RALINK TECHNOLOGY, CORP.

RT3052

Switch Application Note

AP Router

Proprietary Notice and Liability Disclaimer

The confidential Information, technology or any Intellectual Property embodied therein, including without limitation, specifications, product features, data, source code, object code, computer programs, drawings, schematics, know-how, notes, models, reports, contracts, schedules and samples, constitute the Proprietary Information of Ralink (hereinafter "Proprietary Information")

All the Proprietary Information is provided "AS IS". No Warranty of any kind, whether express or implied, is given hereunder with regards to any Proprietary Information or the use, performance or function thereof. Ralink hereby disclaims any warranties, including but not limited warranties of non-infringement, merchantability, completeness, accuracy, fitness for any particular purpose, functionality and any warranty related to course of performance or dealing of Proprietary Information. In no event shall Ralink be liable for any special, indirect or consequential damages associated with or arising from use of the Proprietary Information in any way, including any loss of use, data or profits.

Ralink retains all right, title or interest in any Proprietary Information or any Intellectual Property embodied therein. The Proprietary Information shall not in whole or in part be reversed, decompiled or disassembled, nor reproduced or sublicensed or disclosed to any third party without Ralink's prior written consent.

Ralink reserves the right, at its own discretion, to update or revise the Proprietary Information from time to time, of which Ralink is not obligated to inform or send notice. Please check back if you have any question. Information or items marked as "not yet supported" shall not be relied on, nor taken as any warranty or permission of use.

Ralink Technology Corporation (Taiwan)
5F, No.36, Tai-Yuen Street,
ChuPei City
HsinChu Hsien 302, Taiwan, ROC
Tel +886-3-560-0868
Fax +886-3-560-0818

Sales Taiwan: Sales@ralinktech.com.tw
Technical Support Taiwan: FAE@ralinktech.com.tw
http://www.ralinktech.com/

Ralink RT3052 Switch

Content

1. VLAN

1.1. Introduction

This Application Note will introduce how to configure VLAN on the RT3050/RT3052 embedded switch.

5.1.1. VID and VLAN member set

RT3052 support 16 VLANs. It can be configured to identify any 16 out of 4096 possible VIDs. These 16 VIDs could be configured by setting VIDx (X=0~15) registers. To configure the member set ports of a given VLAN, one can set the VLAN_MEMSET_x (x=0~15) register. Each bit of the VLAN_MEMSET_x register is corresponding to the associated port. For example, to configure port #1 and port #3 as the member ports of VLAN 5, one can set VLAN_MEMSET_5 as 8'b00001010.

5.1.2. Tag and Untag

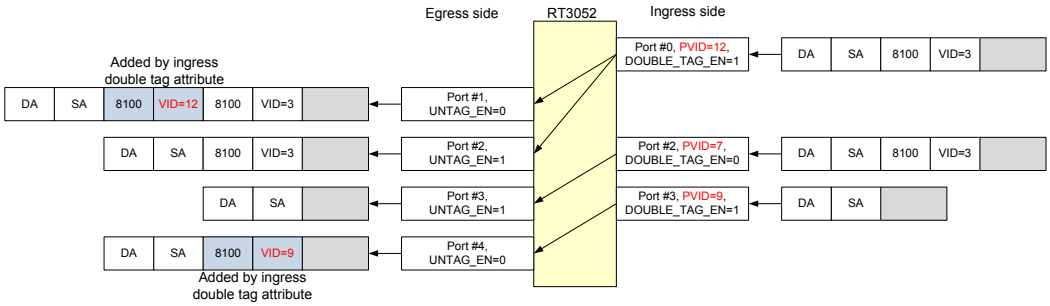
There is a per port register to configure the egress tag and untag setting. If one does not want VLAN tagged frame transmitted from a given port x, he could set UNTAG_EN[x]=1; If one wants VLAN tagged frame transmitted from port y, he could set UNTAG_EN[y]=0. RT3052 support VLAN tag/untag by per egress port basis. It does not support per VLAN per port basis.

5.1.3. Port VID

There is per port Px_PVID register to support PVID. The Px_PVID is assigned to a incoming frame which is untagged or priority tagged (i.e. VID filed =0).

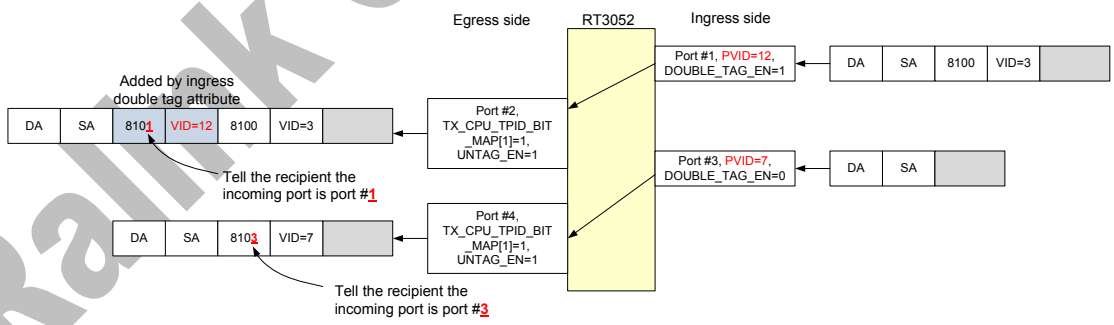
5.1.4. Double Tag

RT3052 supports double VLAN tags by setting a per ingress port register – DOUBLE_TAG_EN[x]. When RT3052 receives a frame from a port with DOUBLE_TAG_EN = 1, it will ignore the VLAN tag filed, if any, and insert the associated PVID in front of the frame after the MAC SA field. Then, it will go on the frame forwarding decision based on this PVID. When this frame is finally be transmitted to an egress port with UNTAG_EN=0, the egress packet will be double VLAN tagged if its incoming format is single VLAN tag; it will be single VLAN tagged if its incoming format is non-VLAN tagged. Please see the following figure for some examples.



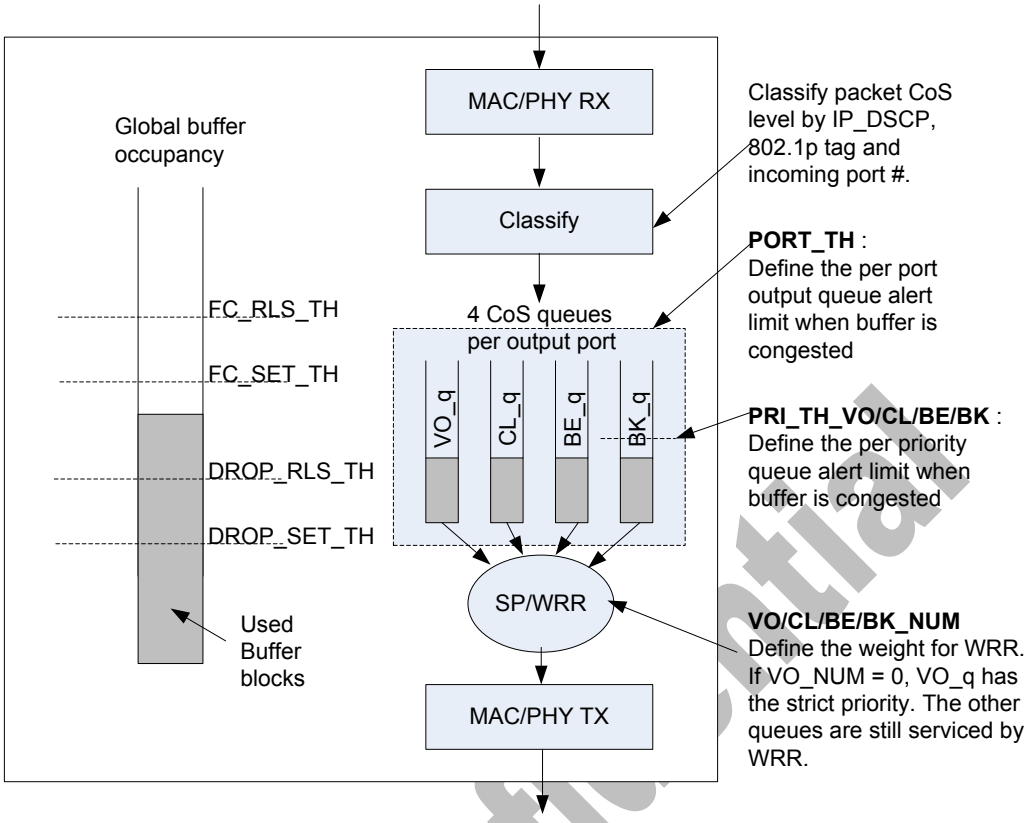
5.1.5. Special Tag

In order to let the recipient (e.g. RT3052 internal CPU or external 3rd party CPU) knows the incoming port number of a received frame, a special tag is supported to rewrite the TPID (0x8100) filed with the incoming port number. The format of this rewritten TPID is : 810x, where x specifies the incoming port number. To enable this feature, one should set CPU_TPID_EN=1 first and specify output ports that need incoming port number to be carried by TPID by setting the associated ports in TX_CPU_TPID_BIT_MAP[6:0]. Please be noted, this special tag feature is a supplement to the existing VLAN tag feature. If the egress frame does not have VLAN tag, there is no way for RT3052 to put incoming port number into the modified TPID field. If the egress frame is double VLAN tagged, the special tag applies to the outer VLNA tag only. Please see the following figure for some examples.



5.1.6. Packet Classification, QoS, Scheduling and Buffer control

RT3052 supports 4 CoS queues per egress port. When a frame is received, it is classified by IP DSCP, 802.1p tag and incoming port priority. The classify sequence is IP DSCP first, then 802.1p tag, and finally the incoming port priority. To enable IP DSCP classification for port x, one has to set EN_TOS[x] to 1; To enable 802.1p tag classification for port x, one has to set EN_VLAN[x] to 1. If both EN_TOS[x] and EN_VLAN[x] are zero or could not be applied (for non-IP or non-VLAN frames), frame will be classified by the PORT_PRIx register.



At the egress side, there is a SP/WRR scheduler for each output port to schedule the frame transmission opportunity. One can assign the weight for each of the VO/CL/BE/BK queues to specify the service ratio. It also support a mixed schedule mode to treat VO queue as the strict priority by assigning its weight (VO_NUM) to zero.

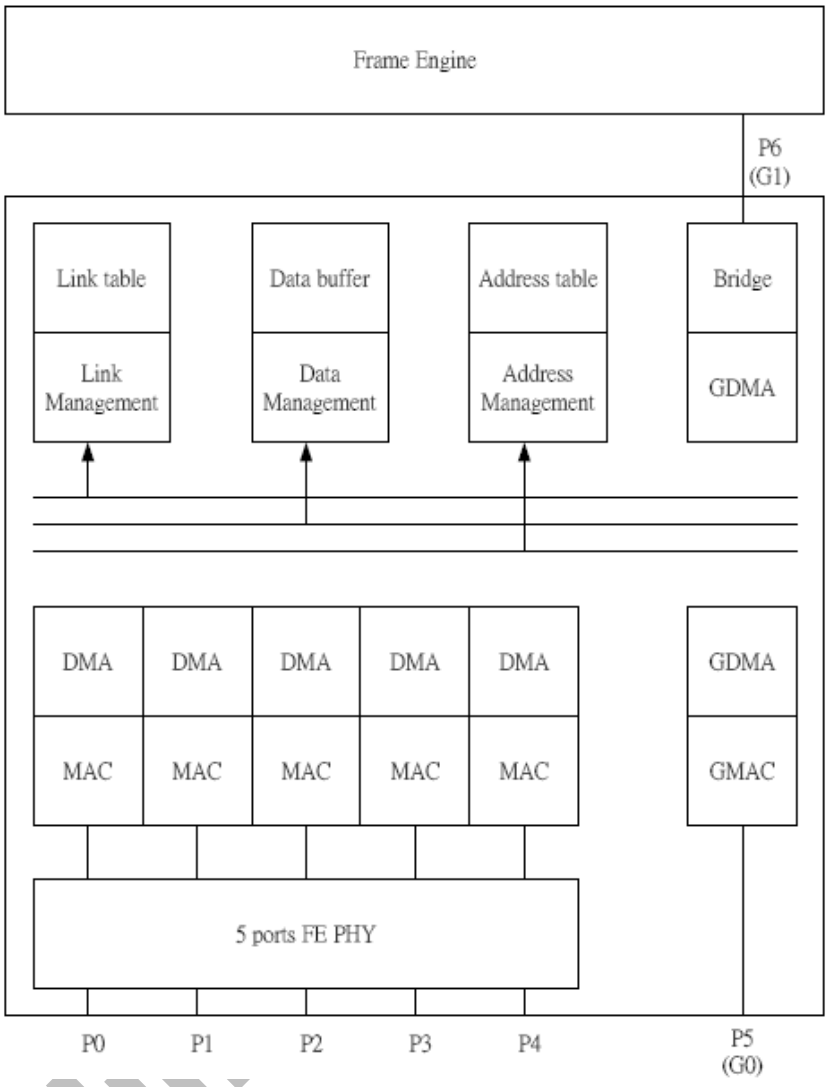
To support QoS-awared flow control, there is a global per CoS queue threshold setting to define the alert threshold when global packet buffer is getting congested. When the global buffer block count is lower than FC_SET_TH, an incoming frame will trigger a pause_ON frame to be transmitted if the PORT_TH of the destination port and PRI_TH_xx (xx = VO or CL or BE or BK) are both reached. By this sophisticated buffer control mechanism, the high priority traffic (e.g. VoIP) will not get dropped or paused if it is put in strict priority VO_q and its source rate is controlled.

The above description for QoS-awared flow control could be worked even well if we turn on FE2SW_WL_FC_EN (frame engine to Ethernet switch WAN-LAN flow control) for one-armed router application. Since there is only a single GE port connecting the frame engine and the embedded Ethernet switch, the traditional 802.3x pause mechanism might block all frames from CPU to the Ethernet switch regardless of

frame's destination (LAN or WAN). In other words, there will be HOL (Head-of-Line blocking in this one-armed router case. To relief the HOL, one could tell the Ethernet switch which ports are LAN ports by specifying the ports into LAN_PMAP register. Together with separated LAN/WAN GDMA in the frame engine, a better QoS-awared flow control could be supported.

1.2. Architecture

Block Diagram



1.3. Description

1.3.1. VLAN Definitions

RT305x embedded switch supports 16 VLAN groups. Each VLAN group has all ports (port 0 to port 6) in its member port by default. The default VLAN member table would be:

VLAN Group Index	Default Group Member
0	Port 0 ~ 6
1	Port 0 ~ 6
…	Port 0 ~ 6
15	Port 0 ~ 6

- VLAN Group Index:

  RT305x embedded switch physically equip with 16 VLANs, index 0 ~ 15.

- VLAN Group Member:

  Port members belong to the same VLAN group.

1.3.2. Switch Behavior

When RT305x embedded switch receives a packet, it will

1. Find its VLAN-ID (VID) according to
   - the Port VID (PVID) value configured in the register, if the receiving packet is an untagged packet
   - the VID value in the packet, if the receiving packet is an 802.1Q-tagged packet
2. Find the VLAN Group Index of the packet according to its VID
3. Forward the packet according to the VLAN member table

1.4. Registers

Register Definitions

The base address of RT305x embedded switch registers is 0x1011.0000.

1.4.1. PVIDC0: PVID Configuration 0 (offset: 0x40)

PVID configuration for port 0 and 1

Bits	Type	Name	Description	Initial value
31:24	RO	-	Reserved	8'd0
23:12	R/W	P1_PVID	Port1 PVID setting	12'd1
11:0	R/W	P0_PVID	Port0 PVID setting	12'd1

1.4.2. PVIDC1: PVID Configuration 1 (offset: 0x44)

PVID configuration for port 2 and 3

Bits	Type	Name	Description	Initial value
31:24	RO	-	Reserved	8'd0
23:12	R/W	P3_PVID	Port3 PVID setting	12'd1
11:0	R/W	P2_PVID	Port2 PVID setting	12'd1

1.4.3. PVIDC2: PVID Configuration 2 (offset: 0x48)

PVID configuration for port 4 and 5

Bits	Type	Name	Description	Initial value
31:24	RO	-	Reserved	8'd0
23:12	R/W	P5_PVID	Port5 PVID setting	12'd1
11:0	R/W	P4_PVID	Port4 PVID setting	12'd1

1.4.4. PVIDC3: PVID Configuration 3 (offset: 0x4C)

PVID configuration for port 6

Bits	Type	Name	Description	Initial value

31:12	RO	-	Reserved	20'd0
11:0	R/W	P6_PVID	Port6 PVID setting	12'd1

1.4.5. VLANI0: VLAN Identifier 0 (offset: 0x50)

VID for VLAN Group Index 0 and 1

Bits	Type	Name	Description	Initial value
31:24	RO	-	Reserved	8'd0
23:12	R/W	VID1	VLAN field Identifier for VLAN 1	12'd2
11:0	R/W	VID0	VLAN field Identifier for VLAN 0	12'd1

1.4.6. VLANI1: VLAN Identifier 1 (offset: 0x54)

VID for VLAN Group Index 2 and 3

Bits	Type	Name	Description	Initial value
31:24	RO	-	Reserved	8'd0
23:12	R/W	VID3	VLAN field Identifier for VLAN 3	12'd4
11:0	R/W	VID2	VLAN field Identifier for VLAN 2	12'd3

1.4.7. VLANI2: VLAN Identifier 2 (offset: 0x58)

VID for VLAN Group Index 4 and 5

Bits	Type	Name	Description	Initial value
31:24	RO	-	Reserved	8'd0
23:12	R/W	VID5	VLAN field Identifier for VLAN 5	12'd6
11:0	R/W	VID4	VLAN field Identifier for VLAN 4	12'd5

1.4.8. VLANI3: VLAN Identifier 3 (offset: 0x5C)

VID for VLAN Group Index 6 and 7

Bits	Type	Name	Description	Initial value
31:24	RO	-	Reserved	8'd0
23:12	R/W	VID7	VLAN field Identifier for VLAN 7	12'd8
11:0	R/W	VID6	VLAN field Identifier for VLAN 6	12'd7

1.4.9. VLANI4: VLAN Identifier 4 (offset: 0x60)

VID for VLAN Group Index 8 and 9

Bits	Type	Name	Description	Initial value
31:24	RO	-	Reserved	8'd0
23:12	R/W	VID9	VLAN field Identifier for VLAN 9	12'd10
11:0	R/W	VID8	VLAN field Identifier for VLAN 8	12'd9

1.4.10. VLANI5: VLAN Identifier 5 (offset: 0x64)

VID for VLAN Group Index 10 and 11

Bits	Type	Name	Description	Initial value
31:24	RO	-	Reserved	8'd0
23:12	R/W	VID11	VLAN field Identifier for VLAN 11	12'd12
11:0	R/W	VID10	VLAN field Identifier for VLAN 10	12'd1

1.4.11. VLANI6: VLAN Identifier 6 (offset: 0x68)

VID for VLAN Group Index 12 and 13

Bits	Type	Name	Description	Initial value

31:24	RO	-	Reserved	8'd0
23:12	R/W	vid13	VLAN field Identifier for VLAN 13	12'd14
11:0	R/W	vid12	VLAN field Identifier for VLAN 12	12'd13

1.4.12. VLANI7: VLAN Identifier 7 (offset: 0x6C)

VID for VLAN Group Index 14 and 15

Bits	Type	Name	Description	Initial value
31:24	RO	-	Reserved	8'd0
23:12	R/W	VID15	Identifier for VLAN 15	12'd16
11:0	R/W	VID14	Identifier for VLAN 14	12'd1

1.4.13. VMSC0: VLAN Member Port Configuration 0 (offset: 0x70)

VLAN member port configuration for VLAN Group Index 0, 1, 2, and 3

Bits	Type	Name	Description	Initial value
30:24	R/W	VLAN_MEMSET_3	VLAN 3 member port	8'hff
22:16	R/W	VLAN_MEMSET _2	VLAN 2 member port	8'hff
15:8	R/W	VLAN_MEMSET _1	VLAN 1 member port	8'hff
7:0	R/W	VLAN_MEMSET _0	VLAN 0 member port	8'hff

1.4.14. VMSC1: VLAN Member Port Configuration 1 (offset: 0x74)

VLAN member port configuration for VLAN Group Index 4, 5, 6, and 7

Bits	Type	Name	Description	Initial value
30:24	R/W	VLAN_MEMSET_7	VLAN 7 member port	8'hff
22:16	R/W	VLAN_MEMSET _6	VLAN 6 member port	8'hff
15:8	R/W	VLAN_MEMSET _5	VLAN 5 member port	8'hff
7:0	R/W	VLAN_MEMSET _4	VLAN 4 member port	8'hff

1.4.15. VMSC2: VLAN Member Port Configuration 2 (offset: 0x78)

VLAN member port configuration for VLAN Group Index 8, 9, 10, and 11

Bits	Type	Name	Description	Initial value
30:24	R/W	VLAN_MEMSET_11	VLAN 11 member port	8'hff
22:16	R/W	VLAN_MEMSET _10	VLAN 10member port	8'hff
15:8	R/W	VLAN_MEMSET _9	VLAN 9 member port	8'hff
7:0	R/W	VLAN_MEMSET _8	VLAN 8 member port	8'hff

1.4.16. VMSC3: VLAN Member Port Configuration 3 (offset: 0x7C)

VLAN member port configuration for VLAN Group Index 12, 13, 14, and 15

Bits	Type	Name	Description	Initial value
30:24	R/W	VLAN_MEMSET_15	VLAN 15 member port	8'hff
22:16	R/W	VLAN_MEMSET _14	VLAN 14 member port	8'hff
15:8	R/W	VLAN_MEMSET _13	VLAN 13 member port	8'hff
7:0	R/W	VLAN_MEMSET _12	VLAN 12 member port	8'hff

1.4.17. PFC1: Priority Flow control –1 (offset: 0x14)

Bits	Type	Name	Description	Initial value
31	R/W	P6_USE_Q1_EN	Port6 only use q1 enable	1'd0
30:24	R/W	EN_TOS[7:0]	Port6 ~ port0 TOS_en. 0: disable	7'd0
23:16	R/O	EN_VLAN	Enable per port VLAN ID and priority check.0: disable.	8'd0

15:14	R/W	-	Reserved	2'b01
13:12	R/W	PORT_PRI6	Port priority By setting this register to force per port' default priority.	2'b01
11:10	R/W	PORT_PRI5	Port priority By setting this register to force per port' default priority.	2'b01
9:8	R/W	PORT_PRI4	Port priority By setting this register to force per port' default priority.	2'b01
7:6	R/W	PORT_PRI3	Port priority By setting this register to force per port' default priority.	2'b01
5:4	R/W	PORT_PRI2	Port priority By setting this register to force per port' default priority.	2'b01
3:2	R/W	PORT_PRI1	Port priority By setting this register to force per port' default priority.	2'b01
1:0	R/W	PORT_PRI0	Port priority By setting this register to force per port' default priority.	2'b01

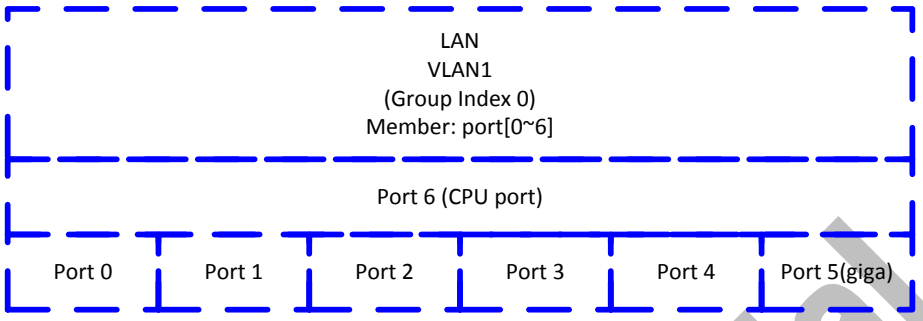
1.4.18. POC2: Port control 2 (offset: 0x98)

Bits	Type	Name	Description	Initial value
31	RO	-	Reserved	1'b0
30	R/W	G1_TXC_CHECK	Check the Port 6 TXC if no txc clock, then disable MII port 1: enable, check TXC	1'b0
29	R/W	G0_TXC_CHECK	Check the port 5 TXC if no txc clock, then disable MII port 1: enable, check TXC	1'b0
28:23	RO	-	Reserved	6'b0
22:16	R/W	DIS_UC_PAUSE	1: switch will not consider pause frame when DA!= 0180c20001 and unicast to CPU 0: switch will consider pause frame when DA!=0180c20001 but unicast to CPU,	7'b0
15	RO	-	Reserved	1'b0
14:8	R/W	ENAGING PORT	Port aging 1: enable aging, 0: disable aging that the MAC address is belong to programmed port(s)	7'h7f
7:0	R/W	UNTAG_EN	Remove VLAN tag field 1: enable VLAN tag field removal.	8'h0

1.5. Configurations

This chapter gives examples of several different configurations for reference.

1.5.1. Dump Switch

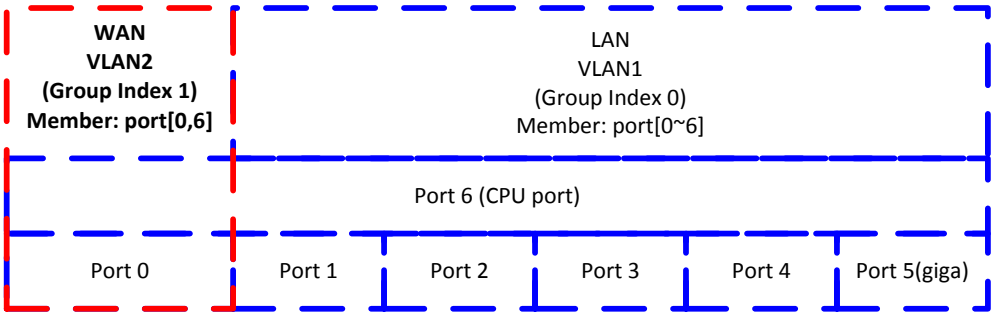


The default value in the registers makes RT305x embedded operates as a dump switch. All ports belong to VLAN Group Index 0 (VID 1).

Register values:

Register offset	Value
0x14	0x5555
0x40	0x1001
0x44	0x1001
0x48	0x1001
0x50	0x2001
0x70	0xffffffff
0x98	0x7f7f

1.5.2. LAN/WAN partition – WAN at port 0

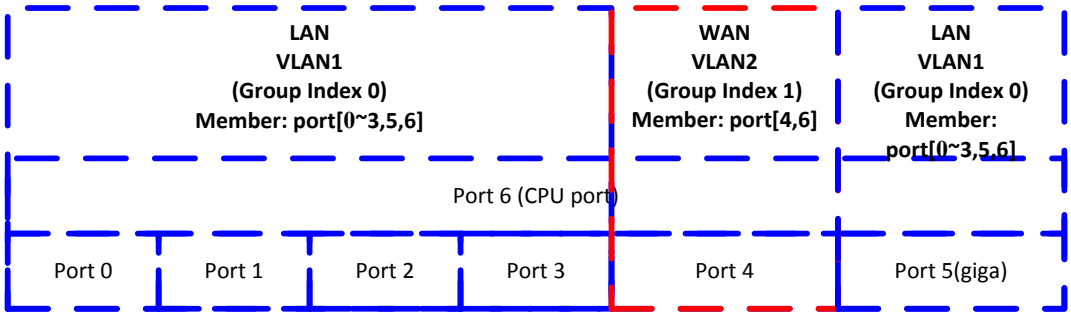


The configuration of WAN on port 0 and LAN on port 1 ~ port 5, and CPU port is a member port of both WAN and LAN. When a packet receiving from port 0 and forwarding to CPU port, RT305x embedded switch will insert a 802.1Q VLAN tag with VID equal to 2 to the packet. Similarly, when a packet receiving from port 1~5 and forwarding to CPU port, the switch will insert a 802.1Q VLAN tag with VID equal to 1 to the packet. On the reversed direction, if CPU needs to forward a packet to WAN or LAN, CPU will insert a 802.1Q VLAN tag to the packet. Then, the switch will forward the packet to port 0 if the VID of the VLAN tag is equal to 2 and to port 1~5 if the VID is equal to 1.

Register values

Register offset	Value
0x14	0x405555
0x40	0x1002
0x44	0x1001
0x48	0x1001
0x50	0x2001
0x70	0xffff417E
0x98	0x7f3f

1.5.3. LAN/WAN partition – WAN at port 4

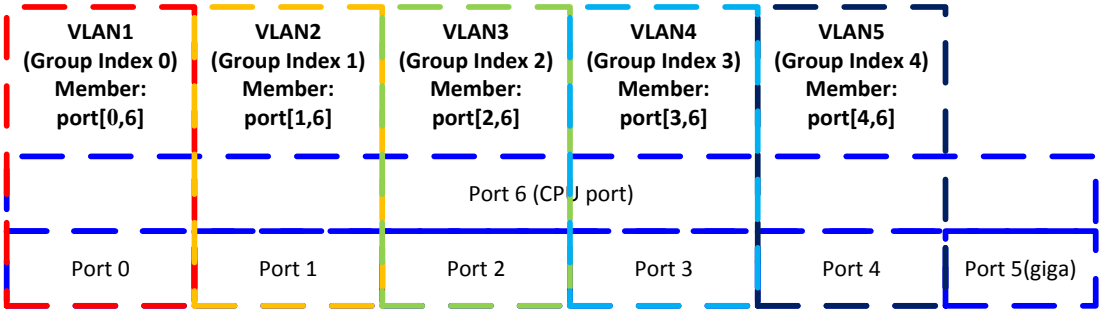


The configuration of WAN on port 4 and LAN on port 0 ~ port 3, and port 5, and CPU port is a member port of both WAN and LAN. When a packet receiving from port 4 and forwarding to CPU port, RT305x embedded switch will insert a 802.1Q VLAN tag with VID equal to 2 to the packet. Similarly, when a packet receiving from port 0~3,5 and forwarding to CPU port, the switch will insert a 802.1Q VLAN tag with VID equal to 1 to the packet. On the reversed direction, if CPU needs to forward a packet to WAN or LAN, CPU will insert a 802.1Q VLAN tag to the packet. Then, the switch will forward the packet to port 4 if the VID of the VLAN tag is equal to 2 and to port 0~3,5 if the VID is equal to 1.

Register values

Register offset	Value
0x14	0x405555
0x40	0x1001
0x44	0x1001
0x48	0x1002
0x50	0x2001
0x70	0xffff506f
0x98	0x7f3f

1.5.4. Per VLAN per port



Register values

Register offset	Value
0x14	0x405555
0x40	0x2001
0x44	0x4003
0x48	0x1005
0x50	0x2001
0x54	0x4003
0x58	0x6005
0x70	0x7e7e7e41
0x74	0xffffff7e
0x98	0x7f3f

1.6. Shell Script

SDK 3.2 provides a shell script to configure VLAN for RT385x embedded switch conveniently. Please reference SDK/source/user/rt2880_app/scripts/config-vlan.sh.

Usage:

- ***config-vlan.sh 2 0***
  It will configure the embedded switch as a dump switch
- ***config-vlan.sh 2 LLLLW***
  It will configure the embedded switch with WAN port at port 4 and LAN port from port 0 to port 3, where WAN port uses VID 2 and LAN ports use VID 1.
- ***config-vlan.sh 2 WLLLL***
  It will configure the embedded switch with WAN port at port 0 and LAN port from port 1 to port 4, where WAN port uses VID 2 and LAN ports use VID 1.
- ***config-vlan.sh 2 12345***
  It will configure the embedded switch with per VLAN per port.

For example, after excuting "config-vlan.sh 2 WLLLL" , we have to create a virtual interface eth2.2 in order to communicate with hosts attached to WAN port.

```
# vconfig add eth2 2
# ifconfig eth2.2 192.168.1.1 netmask 255.255.255.0
# ping 192.168.1.123
```

Similarly, we have to create a virtual interface eth2.1 in order to communicate with hosts attached to LAN ports.

```
# vconfig add eth2 1
# ifconfig eth2.1 10.10.10.254 netmask 255.255.255.0
# ping 10.10.10.123
```

1.7. VLAN FAQ

The following FAQ is for your reference, I think many customers may ask the same question.

1) Priority

Rt305x switch has 4 queues per port, and it supports three method to decide his output queue – VLAN, TOS, Port.

If you enabled TOS, VLAN and port priority at the same time, the priority is VLAN > TOS > Port.

a) VLAN: use priority field in vlan tag

b) TOS: use tos filed in ip header

c) Port: use incoming port

The mapping table is hard-code in switch, and it can't be changed.

802.1p -> Queue	
Priority	Queue
0	1
1	0
2	0
3	1
4	2
5	2
6	3
7	3

TOS -> Queue	
TOS	Queue
0-15	0
16-31	1
32-47	2
48-63	3

Port -> Queue	
Priority	Queue
0	1
1	1
2	2
3	3

PFC1: Priority Flow control –1 (offset: 0x14)				
Bits	**Type**	**Name**	**Description**	**Initial value**
31	R/W	P6_USE_Q1_EN	Port6 only use q1 enable	1'd0
30:24	R/W	EN_TOS[7:0]	Port6 ~ port0 TOS_en. 0: disable	7'd0
23:16	R/O	EN_VLAN	Enable per port VLAN ID and priority check.0: disable.	8'd0
15:14	R/W	-	Reserved	2'b01
13:12	R/W	PORT_PRI6	Port priority By setting this register to force per port's default priority.	2'b01
11:10	R/W	PORT_PRI5	Port priority By setting this register to force per port's default priority.	2'b01
9:8	R/W	PORT_PRI4	Port priority By setting this register to force per port's default priority.	2'b01
7:6	R/W	PORT_PRI3	Port priority By setting this register to force per port's default priority.	2'b01
5:4	R/W	PORT_PRI2	Port priority By setting this register to force per port's default priority.	2'b01
3:2	R/W	PORT_PRI1	Port priority By setting this register to force per port's default priority.	2'b01

1:0	R/W	PORT_PRI0	Port priority By setting this register to force per port's default priority.	2'b01

2) POC: set this bit to notify switch whether it can remove VLAN tag or not

For example: POC=0x7f3f => switch will remove vlan tag except port 6 (cpu port)

POC2: Port control 2 (offset: 0x98)				
Bits	Type	Name	Description	Initial value
31	RO	-	Reserved	1'b0
30	R/W	G1_TXC_CHECK	Check the Port 6 TXC if no txc clock, then disable MII port 1: enable, check TXC	1'b0
29	R/W	G0_TXC_CHECK	Check the port 5 TXC if no txc clock, then disable MII port 1: enable, check TXC	1'b0
28:23	RO	-	Reserved	6'b0
22:16	R/W	DIS_UC_PAUSE	1: switch will not consider pause frame when DA!= 0180c20001 and unicast to CPU 0: switch will consider pause frame when DA!=0180c20001 but unicast to CPU,	7'b0
15	RO	-	Reserved	1'b0
14:8	R/W	ENAGING PORT	Port aging 1: enable aging, 0: disable aging that the MAC address is belong to programmed port(s)	7'h7f
7:0	R/W	UNTAG_EN	Remove VLAN tag field 1: enable VLAN tag field removal.	8'h0

3) double vlan tag

- add pvid in the front of original packet

  vlan | pkt --------(double_tag_en=1) -------> pvid | vlan | pkt
- replace original vlan tag

  vlan | pkt ---------(double_tag_en=0) ------ > pvid | pkt

SGC2: Switch Global Control 2 (offset: 0xE4)				
Bits	**Type**	**Name**	**Description**	**Initial value**
31	R/W	-	Reserved	1'b0
30	R/W	FE2SW_WL_FC_EN	Frame engine to switch WAN/LAN port flow control enable 1:frame engine to switch flow control by WAN/LAN port congestion 0: frame engine to switch flow control by any port congestion	1'b0
29:24	R/W	LAN_PMAP	Lan port bit map 1:Lan port 0:Wan port	6'b0
23	RO	-	Reserved	9'b0
22:16	R/W	TX_CPU_TPID_BIT_MAP	Transmit CPU TPID(810x) port bit map	7'b0
15:12	RO	-	Reserved	4'b0
11	R/W	ARBITER_LAN_EN	memory arbiter only for P0~P4 enable 0: normal 1: memory arbiter only for P0~P4.	1'b0

10	R/W	CPU_TPID_EN	CPU TPID(81xx) enable 0: disable. CPU TPID=8100 1: enable. CPU TPID=810x.	1'b0
9:7	RO	-	Reserved	3'b0
6:0	R/W	DOUBLE_TAG_EN	Insert double tag field 1: enable double tag field	7'b0

4) Switch will add vlan tag for all incoming packets (Lan=1, Wan=2)

So all LAN->WAN or WAN->LAN traffic will be pass to cpu no matter its vlan id.