
21-20	0b01	RW	**DSCP Color 42**
19-18	0b00	RW	**DSCP Color 41**
17-16	0b00	RW	**DSCP Color 40**
15-14	0b11	RW	**DSCP Color 39**
13-12	0b11	RW	**DSCP Color 38**
11-10	0b10	RW	**DSCP Color 37**

9-8	0b10	RW	**DSCP Color 36**
7-6	0b01	RW	**DSCP Color 35**
5-4	0b01	RW	**DSCP Color 34**
3-2	0b00	RW	**DSCP Color 33**
1-0	0b00	RW	**DSCP Color 32**

Port MRI Police Color Remap Register 2 (0xN818 – 0xN81B)

Bit	Default	R/W	Description
31-30	0b11	RW	**DSCP Color 31**
29-28	0b11	RW	**DSCP Color 30**
27-26	0b10	RW	**DSCP Color 29**
25-24	0b10	RW	**DSCP Color 28**
23-22	0b01	RW	**DSCP Color 27**
21-20	0b01	RW	**DSCP Color 26**
19-18	0b00	RW	**DSCP Color 25**
17-16	0b00	RW	**DSCP Color 24**
15-14	0b11	RW	**DSCP Color 23**
13-12	0b11	RW	**DSCP Color 22**
11-10	0b10	RW	**DSCP Color 21**
9-8	0b10	RW	**DSCP Color 20**
7-6	0b01	RW	**DSCP Color 19**
5-4	0b01	RW	**DSCP Color 18**
3-2	0b00	RW	**DSCP Color 17**
1-0	0b00	RW	**DSCP Color 16**

Port MRI Police Color Remap Register 3 (0xN81C – 0xN81F)

Bit	Default	R/W	Description
31-30	0b11	RW	**DSCP Color 15**
29-28	0b11	RW	**DSCP Color 14**
27-26	0b10	RW	**DSCP Color 13**
25-24	0b10	RW	**DSCP Color 12**
23-22	0b01	RW	**DSCP Color 11**
21-20	0b01	RW	**DSCP Color 10**
19-18	0b00	RW	**DSCP Color 9**

17-16	0b00	RW	**DSCP Color 8**
15-14	0b11	RW	**DSCP Color 7**
13-12	0b11	RW	**DSCP Color 6**
11-10	0b10	RW	**DSCP Color 5**
9-8	0b10	RW	**DSCP Color 4**
7-6	0b01	RW	**DSCP Color 3**
5-4	0b01	RW	**DSCP Color 2**
3-2	0b00	RW	**DSCP Color 1**
1-0	0b00	RW	**DSCP Color 0**

Port MRI Police Queue Rate Register (0xN820 – 0xN823)

Bit	Default	R/W	Description
31-16	0x1000	RW	**Committed Information Rate**
15-0	0x2000	RW	**Peak Information Rate**

Port MRI Police Queue Size Register (0xN824 – 0xN827)

Bit	Default	R/W	Description
31-16	0x1000	RW	**Committed Burst Size**
15-0	0x3000	RW	**Peak Burst Size**

Port MRI WRED PM Control Register 0 (0xN830 – 0xN833)

Bit	Default	R/W	Description
26-16	0x400	RW	**WRED Maximum Threshold of PM Buffer**
10-0	0x080	RW	**WRED Minimum Threshold of PM Buffer**

Port MRI WRED PM Control Register 1 (0xN834 – 0xN837)

Bit	Default	R/W	Description
26-16	0x020	RW	**Probability Multiplier of PM Buffer**
10-0	0x0	RO	**Average Queue Size**

Port MRI WRED Queue Control Register 0 (0xN840 – 0xN843)

Bit	Default	R/W	Description

26-16	0x080	RW	**WRED Maximum Threshold of Queue Connection to Port**
10-0	0x009	RW	**WRED Minimum Threshold for Queue Connection to Port**

Port MRI WRED Queue Control Register 1 (0xN844 – 0xN847)

Bit	Default	R/W	Description
26-16	0x010	RW	**Probability Multiplier for Queue Connection to Port**
10-0	0x0	RO	**Average Queue Size for Queue Connection to Port**

Port MRI WRED Queue PMON Register (0xN848 – 0xN84B)

Bit	Default	R/W	Description
31	0	RW	**Random Drop Enable**
30	0	RW	**Flush PMON Counters**
29	0	RW	**WRED Drop Green/Yellow/Red Disable**
28	0	RW	**WRED Drop Yellow/Red Disable**
27	0	RW	**WRED Drop Red Disable**
26	0	RW	**WRED Drop All**
23-0	0x0	RO	**Packet Events for Queue Connection to Port**

23.2 Sysfs Variables

```
sw6/6_color_map
sw6/6_tc_map
sw6/6_p_index
sw6/6_q_index
sw6/6_police
sw6/6_police_type
sw6/6_police_drop_all
sw6/6_police_port_based
sw6/6_non_dscp_color
sw6/6_color_aware
sw6/6_color_mark
sw6/6_color_remap
sw6/6_drop_srp
sw6/6_q_cir
sw6/6_q_pir
sw6/6_q_cbs
sw6/6_q_pbs
```

```
sw6/6_wred_max
sw6/6_wred_min
sw6/6_wred_multiplier
sw6/6_wred_avg_size
sw6/6_wred_q_max
sw6/6_wred_q_min
sw6/6_wred_q_multiplier
sw6/6_wred_q_avg_size
sw6/6_wred_random_drop
sw6/6_wred_drop_gyr
sw6/6_wred_drop_yr
sw6/6_wred_drop_r
sw6/6_wred_drop_all
sw6/6_wred_q_pmon
```

The `p_index` and `q_index` variables are used to set the port and queue indexes in some cases, like `q_cir`, `q_pir`, `q_cbs`, `q_pbs`, `wred_q_max`, `wred_q_min`, `wred_q_multiplier`, `wred_q_avg_size`, and `wred_q_pmon`.

24 Credit Shaping

24.1 Register Description

Switch AVB Strategy Register (0x030E – 0x030F)

Bit	Default	R/W	Description
1	0	RW	**Shaping Credit Deduction on Both Data and Preamble**
0	1	RW	**Policing Credit Deduction on Both Data and Preamble**

Port MTI Queue Index Register (0xN900)

Bit	Default	R/W	Description
1-0	0	RW	**Queue Index**

The queue index is used whenever one of the 4 queues needs to be accessed.

Port MTI Queue Control Register 0 (0xN914)

Bit	Default	R/W	Description
7-6	2	RW	**Schedule Mode**
5-4	0	RW	**Shaping Enable**

The Schedule Mode is 0 for strict priority, or 2 for WRR.

The Shaping Enable is 0 for disabled, or 1 for SRP enabled.

Port MTI Queue Control Register 1 (0xN915)

Bit	Default	R/W	Description
6-0	1	RW	**Queue Ratio**

The default value is 1 for all 4 queues. For WWR scheduling it is better to program 1 for queue 1, 2 for queue 2, 4 for queue 3, and 8 for queue 4 as this is the default setting from the other KSZ switches.

Port MTI Queue Control Register 2 (0xN916 – 0xN917)

Bit	Default	R/W	Description
15-0	0x0534	RW	**Shaper Credit High Watermark**

The credit high watermark is used to limit the effect of a large frame to reduce the bandwidth. The formula is the maxim interference size multiples with the idle slope, which is the bandwidth percentage. The maximum interference size is the biggest frame size encountered in the network traffic. Typically it is the maximum Ethernet frame size. The actual size is increased by 24 bytes to account for preamble, CRC, and IFG. Note the AVB Strategy register can make this not necessary. When set to a low enough value the allowed bandwidth is reduced a little bit, so it is better to use a big number, although the actual effect is unknown.

Port MTI Queue Control Register 3 (0xN918 – 0xN919)

Bit	Default	R/W	Description
15-0	0x05f2	RW	**Shaper Credit Low Watermark**

The credit low watermark is used to cap the bandwidth. The formula is the stream maximum frame size multiples with the send slope, which is 1 minus the idle slope. The actual size is increased by 42 bytes to account for preamble, VLAN header, CRC, and IFG. The bandwidth can increase if this value decreases from a certain mark, so maybe it is better to set this value to a high number and let the credit increment register to determine the allowed bandwidth.

Port MTI Queue Control Register 4 (0xN91A – 0xN91B)

Bit	Default	R/W	Description
15-0	0x2000	RW	**Shaper Credit Increment**

The credit increment is used to limit or guarantee the bandwidth of the transmit queue. The formula is simply the bandwidth percentage of the network speed multiples with 65536. So the maximum 100% value is 65535, and the minimum value is 1 as zero value may not be allowed. Note the network speed can change from 1 gigabit to 100 Mbit and so this value needs to be adjusted to reflect the actual bandwidth.

Port Control Register 1 (0xN021)

Bit	Default	R/W	Description
1-0	0x0	RW	**SRP Enable**

The SRP Enable is 0 for disabled, 1 for SRP_1 enabled, and 2 for SRP_2 enabled.

The function of this register is unclear. Basically it adds a VLAN tag as specified by the associated AVB SR class VID registers when the frame is untagged.

24.2 Sysfs Variables

```
sw6/6_q_index
sw6/6_q_scheduling
sw6/6_q_shaping
sw6/6_q_tx_ratio
sw6/6_q_credit_hi
sw6/6_q_credit_lo
sw6/6_q_credit_incr
sw6/6_srp
```

Most of the variables that have anything to do with transmit queues will have a “q_” prefix to remind user to change the queue index first to return the right value associated with the queue.

25 Queue Management

25.1 Register Description

Port QM Control Register (0xNA00 – 0xNA03)

Bit	Default	R/W	Description
1-0	0	RW	**Drop Mode**

The Drop Mode is 0 for no drop, 1 for dropping priority 0, 2 for dropping priorities 0 and 1, and 3 for dropping priorities 0, 1, and 2.

Port QM Queue Index Register (0xNA08 – 0xNA0B)

Bit	Default	R/W	Description
25-24	0x0	RW	**Queue Index**
17-16	0x0	RW	**Burst Size**
10-0	0x040	RW	**Minimum Reserved Block**

Port QM Water Mark Register (0xNA0C – 0xNA0F)

Bit	Default	R/W	Description
26-16	0x200	RW	**High Water Mark**
10-0	0x180	RW	**Low Water Mark**

Port QM TX Block Count Register 0 (0xNA10 – 0xNA13)

Bit	Default	R/W	Description
10-0	0	RO	**Number of Block Used**

Port QM TX Block Count Register 1 (0xNA14 – 0xNA17)

Bit	Default	R/W	Description
26-16	0	RO	**Calculated Available Block Count**
10-0	0	RO	**Available Block Count**

25.2 Sysfs Variables

sw6/6_q_index

```
sw6/6_qm_drop
sw6/6_qm_burst
sw6/6_qm_resv_space
sw6/6_qm_hi
sw6/6_qm_lo
sw6/6_qm_tx_used
sw6/6_qm_tx_avail
sw6/6_qm_tx_calc
```

26 Power Management

26.1 Register Description

Switch Power Management Control Register 0 (0x0201)

Bit	Default	R/W	Description
5	0	RW	**PLL Power Down Disable**
4-3	0	RW	**Power Mode**

The Power Mode is 0 for normal, 1 for energy detection, 2 for soft power down, and 3 for power saving.

Switch PME Control Register 0 (0x0006)

Bit	Default	R/W	Description
1	0	RW	**PME Enable**
0	0	RW	**PME Polarity**

Port PME Status Register (0xN013)

Bit	Default	R/W	Description
2	0	WC	**Magic Packet Detect**
1	0	WC	**Link Up Detect**
0	0	WC	**Energy Detect**

Port PME Control Register (0xN017)

Bit	Default	R/W	Description
2	0	RW	**Magic Packet Detect**
1	0	RW	**Link Up Detect**
0	0	RW	**Energy Detect**

PME control is used to assert the PME pin when the specified condition happens.

26.2 Sysfs Variables

```
sw/pme
sw/pme_polarity
sw6/6_pme_ctrl
sw6/6_pme_status
```

27 IBA

27.1 Register Description

Switch IBA Control Register (0x0104 – 0x0107)

Bit	Default	R/W	Description
31	0	RW	**IBA Enable**
30	0	RW	**IBA Destination Address Match**
29	0	RW	**IBA Reset**
25-22	0x1	RW	**Priority/Queue ID for IBA Response**
21-16	0	RW	**IBA Port**
15-0	0x40FE	RW	**IBA Frame Tag**

This register is used to enable IBA and select which port to become IBA port. IBA strap option enables IBA and selects the last port as IBA port.

28 DLR

28.1 Register Description

Switch DLR Source Port Register (0x0604 – 0x0607)

Bit	Default	R/W	Description
31	0	RW	**DLR Beacon Unicast Send**
1-0	0b00	RW	**Source Port Send Mode**

This controls the way the beacons are sent. There is no reason to change it.

Switch DLR Source IP Address Register (0x0608 – 0x060B)

Bit	Default	R/W	Description
31-0	0	RW	**Source IP Address**

The Source IP address is used in the generated beacons.

Switch DLR Control Register (0x0610)

Bit	Default	R/W	Description
3	0	RW	**Reset Sequence ID**
2	0	RW	**Backup Supervisor Enable**
1	0	RW	**Beacon TX Enable**
0	0	RW	**DLR Assist Enable**

The Reset Sequence ID bit is used to reset the sequence id of the beacon to 0. Note that when beacons are being sent the sequence id will be reset so the sequence id will not be consecutive. The DLR Beacon TX Enable bit should be off when this bit is set and then reset.

The Backup Supervisor Enable can cause the switch to start sending beacons when both beacon timeouts are detected.

The Beacon TX Enable causes the switch to start sending beacons.

The DLR Assist Enable should be on when DLR is being used.

Switch DLR State Register (0x0611)

Bit	Default	R/W	Description
2-1	0b00	RW	**Node State**
0	0	RW	**Ring State**

The Node State can be 0 for IDLE_STATE, 1 for FAULT_STATE, and 2 for NORMAL_STATE. It does not have any effect on the switch.

The Ring State is 1 for RING_NORMAL_STATE and 0 for RING_FAULT_STATE. This is used in the generated beacons. Note the actual value of RING_FAULT_STATE is 2.

Switch DLR Precedence Register (0x0612)

Bit	Default	R/W	Description
7-0	0x0	RW	**Supervisor Precedence**

The Supervisor Precedence value is used in the generated beacons.

Switch DLR Beacon Interval Register (0x0614 – 0x0617)

Bit	Default	R/W	Description
31-0	400	RW	**Beacon Interval in Microseconds**

The Beacon Interval value is used in the generated beacons. It also specifies how fast the generated beacons are sent.

Switch DLR Beacon Timeout Register (0x0618 – 0x061B)

Bit	Default	R/W	Description
21-0	1960	RW	**Beacon Timeout in Microseconds**

The Beacon Timeout value is used in the generated beacons.

Switch DLR Beacon Timeout Window Register (0x061C – 0x061F)

Bit	Default	R/W	Description
21-0	6000	RW	**Beacon Timeout Window in Microseconds**

The window is the delay between the first beacon timeout from one port and the timeout of the other port.

Switch DLR VLAN ID Register (0x0620 – 0x0621)

Bit	Default	R/W	Description

11-0	0	RW	**VLAN ID**

The VLAN ID is used in the generated beacons.

Switch DLR Destination Address Register (0x0622 – 0x0627)

Bit	Default	R/W	Description
47-0	01:21:6C:00:01	RW	**Destination Address**

The Destination Address is used in the generated beacons.

Switch DLR Port Map Register (0x0628 – 0x062B)

Bit	Default	R/W	Description
15-0	0x0	RW	**DLR Port Map**

The Port Map is used to select which ports are used as DLR ports. Typically it will be 3.

Switch DLR Source IP Address Register (0x062C)

Bit	Default	R/W	Description
1-0	3	RW	**Beacon Frame QID**

This selects the transmit queue of the generated beacons.

28.1.1 Sysfs Variables

```
dlrfs/prec
dlrfs/interval
dlrfs/timeout
dlrfs/vid
```

DLR requires ACL for detecting beacon timeout. Please refer the ACL section for how to program the ACL table.

29 HSR

29.1 Register Description

Switch HSR TPID Register (0x030C – 0x030D)

Bit	Default	R/W	Description
15-0	0x892F	RW	**HSR Tag**

This is used to change the HSR tag in case a different value is used in the future.

Switch HSR Port Map Register (0x0640 – 0x0643)

Bit	Default	R/W	Description
15-0	0x0	RW	**HSR Port Map**

Although the default value is 0x3 it still requires a write to this register to enable HSR for the first revision of the chip.

Switch HSR ALU Control Register 0 (0x0644)

Bit	Default	R/W	Description
7	1	RW	**HSR Duplicate Discard**
6	1	RW	**HSR Node Unicast**
2	0	RW	**HSR Multicast Learning Disable**
1-0	0x3	RW	**HSR Hash Table Option**

HSR Node Unicast should be turned off for multicast address to work.

Switch HSR ALU Control Register 1 (0x0645)

Bit	Default	R/W	Description
7	0	RW	**HSR Unicast Learning Disable**
5	0	RW	**HSR Flush Table**
3	1	RW	**Process Source Multicast Address Packets**
2	1	RW	**HSR Aging Enable**

29.1.1 Sysfs Variables

```
sw/hsr
```

Although there is a HSR table the entries age out too fast that there is nothing to view.

30 Debugging

30.1 Register Description

Switch MAC Control Register 1 (0x0334)

Bit	Default	R/W	Description
3	0	RW	**Pass Flow Control Packets**

Port MAC Control Register 1 (0xN401)

Bit	Default	R/W	Description
1	0	RW	**Pass All Frames**

The Pass All Frames bit is used together with Port Mirroring for debug purpose only.

The Pass Flow Control Packets bit can be used to debug flow control.

30.2 Sysfs Variables

```
sw6/6_pass_all
sw/pass_pause
```

31 Other Information

There are some more variables available in the sysfs directory. They are primarily used by the driver to report information about the switch so that outside applications can know what to do. Users should not be concerned about those unless there are problems and users are asked to

report this information.

31.1 Sysfs Variables

```
sw/authen
sw/avb
sw/dev_start
sw/host_port
sw/features
sw/overrides
sw/ports
sw/two_dev
sw/version
```