

Shell 命令的使用

Table of contents 目录

TABLE OF CONTENTS 目录.....	1
1 SHELL 简介.....	2
2 SHELL 的使用.....	2
2.1 帮助命令.....	2
2.2 系统基本信息获取命令.....	7
2.2.1 版本命令.....	7
2.2.2 Show 命令.....	8
2.3 常用检测命令.....	8
2.3.1 端口类.....	8
2.3.2 收发包类.....	10
2.3.3 程序调试类.....	12
2.4 检测芯片的命令.....	12
2.4.1 寄存器的访问.....	12
2.4.2 内存的访问.....	16
2.4.3 中断等其他命令.....	20
2.5 高级配置命令.....	21
2.5.1 Vlan 管理命令.....	21
2.5.2 L2 命令.....	22
2.5.3 L3 命令.....	23
2.5.4 Mirroring 命令.....	25
2.5.5 Filtering.....	25
2.5.6 Spanning Tree Group Management.....	25
2.6 错误检测命令.....	25
2.6.1 Text.....	25
2.6.2 CONFig.....	26
3 REFERENCE DOCUMENTS 参考文献.....	26

1 shell 简介

shell 即 BCM Diagnostic Shell 是一个普通的命令行解释器（CLI），它可以提供对嵌入了 BCM 交换芯片的嵌入式平台的一种精确控制作用。shell 支持所有 Broadcom 交换芯片，并提供了对所有寄存器，内存和最高层 API 的直接访问功能。此外，由于 shell 支持交互和非交互两种模式，所以它可以嵌入到客户的设计中，以提供额外诊断服务。

Shell 命令可被分为六大类：

1. 帮助命令：??，?，help
2. 系统基本信息获取命令：Version，show
3. 常用检测命令：端口类：PORT，COMBO
收发包类：TX, PacketWatcher
程序调试类：Debug，DebugMod
4. 检测芯片的命令：寄存器的访问：Getreg，EditReg 等
内存访问：Dump, Write 等
中断管理：INTR
5. 高级配置命令：VLAN, L2, L3 等
6. 错误检测命令：Test，CONFig，等

2 Shell 的使用

以 DCN DS45XX 项目为例来说明 shell 的使用。

2.1 帮助命令

BCM.0> help //输入 help 命令得到 shell 中所有可用的命令共 201 条

Help: Type help "command" for detailed command usage

Help: Upper case letters signify minimal match

Commands common to all modes:

?	Display list of commands
ASsert	Assert
BackGround	Execute a command in the background.
BCM	Set shell mode to BCM.
BCMx	Set shell mode to BCMx.
break	place to hang a breakpoint
CASE	Execute command based on string match
CD	Change current working directory
cint	Enter the C interpreter
CONFig	Configure Management interface
CONSOLE	Control console options
CoPy	Copy a file
CPUDB	Update the CPU database
CTEcho	Send an echo request using CPUTRANS

	CTInstall	Set up transport pointers in CPU transports
	CTSetup	Modify the CPUTRANS setup
	DATE	Set or display current date
	DBDump	Dump the current StackTask CPUDB
	DBParse	Parse a line of CPUDB dumped code
	DeBug	Enable/Disable debug output
	DeBugMod	Enable/Disable debug output per module
	DELAY	Put CLI task in a busy-wait loop for some amount of
time		
	DEVice	Device add/remove
	DISPatch	BCM Dispatch control.
	Echo	Echo command line
	EDline	Edit file using ancient line editor
	EXIT	Exit the current shell (and possibly reset)
	EXPR	Evaluate infix expression
	FLASHINIT	Initialize on board flash as a file system
	FLASHSYNC	Sync up on board flash with file system
	FOR	Execute a series of commands in a loop
	Help	Print this list OR usage for a specific command
	HISTory	List command history
	IF	Conditionally execute commands
	JOBS	List current background jobs
	KILL	Terminate a background job
	LOCal	Create/Delete a variable in the local scope
	LOG	Enable/Disable logging and set log file
	LOOP	Execute a series of commands in a loop
	LS	List current directory
	MKDIR	Make a directory
	MODE	Set shell mode
	MORe	Copy a file to the console
	MoVe	Rename a file on a file system
	NOEcho	Ignore command line
	Pause	Pause command processing and wait for input
	PRINTENV	Display current variable list
	RCCache	Save contents of an rc file in memory
	RCLoad	Load commands from a file
	REBOOT	Reboot the processor
	RM	Remove a file from a file system
	RMDIR	Remove a directory
	RPC	Control BCM API RPC daemon.
	SAVE	Write data to a file
	SET	Set various configuration options
	SETENV	Create/Delete a variable in the global scope
	SHELL	Invoke a system dependent shell

SLeep	Suspend the CLI task for specified amount of time
TIME	Time the execution of one or more commands
Version	Print version and build information

Commands for current mode:

ADC	Show MAX127 A/D Conversions
AGE	Set ESW hardware age timer
Attach	Attach SOC device(s)
Auth	Port-based network access control
BaseBoard	Configure baseboard system parameters.
BIST	Run on-chip memory built-in self tests
BPDU	Manage BPDU addresses
BTiMeout	Set BIST operation timeout in microseconds
CABLEdiag	Run Cable Diagnostics
CACHE	Turn on/off software caching of tables
CellDataTest	External packet buffer tuning routine
CellHDR	Dump cell header given CBPHEADER index
CHeCK	Check a sorted memory table
CLEAR	Clear a memory table or counters
CLOCKS	Set core clock frequency.
COLOR	Manage packet color
COMBO	Control combination copper/fiber ports
COS	Manage classes of service
CounTeR	Enable/disable counter collection
CustomSTAT	Enable/disable counter collection
DAC	Set DAC register
DELeTe	Delete entry by key from a sorted table
DETaCh	Detach SOC device(s)
DMA	DMA Facilities Interface
DmaRomTest	Simple test of the SOC DMA ROM API
DMIRror	Manage directed port mirroring
DMux	Configure DMUX on a port
DSCP	Map Diffserv Code Points
DTAG	Double Tagging
Dump	Dump an address space or registers
EditReg	Edit each field of SOC internal register
EGRess	Manage source-based egress enabling
EthernetAV	Set/Display the Ethernet AV characteristics
EXTeRnalTuning	External memory automatic tuning
EXTeRnalTuning2	External memory automatic tuning 2
EXTeRnalTuningSum	External memory automatic tuning (summary)
FieldProcessor	Manage Field Processor
Filter	Create/Destroy CPU filter entries
Getreg	Get register

GPORT	Get a GPORT id
H2HIGIG	Convert hex words to higig info
H2HIGIG2	Convert hex words to higig2 info
HASH	Get or set hardware hash modes
HClkSel	Set I2C HClk (MUX for clock-chip-selects)
HeaderMode	Get or set packet tx header mode
I2C	Inter-Integrated Circuit (I2C) Bus commands
IBDump	Display packets pending in the Ingress Buffer
INIT	Initialize SOC and S/W
Insert	Insert into a sorted table
INTR	Enable, disable, show interrupts
IPFIX	IPFIX
IPG	Set default IPG values for ports
IPMC	Manage IPMC (IP Multicast) addresses
L2	Manage L2 (MAC) addresses
L2MODE	Change ARL handling mode
L3	Manage L3 (IP) addresses
LCDMSG	Print message on Matrix Orbital LCD display (via
I2C)	
LED	Control/Load LED processor
LINKscan	Configure/Display link scanning
LISTmem	List the entry format for a given table
Listreg	List register fields
LOOKup	Look up a table entry
MCAST	Manage multicast table
MemSCAN	Turn on/off software memory error scanning
MemWatch	Turn on/off memory snooping
MIM	Manage XGS4 Mac-in-MAC
MIRror	Manage port mirroring
MmuConFiG	Configure MMU mode
MODify	Modify table entry by field names
ModMap	MODID Remapping
Modreg	Read/modify/write register
MPLS	Manage XGS4 MPLS
MPLSER	Manage XGS3 MPLS
MTiMeout	Set MIIM operation timeout in usec
MultiCast	Manage multicast operation
MUXsel	Set I2C LPT state (MUX for clock-chip-selects)
NVram	Manipulate Nonvolatile memory
OAM	Manage OAM groups and endpoints
PacketWatcher	Monitor ports for packets
PBMP	Convert port bitmap string to hex
PCIE	R/W PCIE core registers
PHY	Set/Display phy characteristics /

POE	Configure PowerOverEthernet controllers.
POESel	Set I2C POE (MUX for poe-chip-selects)
POP	Pop an entry from a FIFO
PORT	Set/Display port characteristics
PortRate	Set/Display port rate metering characteristics
PortSampRate	Set/Display sflow port sampling rate
PortStat	Display port status in table
PPDclk	Show PPD clock delay
PROBE	Probe for available SOC units
PUSH	Push an entry onto a FIFO
PVlan	Port VLAN settings
QDR	Read/write over QDR interface
RATE	Manage packet rate controls
RateBw	Set/Display port bandwidth rate metering
characteristics	
RegCMp	Test a register value
REMove	Delete entry by index from a sorted table
RXCfg	Configure RX settings
RXInit	Call bcm_rx_init
RXMon	Register an RX handler to dump received
packets	
SCHan	Send raw S-Channel message, get response
SEArch	Search a table for a byte pattern
Setreg	Set register
SHOW	Show information on a subsystem
SOC	Print internal Driver control information
SRAM	External DDR2_SRAM test control
STACKMode	Set/get the stack mode
StackPortCb	Manage Stack Port Callbacks
StackPortGet	Get stacking characteristics of a port
StackPortSet	Set stacking characteristics of a port
STG	Manage spanning tree groups
STiMeout	Set S-Channel timeout in microseconds
STKMode	Hardware Stacking Mode Control
StkTask	Stack task control
SwitchControl	General switch control
SYnth	Show synthesizer frequency
SystemSnake	Cycle packets through selected system
TCAM	TCAM control
TEMPerature	Show environmental conditions
TestClear	Clear run statistics for a test
TestList	List loaded tests and status
TestMode	Set global test run modes
TestParameters	Set test Parameters

TestRun	Run a specific or selected tests
TestSelect	Select tests for running
TRUNK	Manage port aggregation
TrunkPool	Trunk pool table configuration
TX	Transmit one or more packets
TXCount	Print current TX statistics
TXSTArt	Transmit one or more packets in background
TXSTOp	Terminate a previous "txstart" command
VLAN	Manage virtual LANs
WARMBOOT	Optionally boot warm
WLAN	Manage XGS4 WLAN
Write	Write entry(s) into a table
XAUI	Run XAUI BERT on specified port pair
XClocks	Configure clocks for PCI, SDRAM, Core clock
XPoe	Communication with PD63000 PowerOverEthernet
MCU.	
XQDump	Display packets pending in the XQ
XQErr	Inject bit errors into packets pending in XQ

2.2 系统基本信息获取命令

2.2.1 版本命令

BCM.0> Version

Broadcom Command Monitor: Copyright (c) 1998-2010 Broadcom Corporation

Release: sdk-xgs-robo-5.9.2 built 20110614 (Tue Jun 14 17:34:14 2011)

From pengzhang@smc-pc:/home/pengzhang/work/sdk/sdk-xgs-robo-5.9.2

Platform: KEYSTONE

OS: Unix (Posix)

ROBO Chips: BCM5324_A0, BCM5324_A1, BCM5396_A0, BCM5389_A0,
 BCM5398_A0, BCM5397_A0, BCM5348_A0, BCM5347_A0,
 BCM5395_A0, BCM53242_A0, BCM53262_A0, BCM53115_A0,
 BCM53118_A0, BCM53280_A0, BCM53280_B0, BCM53101_A0,
 BCM53125_A0, BCM53128_A0

Chips: BCM5690_A0, BCM5670_A0, BCM5673_A0, BCM5674_A0,
 BCM5665_A0, BCM5665_B0, BCM5650_C0, BCM5695_A0,
 BCM5675_A0, BCM56601_A0, BCM56601_B0, BCM56601_C0,
 BCM56602_A0, BCM56602_B0, BCM56602_C0, BCM56504_A0,
 BCM56504_B0, BCM56304_B0, BCM56314_A0, BCM56102_A0,
 BCM56112_A0, BCM56800_A0, BCM56218_A0, BCM56514_A0,
 BCM56624_A0, BCM56624_B0, BCM56680_A0, BCM56680_B0,
 BCM56224_A0, BCM56224_B0, BCM56820_A0, BCM56725_A0,
 BCM53314_A0, BCM53324_A0, BCM56634_A0, BCM56634_B0,
 BCM56524_A0, BCM56524_B0, BCM56685_A0, BCM56685_B0,

BCM56334_A0, BCM56334_B0,
 BCM56840_A0, BCM56840_B0, BCM56142_A0,
 PHYs: BCM5218, BCM5220/21, BCM5226, BCM5228,
 BCM5238, BCM5248, BCM5324/FE, BCM5348/FE,
 BCM53242/FE, BCM53262/FE, BCM53101/FE, BCM53280/FE,
 BCM5400, BCM5401, BCM5402, BCM5404,
 BCM5424/34, BCM5411, BCM5461, BCM5464,
 BCM5466, BCM5478, BCM5488, BCM5482,
 BCM5481, BCM5461, BCM5464, BCM5466,
 BCM5478, BCM5488, BCM54980, BCM54980,
 BCM54980, BCM54980, BCM53314, BCM5398,
 BCM5395, BCM53115, BCM53118, BCM5482/801x,
 BCM54684, BCM54640, BCM54682E, BCM54684E,
 BCM54685, BCM54616, BCM5421S, BCM54680,
 BCM53324, BCM53125, BCM53128, BCM54880,
 BCM54881, BCM54810, BCM54640E, BCM54880E,
 BCM54680E, BCM52681E, BCM8703, BCM8704,
 BCM8705/24/25, BCM8706/8726, BCM8727, BCM8728/8747

2.2.2 Show 命令

SHOW unit: 显示系统的芯片信息

SHOW features : 从驱动的角度显示当前芯片的特点

SHOW params: 从驱动的角度显示当前芯片的配置信息

```
BCM.0> show params
driver BCM56504_A0 (firebolt)
  regsfile          Id: //depot/firebolt_fe/regsfile/bcm56120_a0#253
  pci identifier     vendor 0x14e4 device 0xb504 rev 0x01
  classes of service 8
  maximums          block 16 ports 57 mem_bytes 85
  blk 0             bsafe0 schan 11 cmic 11
  blk 1             cmic0 schan 15 cmic 15
  . . . . .
  blk 4             gport0 schan 0 cmic 0
  blk 5             gport1 schan 1 cmic 1
  . . . . .
  blk 9             xport0 schan 2 cmic 2
  blk 10            xport1 schan 3 cmic 3
  blk 11            xport2 schan 4 cmic 4
  blk 12            xport3 schan 5 cmic 5
  port 0            ge blk 4 gport0.0
  port 1            ge blk 4 gport0.1
  port 2            ge blk 4 gport0.2
  port 3            ge blk 4 gport0.3
  port 4            ge blk 4 gport0.4
```

Some blocks are instantiated only once

Other blocks have several copies

A block can have several ports in it

2.3 常用检测命令

2.3.1 端口类

1. PortStat: 该命令可显示所有端口的状态信息

BCM.0> ps											
port	Ena	link	speed/ duplex	link scan	auto neg?	STP state	pause	disacd	lrm	ops	inter face
ge0	En	up	1G FD	SW	Yes	Forward	TX	None	FA	SGMII	
ge1	En	up	100M FD	SW	Yes	Forward	TX	None	FA	SGMII	
ge2	En	down	- HD	SW	Yes	Forward	TX RX	None	FA	SGMII	
ge3	En	down	- HD	SW	Yes	Forward	TX RX	None	FA	SGMII	
ge4	En	down	- HD	SW	Yes	Forward	TX RX	None	FA	SGMII	
ge9	En	down	- HD	SW	Yes	Forward	TX RX	None	FA	SGMII	
ge10	En	down	- HD	SW	Yes	Forward	TX RX	None	FA	SGMII	
ge11	En	down	- HD	SW	Yes	Forward	TX RX	None	FA	SGMII	
hg0	En	up	10G FD	SW	No	Forward		None	FA	XGMII	
BCM.0> ps ?											
Usage (PortStat): Display info about port status in table format.											
Link scan modes:											
SW = software											
HW = hardware											
Learn operations (source lookup failure control):											
F = SLF packets are forwarded											
C = SLF packets are sent to the CPU											
A = SLF packets are learned in L2 table											
D = SLF packets are discarded.											
Pause:											
TX = Switch will transmit pause packets											
RX = Switch will obey pause packets											

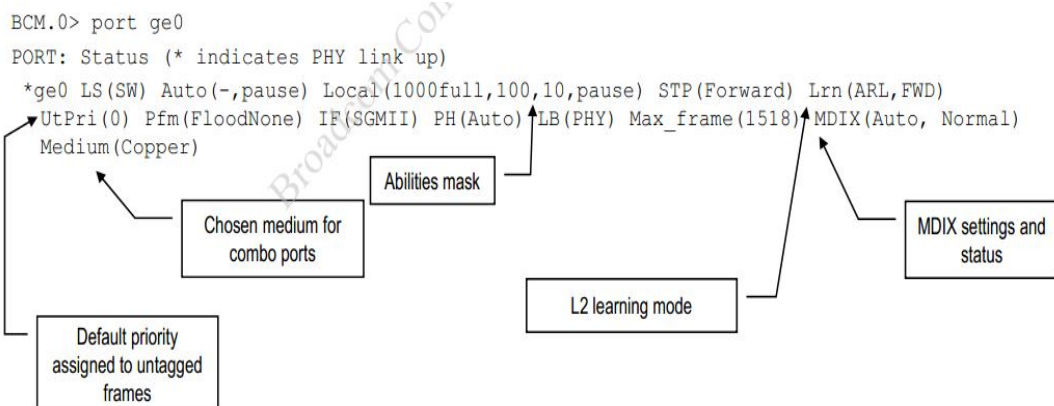
2. Port

Port 的具体使用命令如表 1-1 所示。

表 1-1port 命令使用

Port ge	获得 GE 端口的状态信息
Port ge1 an=off	关掉 GE1 口的自动协商功能
Port ge2-ge5 sp=100	设置端口 2-5 的最大速度到 100Mbps
Port ge10 an=off sp=10 fd=false	是端口 10 速度到 10Mbps
Port ge10 MIDX=Xover	接入端口控制
Help port	列举 port 的其他功能命令

Port 使用的具体例子如下：



3. COMBO

COMBO 的具体使用命令如表 1-2 所示

表 1-2 COMBO 命令使用

Combo ge22-ge23	显示 combo 口的设置
Combo ge22 filber preferred=1	把光纤设置为端口 10 的主流媒介
Combo ge23 copper enable=0	使端口 11 上的铜线媒介失去作用
Combo ge22 watch=on	媒介转换报告使能
Combo ge23 f autoneg_enable=0	为光纤媒介配置设置

force_speed=1000	
------------------	--

具体使用例子如下：

```
BCM.0> combo ge22
Port ge22:
ge22:   Copper medium
        enable=1 preferred=1 force_speed=1000 force_duplex=1 master=Auto
        autoneg_enable=1 autoneg_advert=1000,100,10,pause(0xc3f)
        MDIX=Auto
ge22:   Fiber medium (active)
        enable=1 preferred=0 force_speed=1000 force_duplex=1 master=None
        autoneg_enable=1 autoneg_advert=1000full,pause(0x349c20)
        MDIX=ForcedNormal
```

2.3.2 收发包类

1. Tx: 一个简单的数据包发送器，使用例子如表 1-3 所示

表 1-3 tx 命令使用

tx 10 pbm=ge0 length=100 sm=0x001122334455 dm=0xfffffffffff	从源地址到目的地址传送 10 个 100B 的数据帧
tx 100 pbm=ge0	传输 100 个该数据帧

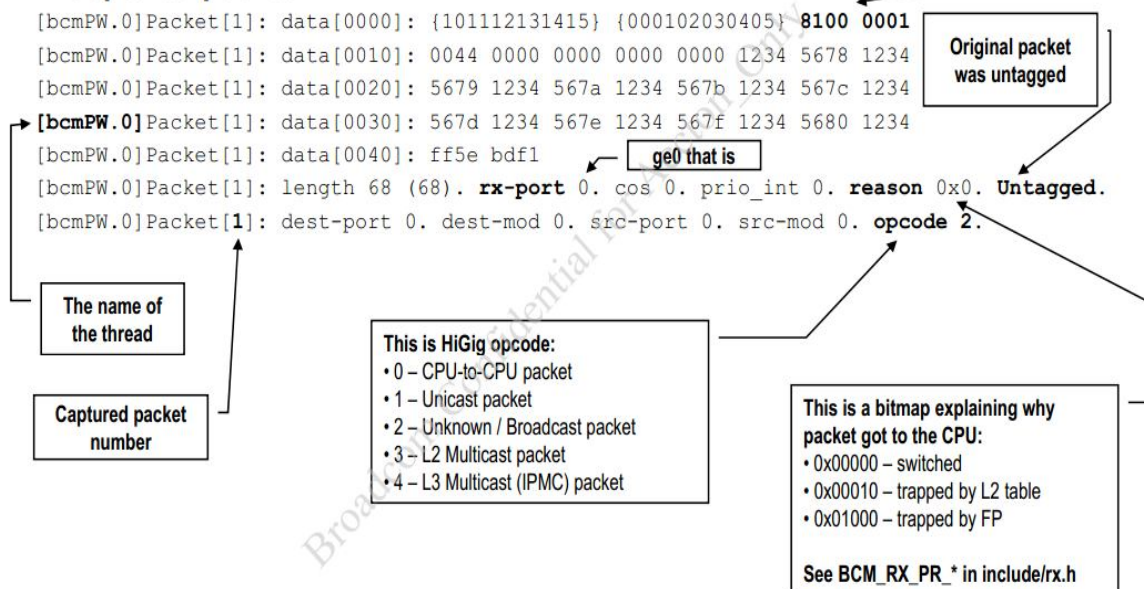
2. PacketWatch: 一个简单的抓包工具，使用例子如表 1-4

表 1-4 PW 命令的使用

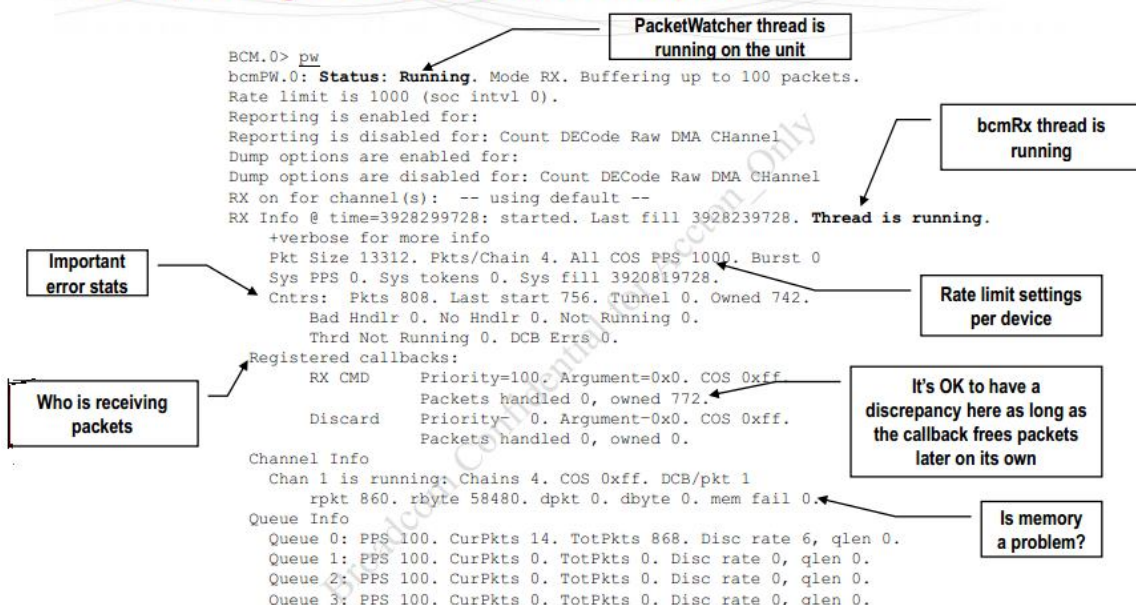
tx 100 pbm=ge0	记忆最后的 20 个数据包
pw report -raw	Do not dump raw packets on the console
pw start	Start the daemon
pw count	收包计数
pw stop	停止计数
pw report +decode	Allow dumping the decoded packets
pw dump	Dump the logged packets

具体使用例子如下：

• Captured packet



Interpreting Packet Watcher Output



3. LINKscan: 如实反映当前端口的链路状态，使用例子如表 1-5 所示

表 1-5 LINKscan 的使用

link	Displays current status of linkscan
link ?	Lists the built-in help for linkscan
link force=ge1	Forces linkscan to update ports status instantly

2.3.3 程序调试类

- 1. DeBug: 控制 debug 输出
 - 2. DeBugMod: 按照模块来控制 debug 设置
- 它们的使用例子如表 1-6 所示

表 1-6 调试命令的使用

db	Shows the debugging settings
db +SOCMEM	Enable debugging output for table accesses
db -intr	Disable interrupt debugging
db tx	Toggle packet transmit debug
dbm soc +verbose +10g	Add verbose messages for HiGig/10GE driver
dbm bcm +verbose +vlan	Add verbose messages for BCM VLAN operations
dbm bcmx	Show debug levels for the BCMX module

2.4 检测芯片的命令

2.4.1 寄存器的访问

系统的寄存器分类如表 1-7 所示

表 1-7 系统的寄存器分类

REGISTER TYPE	DESCRIPTION
PCIC	PCI configuration space registers
PCIM	PCI memory-mapped registers (CMIC registers)
SOC	Registers and memories accessible via S-channel operations
PHY	PHY registers (MIIM addresses)

读 PHY 寄存器实例:

BCM.0> phy 1

Port ge0 (PHY addr 0x01): BCM54682E (54682 Gigabit PHY Driver)

0x00: 0x1140	0x01: 0x79c9	0x02: 0x0362	0x03: 0x5d24
0x04: 0x01e1	0x05: 0x0000	0x06: 0x0064	0x07: 0x2001
0x08: 0x0000	0x09: 0x0600	0x0a: 0x0000	0x0b: 0x0000
0x0c: 0x0000	0x0d: 0x4007	0x0e: 0x0000	0x0f: 0x3000
0x10: 0x0001	0x11: 0x0000	0x12: 0x0000	0x13: 0x0000
0x14: 0x0000	0x15: 0x0000	0x16: 0x0000	0x17: 0x0000
0x18: 0x7067	0x19: 0x1000	0x1a: 0x0000	0x1b: 0xffff
0x1c: 0x2418	0x1d: 0x0000	0x1e: 0x0000	0x1f: 0x0000

BCM.0> phy 1-5

Port ge0 (PHY addr 0x01): BCM54682E (54682 Gigabit PHY Driver)

0x00: 0x1140	0x01: 0x79c9	0x02: 0x0362	0x03: 0x5d24
--------------	--------------	--------------	--------------

0x04: 0x01e1	0x05: 0x0000	0x06: 0x0064	0x07: 0x2001
0x08: 0x0000	0x09: 0x0600	0x0a: 0x0000	0x0b: 0x0000
0x0c: 0x0000	0x0d: 0x4007	0x0e: 0x0000	0x0f: 0x3000
0x10: 0x0001	0x11: 0x2000	0x12: 0x0000	0x13: 0x0000
0x14: 0x0000	0x15: 0x0000	0x16: 0x0000	0x17: 0x0000
0x18: 0x7067	0x19: 0x1000	0x1a: 0x0000	0x1b: 0xffff
0x1c: 0x2418	0x1d: 0x0000	0x1e: 0x0000	0x1f: 0x0000
Port ge1 (PHY addr 0x02): BCM54682E (54682 Gigabit PHY Driver)			
0x00: 0x1140	0x01: 0x79c9	0x02: 0x0362	0x03: 0x5d24
0x04: 0x01e1	0x05: 0x0000	0x06: 0x0064	0x07: 0x2001
0x08: 0x0000	0x09: 0x0600	0x0a: 0x0000	0x0b: 0x0000
0x0c: 0x0000	0x0d: 0x4007	0x0e: 0x0000	0x0f: 0x3000
0x10: 0x0001	0x11: 0x0001	0x12: 0x0000	0x13: 0x0000
0x14: 0x0000	0x15: 0x0000	0x16: 0x0000	0x17: 0x0000
0x18: 0x7067	0x19: 0x1000	0x1a: 0x0000	0x1b: 0xffff
0x1c: 0x2418	0x1d: 0x0000	0x1e: 0x0000	0x1f: 0x0000
Port ge2 (PHY addr 0x03): BCM54682E (54682 Gigabit PHY Driver)			
0x00: 0x1140	0x01: 0x79c9	0x02: 0x0362	0x03: 0x5d24
0x04: 0x01e1	0x05: 0x0000	0x06: 0x0064	0x07: 0x2001
0x08: 0x0000	0x09: 0x0600	0x0a: 0x0000	0x0b: 0x0000
0x0c: 0x0000	0x0d: 0x4007	0x0e: 0x0000	0x0f: 0x3000
0x10: 0x0001	0x11: 0x0000	0x12: 0x0000	0x13: 0x0000
0x14: 0x0000	0x15: 0x0000	0x16: 0x0000	0x17: 0x0000
0x18: 0x7067	0x19: 0x1000	0x1a: 0x0000	0x1b: 0xffff
0x1c: 0x2418	0x1d: 0x0000	0x1e: 0x0000	0x1f: 0x0000
Port ge3 (PHY addr 0x04): BCM54682E (54682 Gigabit PHY Driver)			
0x00: 0x1140	0x01: 0x79c9	0x02: 0x0362	0x03: 0x5d24
0x04: 0x01e1	0x05: 0x0000	0x06: 0x0064	0x07: 0x2001
0x08: 0x0000	0x09: 0x0600	0x0a: 0x0000	0x0b: 0x0000
0x0c: 0x0000	0x0d: 0x4007	0x0e: 0x0000	0x0f: 0x3000
0x10: 0x0001	0x11: 0x2001	0x12: 0x0000	0x13: 0x0000
0x14: 0x0000	0x15: 0x0000	0x16: 0x0000	0x17: 0x0000
0x18: 0x7067	0x19: 0x1000	0x1a: 0x0000	0x1b: 0xffff
0x1c: 0x2418	0x1d: 0x0000	0x1e: 0x0000	0x1f: 0x0000
Port ge4 (PHY addr 0x05): BCM54682E (54682 Gigabit PHY Driver)			
0x00: 0x1140	0x01: 0x79c9	0x02: 0x0362	0x03: 0x5d24
0x04: 0x01e1	0x05: 0x0000	0x06: 0x0064	0x07: 0x2001
0x08: 0x0000	0x09: 0x0600	0x0a: 0x0000	0x0b: 0x0000
0x0c: 0x0000	0x0d: 0x4007	0x0e: 0x0000	0x0f: 0x3000
0x10: 0x0001	0x11: 0x2000	0x12: 0x0000	0x13: 0x0000
0x14: 0x0000	0x15: 0x0000	0x16: 0x0000	0x17: 0x0000
0x18: 0x7067	0x19: 0x1000	0x1a: 0x0000	0x1b: 0xffff
0x1c: 0x2418	0x1d: 0x0000	0x1e: 0x0000	0x1f: 0x0000

关于 PHY 寄存器的其他命令如表 1-8 所示

表 1-8 PHY 命令

phy all	Dumps all registers on all PHYs attached to the chip. This command is equivalent to command <code>dump phy</code> .
phy ge1	Dumps all registers on the PHY attached to port ge1. This command is equivalent to command <code>dump phy ge1</code> .
phy ge 0x2	Prints the first half of the PHY ID register on all GE PHYs
phy ge0 0x0 0x8000	Resets the PHY on ge0 (by writing 1 into the Reset bit in register 0)

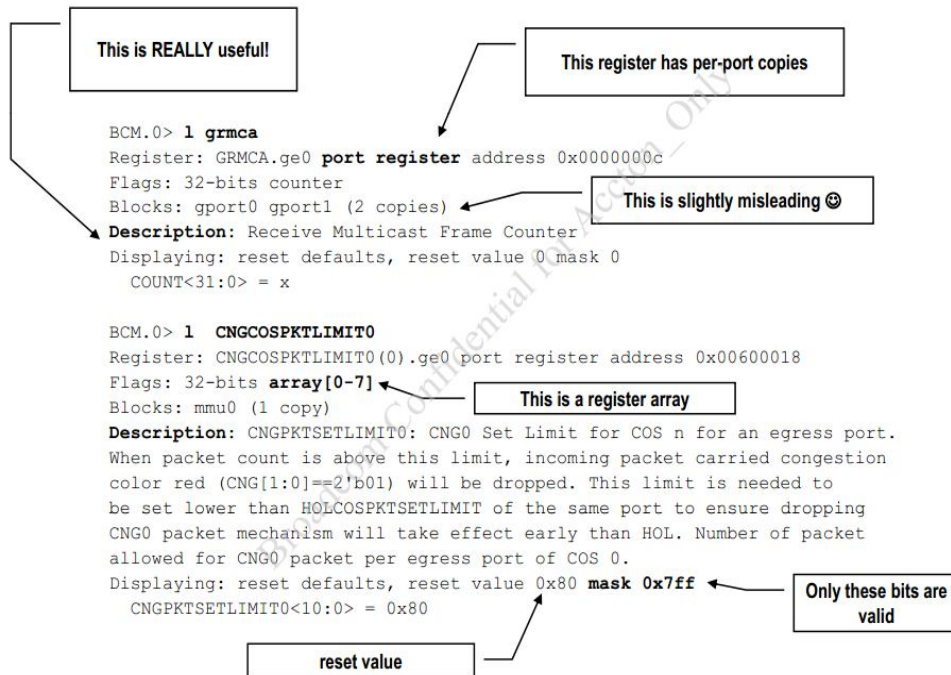
关于 PCI 总线的命令如表 1-9 所示

表 1-9 PCI 命令

show pci	Displays the summary of all devices discovered on the PCI bus
dump pcic	Dumps the PCI configuration registers for the current chip
g pcic 0x10	Dumps a specific PCI configuration register for the current chip
s pcic 0x10 0x81000004	Sets a specified value to a PCI configuration register

1. Listreg : 此命令将显示寄存器的所有详细信息，使用例子如表 1-10 所示
表 1-10 Listreg 命令的使用

EXAMPLE	DESCRIPTION
<code>l ipg</code>	Get a list of matching register names
<code>l config</code>	Get all the info about a register
<code>l -t</code>	Get all the chip register names organized by block
<code>l -c</code>	Get all the counter registers
<code>l 0x00500100</code>	Get all the info about a (SOC) register with the specified address
<code>l config 0x123</code>	Interpret value 0x00000123 as the contents of CONFIG register



2. Getreg : 这个命令将会获取某个寄存器的简要信息，使用例子如表 1-11 所示
 表 1-11 Getreg 命令的使用

EXAMPLE	DESCRIPTION
g CMIC_IRQ_STAT	Prints CMIC_IRQ_STAT in hex, then field-by-field
g CONFIG	Prints all the copies of CONFIG in all blocks
g raw CONFIG	Prints all the copies of CONFIG in all blocks in short form
g chg CONFIG.ge1	It shows only fields/registers that have changed from defaults
g pcic 0x10	Gets the value of PCIC-type register with address 0x10

读 pcic、pcim 寄存器某一位上的值：

BCM.0> get pcic 0x0
 pci_cfg[0x0] = 0xb14314e4
 BCM.0> get pcim 0x4
 cpu[0x4] = 0x40

3. Setreg: 设置一个寄存器的值，使用例子如表 1-12 所示

表 1-12 Setreg 命令的使用

s mac_ctrl 0x1	Sets the value of MAC_CTRL register to 0x1
s mac_ctrl rxen=1	Set RXEN field in MAC_CTRL register to 1 (BUT all other fields will be set to 0!)
s mac_ctrl txen=1,rxen=1	A more reasonable setting
s gmacc2 ipgt=0xe	Sets the value of IPG fields of all GE ports to 0xe
s gmacc2.ge0 ipgt=0xe	Sets the value of IPG fields of ge0 to 0xe

4. Modreg: 设置寄存器值的命令，具体例子如表 1-13 所示

表 1-13 Modreg 的使用

m mac_ctrl rxen=0	Disable Rx, but Tx will be unaffected
m cpu_control_1 uvlan_tocpu=1	Trap packets with unknown VLAN to CPU (without affecting other settings)
m cos_sel.ge0 cos6=2	On ge0, map 802.1p priority 6 onto COS queue 2
m cpu_control_2 cpu_protocol_priority=7	Use higher priority for trapped BPDUs

Setreg 与 Modreg 的区别在于：用 Setreg 命令设置寄存器的值时，寄存器的其他为被自动清零；而用 Modreg 设置寄存器值时，寄存器的其他位保持不变，因此 Modreg 为更安全的寄存器设置命令：

```
BCM.0> g gmaccl.ge0
GMACCl.ge0[0x1]=1: <FULLD=1, PADEN=0, VLPAD=0, ADPAD=0, CRCEN=0, FLCHK=0,
HUGEN=0, JUMBO=0, PUREP=0, FCTX=0, FCRX=0, PARF=0, LONGP=0, MIFG=0,
GLVR=0, DCRC12=0, CLRCNT=0, RXEN0=0, TXEN0=0, AUTOZ=0>
BCM.0> m gmaccl.ge0 jumbo=1
BCM.0> g gmaccl.ge0
GMACCl.ge0[0x1]=0x81: <FULLD=1, PADEN=0, VLPAD=0, ADPAD=0, CRCEN=0,
FLCHK=0, HUGEN=0, JUMBO=1, PUREP=0, FCTX=0, FCRX=0, PARF=0, LONGP=0,
MIFG=0, GLVR=0, DCRC12=0, CLRCNT=0, RXEN0=0, TXEN0=0, AUTOZ=0>
BCM.0> s gmaccl.ge0 fctx=1
BCM.0> g gmaccl.ge0
GMACCl.ge0[0x1]=0x201: <FULLD=1, PADEN=0, VLPAD=0, ADPAD=0, CRCEN=0,
FLCHK=0, HUGEN=0, JUMBO=0, PUREP=0, FCTX=1, FCRX=0, PARF=0, LONGP=0,
MIFG=0, GLVR=0, DCRC12=0, CLRCNT=0, RXEN0=0, TXEN0=0, AUTOZ=0>
```

2.4.2 内存的访问

1. LISTmem: 该命令的作用是显示关于本表的所有细节，具体例子如表 1-14 所示

表 1-14 LISTmem 的使用

list	Lists all the chip memories
list l3	Lists all the chip memories with the names containing "L3" as a substring
list l3_entry	Lists all information about the L3_ENTRY table and its fields
list l3_entry 0x01 0x02 0x03 0x04	Interprets the value 0x000000001000000020000000300000004 as an L3_ENTRY
list gimask.ge3	Lists the gimask table located on block ge3

2. Dump: 该命令的作用是打印所指定表的所有表项及其相应的值。

打印 PCI 的配置情况：

```
BCM.0> DUMP PCIC
```

```
0000: b14314e4 DeviceID=b143 VendorID=14e4
0004: 00100006 Status=0010 Command=0006
0008: 02000002 ClassCode=020000 RevisionID=02
000c: 00000008 BIST=00 HeaderType=00 LatencyTimer=00
CacheLineSize=08
0010: 08000004 BaseAddress0=08000004
```



```

0014: 00000000 BaseAddress1=00000000
0018: 00000000 BaseAddress2=00000000
001c: 00000000 BaseAddress3=00000000
0020: 00000000 BaseAddress4=00000000
0024: 00000000 BaseAddress5=00000000
0028: 00000000 CardbusCISPointer=00000000
002c: b14314e4 SubsystemID=b143 SubsystemVendorID=14e4
0030: 00000000 ExpansionROMBaseAddress=00000000
0034: 00000048 Reserved=000000 CapabilitiesPointer=48
0038: 00000000 Reserved=00000000
003c: 00000105 Max_Lat=00 Min_Gnt=00 InterruptPin=01 InterruptLine=05
0040: 00000000 Reserved=00 RetryTimeoutValue=00 TRDYTimeoutValue=00
0044: dead0000 PLLConf=0
0048: c8035001 -
0048: c8035001 CapabilityID=01 CapabilitiesPointer=50 PWR-MGMT
004c: 00002008 -
0050: 00005803 CapabilityID=03 CapabilitiesPointer=58 VPD
0054: 00000000 -
0058: 0086a005 CapabilityID=05 CapabilitiesPointer=a0 MSI
005c: 00000000 -
0060: 00000000 -
0064: 00000000 -
0068: 00000000 -
006c: 00000000 -
00a0: 0000ac11 CapabilityID=11 CapabilitiesPointer=ac MSI-X
00a4: 00000000 -
00a8: 00000000 -
00ac: 00020010 CapabilityID=10 CapabilitiesPointer=00 PCIE
00b0: 00008002 -
00b4: 00101c20 -

```

其他应用例子如表 1-15 所示：

表 1-15 Dump 的使用

d vlan	Prints all VALID entries from the VLAN table
d vlan 100	Print an entry corresponding to a VLAN 100 (if it is valid)
d all vlan 110	Print an entry corresponding to VLAN 100 (regardless)
d vlan 100 3	Prints valid entries from the VLAN 100 – 102 range
d chg l2_table	Prints all the entries in GIRULE.ge1 we've messed up

```

BCM.0> vlan show
vlan 1 ports cpu,ge,hg {0x00000000000003fff}, untagged ge {0x0000000000000fff}
vlan 100 ports ge0-ge3 {0x000000000000000f}, untagged none {0x0000000000000000}
vlan 101 ports ge4-ge6 {0x0000000000000070}, untagged none {0x0000000000000000}
vlan 102 ports ge7-ge9 {0x0000000000000380}, untagged none {0x0000000000000000}
vlan 103 ports ge10-ge11 {0x0000000000000c00}, untagged none {0x0000000000000000}
vlan 104 ports ge3-ge8 {0x00000000000001f8}, untagged none {0x0000000000000000}
vlan 105 ports ge {0x0000000000000fff}, untagged none {0x0000000000000000}
BCM.0> d vlan
VLAN.ar10[1]: <VALID=1,STG=1,PORT_BITMAP=0x3fff,UT_BITMAP=0xffff>
VLAN.ar10[100]: <VALID=1,STG=1,PORT_BITMAP=0xf,UT_BITMAP=0>
VLAN.ar10[101]: <VALID=1,STG=1,PORT_BITMAP=0x70,UT_BITMAP=0>
VLAN.ar10[102]: <VALID=1,STG=1,PORT_BITMAP=0x380,UT_BITMAP=0>
VLAN.ar10[103]: <VALID=1,STG=1,PORT_BITMAP=0xc00,UT_BITMAP=0>
VLAN.ar10[104]: <VALID=1,STG=1,PORT_BITMAP=0x1f8,UT_BITMAP=0>
VLAN.ar10[105]: <VALID=1,STG=1,PORT_BITMAP=0xffff,UT_BITMAP=0>
BCM.0> d vlan 102
VLAN.ar10[102]: <VALID=1,STG=1,PORT_BITMAP=0x380,UT_BITMAP=0>
BCM.0> d vlan 110
BCM.0> d all vlan 110
VLAN.ar10[110]: <VALID=0,STG=0,PORT_BITMAP=0,UT_BITMAP=0>
BCM.0> d vlan 100 3
VLAN.ar10[100]: <VALID=1,STG=1,PORT_BITMAP=0xf,UT_BITMAP=0>
VLAN.ar10[101]: <VALID=1,STG=1,PORT_BITMAP=0x70,UT_BITMAP=0>
VLAN.ar10[102]: <VALID=1,STG=1,PORT_BITMAP=0x380,UT_BITMAP=0>
BCM.0> d chg l2_table
L2_TABLE.ar10[3080]: <MAC_ADDR=1,VLAN_ID=1>

```

3. Write: 该命令的作用是该某个表里写一些值或者是表项，具体例子如表 1-16

表 1-16 Write 的使用

w l2_valid_bits 12 1 16	Make l2_valid_bits [12] a valid entry
w gffpcounters 0 32 0	Sets the values all fffpcounters to 0
w vlan21 port_bitmap=0x3 stg=1 valid=1 w erg_valn 21 port_bitmap=0x3 stg=1 valid=1	Creat vlan2 with ge0 and ge1 as the only members
w valn_stg 11 0x0 0x0 w erg_vlan_stg 11 0x0 0x0	Put all port in stg 1 into disabled state
W fp_port_filed_sel 0 28 0x0 0x0 0x0 0x0 0x0 W tcam_plus_policy 01 valid=3 ipbm_sel=1\ f2=0xc0a80100000000000000000000000000\ f2_mask=0xfffff000000000000000000000000000	Creat a filiter that will trap packets with source IP addr 192.168.1/24 to cpu

4. MODify: 此命令可作为一个更安全的写命令代替，具体例子如表 1-17

表 1-17MODify 的使用

mod vlan 110 1 port_bitmap=0xf	Set the port bitmap for VLAN 110 to ge0-ge3
mod vlan 110 10 port_bitmap=0xf	The same operation for VLANs 110-119 (regardless of the VALID bit, BTW!)

Write 与 MODify 的使用区别如下实例所示:

```

BCM.0> d trunk_bitmap
TRUNK_BITMAP.ar10[0]: <TRUNK_BITMAP=1,RTAG=0>
TRUNK_BITMAP.ar10[1]: <TRUNK_BITMAP=1,RTAG=0>
TRUNK_BITMAP.ar10[2]: <TRUNK_BITMAP=1,RTAG=0>
TRUNK_BITMAP.ar10[3]: <TRUNK_BITMAP=1,RTAG=0>
.....
TRUNK_BITMAP.ar10[29]: <TRUNK_BITMAP=1,RTAG=0>
TRUNK_BITMAP.ar10[30]: <TRUNK_BITMAP=1,RTAG=0>
TRUNK_BITMAP.ar10[31]: <TRUNK_BITMAP=1,RTAG=0>
BCM.0> w trunk_bitmap 0 32 rtag=1
BCM.0> d trunk_bitmap
TRUNK_BITMAP.ar10[0]: <TRUNK_BITMAP=0,RTAG=1>
TRUNK_BITMAP.ar10[1]: <TRUNK_BITMAP=0,RTAG=1>
TRUNK_BITMAP.ar10[2]: <TRUNK_BITMAP=0,RTAG=1>
TRUNK_BITMAP.ar10[3]: <TRUNK_BITMAP=0,RTAG=1>
.....
TRUNK_BITMAP.ar10[29]: <TRUNK_BITMAP=0,RTAG=1>
TRUNK_BITMAP.ar10[30]: <TRUNK_BITMAP=0,RTAG=1>
TRUNK_BITMAP.ar10[31]: <TRUNK_BITMAP=0,RTAG=1>
BCM.0> mod trunk_bitmap 0 32 trunk_bitmap=1
BCM.0> d trunk_bitmap
TRUNK_BITMAP.ar10[0]: <TRUNK_BITMAP=1,RTAG=1>
TRUNK_BITMAP.ar10[1]: <TRUNK_BITMAP=1,RTAG=1>
TRUNK_BITMAP.ar10[2]: <TRUNK_BITMAP=1,RTAG=1>
TRUNK_BITMAP.ar10[3]: <TRUNK_BITMAP=1,RTAG=1>
.....
TRUNK_BITMAP.ar10[29]: <TRUNK_BITMAP=1,RTAG=1>
TRUNK_BITMAP.ar10[30]: <TRUNK_BITMAP=1,RTAG=1>
TRUNK_BITMAP.ar10[31]: <TRUNK_BITMAP=1,RTAG=1>

```

5. Insert: 给哈希表或者存储表插入新的条目，具体例子如表 1-18

表 1-18 Insert 的使用

i girule.ge1 filter=0x66778899aabb,dst_port=0xd,action=0x20	Insert an entry into the sorted GIRULE table
i l2x mac_addr=0x001122334455,vlan_id=0x123, tgid_port=0x2	Insert an entry into L2 table
i L2_ENTRY mac_addr=0x001122334455\ vlan_id=1 cpu=1static_bit=1 valid=1	Insert an entry into L2 table to trap All packets destined to MAC 00:11:22:33:44:55 valn 1 to cpu

6. REMove: 删除表项命令，具体例子如表 1-19

表 1-19 REMove 的使用

rem girule.ge1 0	Removes the first entry from GI RULE table on block ge1. And all the remaining entries will shift up automatically.
rem l2x 536	Removes an entry 536 from an L2 table

7. LOOKup: 寻找指定表项命令，具体例子如表 1-20

表 1-20 LOOKup 的使用

look l2x mac_addr=0x66778899aa44,vlan_id=0x6a	Finds a specific entry in L2 table
--	------------------------------------

8. SEArch: 功能更强大的查找条目命令，它支持十六进制数和字段值的参数查找当使用十六进制字符串参数时要小心单词的边界；查找与字节顺序无关(It

accepts patterns both as hex data and as field values. (When using hex string patterns do beware of word boundaries); A search will be successful regardless of byte ordering.) 具体例子如表 1-21

表 1-21 SEArch 的使用

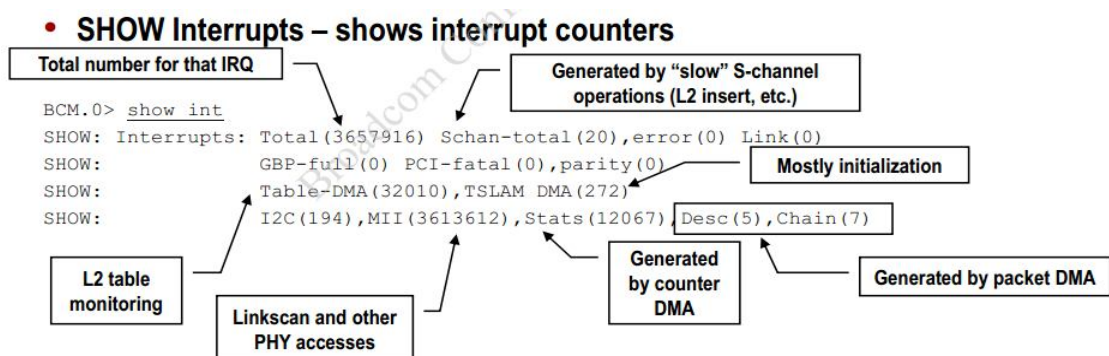
sea l2x all 0x22334455	Both will match entries with MAC_ADDR=0x001122334455
sea l2x all 0x55443322	
sea l2x vlan_id=100	Finds all entries in L2 table corresponding to VLAN 100
sea vlan stg=1	Finds VLANs belonging to Spanning Tree Group 1

2.4.3 中断等其他命令

1. Interrupts: 该命令为我们提供了对某块芯片的中断管理能力，具体例子如表 1-22

表 1-22 Interrupts 的使用

intr names	Lists all interrupt names that you can use in other INTR commands
intr pending	Shows pending interrupts on the current chip
intr mask	Shows which interrupts are currently enabled
intr disable ChainDone3	Disable DMA3
intr enable miimopdone	Enable MII Operation Complete Interrupt



2. Counters 关于计数器的命令，如表 1-23 所示

表 1-24 计数器命令

l-c	Lists all counter register names
g gtpkt	Prints transmit packet counters for all GE ports
show counters	Display all non-zero counters for the current chip
show c changed ge0	Show only non-zero counters that have changed on ge0
show c same Z ge	Show counters that haven't changed on all GE ports including zero counters
show statistics all	Show all non-zero SNMP counters for all ports
show s ge0 all	Show all SNMP counters for ge0 port

2.5 高级配置命令

2.5.1 Vlan 管理命令

表 1-24 vlan 管理命令

vlan create 5 pbm=ge5-ge10 ubm=ge6-ge8	Creates VLAN 11 with member ports ge5-ge10. Only member ports can transmit frames belonging to the VLAN. Ports ge6-ge8 will transmit them untagged, ge5,ge9,ge10 will tag them.
port ge0-ge5 VlanFilter=on	After applying this command, the specified ports can receive frames only from the VLANs they belong to
vlan show	Shows current VLANs configuration
vlan add 5 pbm=ge0-ge3	Add ports to ge0-ge3 to an existing VLAN with ID 5
vlan destroy 5	Deletes the VLAN with ID 5
Vlan protocol add pbm=ge frame=1 ether=0x806 vlan=2 prio=7	Make all untagged ARP packets(ethertype 0x806) go to vlan 2 with priority 7
Vlan ip4 add ipaddr=192.168.1.0 netmask=255.255.255.0 vlan=3	Make all untagged packets with source ip address 192.168.1.0/24 go on vlan 3
Vlan mac add mac=00:00:01:00:00:05 vlan=5	Make all untagged packets with MAC SA 00:00:01:00:00:05 go on vlan 5
Pvaln set ge0-ge5 5	Tell the chip to assign VID=5 to untagged (or priority tagged) packets arriving on ge0-ge5
pvlan show	Shows which VLANs are assigned to untagged packets that arrive on a specific port
pvlan set ge0-ge5 5	Tells the chip to assign VID=5 to untagged (or priority tagged) packets arriving on ge0-ge5

```

BCM.0> vlan show
vlan 1 ports cpu,ge,hg (0x00000000000003fff), untagged ge
(0x00000000000000fff)
BCM.0> vlan add 5 pbm=ge0-ge3
VLAN: ERROR: Entry not found
BCM.0> vlan create 5 pbm=ge5-ge10 ubm=ge6-ge8
BCM.0> vlan show
vlan 1 ports cpu,ge,hg (0x00000000000003fff), untagged ge
(0x00000000000000fff)
vlan 5 ports ge5-ge10 (0x00000000000007e0), untagged ge6-ge8
(0x000000000000001c
0)
BCM.0> vlan add 5 pbm=ge0-ge3

BCM.0> vlan show
vlan 1 ports cpu,ge,hg (0x00000000000003fff), untagged ge
(0x00000000000000fff)
vlan 5 ports ge0-ge3,ge5-ge10 (0x00000000000007ef), untagged ge6-ge8
(0x00000000
0000001c0)
BCM.0> port all VlanFilter=on
PORT: Error: Could not set port hg0 information: Feature unavailable
BCM.0> port ge0-ge5 VlanFilter=on
BCM.0> vlan show
vlan 1 ports cpu,ge,hg (0x00000000000003fff), untagged ge
(0x00000000000000fff)
vlan 5 ports ge0-ge3,ge5-ge10 (0x00000000000007ef), untagged ge6-ge8
(0x00000000
0000001c0)
BCM.0> vlan destroy 5
BCM.0> vlan show
vlan 1 ports cpu,ge,hg (0x00000000000003fff), untagged ge
(0x00000000000000fff)

```

2.5.2 L2 命令

表 1-25 L2 命令使用

l2 watch start	Starts a thread monitoring dynamic L2 table entry adding and deletion, that is , the MAC addresses learned by the chip will be shown on the console instantly
l2 add mac=0x001122334455 vlan=10 pbm=ge0	Adds an L2 table entry
l2 add mac=0x000011223300 vlan=100 pbm=ge	vlan=100 pbm=ge Adds incrementing entries for all ge ports
l2 show pbm=ge0-ge3	Shows L2 entries for ports ge0-ge3
l2 clear vlan=100	Deletes all l2 entries with VLAN ID 100
l2 hash mac=0x112233445566 v=10	Get an L2 table bucket index for a (potential) entry

```

BCM.0> l2 show
BCM.0> l2 watch start
BCM.0>
L2 ADD: mac=00:00:00:00:00:01 vlan=1 modid=0 port=0/ge0 Hit
L2 ADD: mac=00:00:00:00:00:02 vlan=1 modid=0 port=1/ge1 Hit
BCM.0> l2 show
0:ge0: mac=00:00:00:00:00:01 vlan=1 modid=0 port=0/ge0 Hit
0:ge1: mac=00:00:00:00:00:02 vlan=1 modid=0 port=1/ge1 Hit
BCM.0>
BCM.0> l2 add mac=0x001122334455 vlan=10 pbm=ge0
ADD: mac=00:11:22:33:44:55 vlan=10 modid=0 port=0/ge0
L2 ADD: mac=00:11:22:33:44:55 vlan=10 modid=0 port=0/ge0

BCM.0> l2 show
0:ge0: mac=00:00:00:00:00:01 vlan=1 modid=0 port=0/ge0 Hit
0:ge0: mac=00:11:22:33:44:55 vlan=10 modid=0 port=0/ge0
0:ge1: mac=00:00:00:00:00:02 vlan=1 modid=0 port=1/ge1 Hit
BCM.0> l2 clear v=10
L2: Removing static and non-static addresses by VLAN
L2 DEL: mac=00:11:22:33:44:55 vlan=10 modid=0 port=0/ge0
BCM.0>l2 show
0:ge0: mac=00:00:00:00:00:01 vlan=1 modid=0 port=0/ge0 Hit
0:ge1: mac=00:00:00:00:00:02 vlan=1 modid=0 port=1/ge1 Hit
BCM.0> l2 hash mac=00:00:00:00:00:01 vlan=1
Hash[0] of key 0x0010000000000001 is bucket 0x181 (385)

```

2.5.3 L3 命令

L3 命令包括两类：L3 控制类和 IPMC 类

L3 控制命令有：l3 intf:管理 L3 层接口命令

l3 table: 管理 L3 表命令

l3 defip:管理路由表命令

l3 ip6host: 管理基于 ipv6 地址的 L3 表

l3 ip6route:管理基于 ipv6 地址的路由表

IPMC 控制命令有：ipmc table: 管理 IPMC 组命令

Ipmc repl: 管理 IPMC 响应命令

具体使用例子如表 1-26 所示

表 1-26 L3 命令使用

L3 intf mac=00:00:01:00:00:fe vlan=1 intf=1	创建一个 IP 接口，这个命令会在 L2 表上创建一个新的条目，并且在 L3_INTF 表的 L3bit 位使能
L3 intf add mac=00:00:02:00:00:fe valn=2 intf=2	创建第二个 IP 接口
L3 l3table add ip=192.168.2.1 intf=2 mac=00:00:02::00:00:01 port=ge0	创建一个 L3 条目（地址解析条目），给地二层接口上的的主机 IP 地址为 192.168.2.1 和 MAC 地址为 00:00:02::00:00:01

L3 defip add ip=192.168.3.0 mask=255.255.255.0 gateway=192.168.2.3 mac=00:00:02:00:00:03 intf=2 port=ge3	为 ip192.168.3.0 创建一个 IP 路由通过网关 192.168.2.3，一个 L3 层的地址解析条目将会自动的添加到 L3 表中
Ipmc config enable=1	IPMC 使能
Ipmc table add src_ip =192.168.1.1 vlan=1 mc_ip=244.0.0.1 l3_map=ge0-ge5 Nocheck=1	添加 IPMC 条目给 244.0.0.1，无需检查源端口，这个命令将创建 IPMC 组 0
Ipmc repl add MCG=0 pbm=ge0-ge2 vlan=2	按照所需添加 IPMC 的复本。在 XGS3 芯片中，vlan 参数实际上是指接口数
Ipmc repl add MCG=0 pbm=ge2-ge5 vlan=3	

```

BCM.0> vlan show
vlan 1  ports cpu,ge,hg (0x00000000000003fff), untagged ge (0x0000000000000fff)
BCM.0> vlan create 2 pbm=ge0-ge3
BCM.0> vlan create 3 pbm=ge4-ge7
BCM.0> l3 intf add vlan=2 mac=0x500
BCM.0> L2 ADD: mac=00:00:00:00:05:00 vlan=2 modid=0 port=0/ge0 Static L3
L2 ADD: mac=00:00:00:00:05:00 vlan=2 modid=0 port=0/ge0 Static L3

BCM.0> l3 intf add vlan=3 mac=0x500
BCM.0> L2 ADD: mac=00:00:00:00:05:00 vlan=3 modid=0 port=0/ge0 Static L3
L2 ADD: mac=00:00:00:00:05:00 vlan=3 modid=0 port=0/ge0 Static L3

BCM.0> l3 intf show
Free L3INTF entries: 510
Unit 0 Entry Mac Address          VLAN ID
      0 00:00:00:00:05:00          2
      1 00:00:00:00:05:00          3

BCM.0> l2 show
0:ge0:  mac=00:00:00:00:05:00 vlan=3 modid=0 port=0/ge0 Static L3
0:ge0:  mac=00:00:00:00:05:00 vlan=2 modid=0 port=0/ge0 Static L3
BCM.0> l3 l3table add ip=192.168.1.2 mac=0x202 intf=1 port=ge6
BCM.0> l3 l3table show
Free L3 table entries: 4095
Unit 0 Entry IP address          Mac Address          INTF PORT HIT
L3      3904 192.168.1.2          00:00:00:00:02:02      1      6 no
BCM.0> l3 l3table destroy ip=192.168.1.2
BCM.0> l3 l3table show
Free L3 table entries: 4096
Unit 0 Entry IP address          Mac Address          INTF PORT HIT

BCM.0> l3 intf destroy in=0
BCM.0> l3L2 DEL: mac=00:00:00:00:05:00 vlan=2 modid=0 port=0/ge0 Static L3
L2 DEL: mac=00:00:00:00:05:00 vlan=2 modid=0 port=0/ge0 Static L3
BCM.0> l3 intf show
Free L3INTF entries: 511
Unit 0 Entry Mac Address          VLAN ID
      1 00:00:00:00:05:00          3

```


2.5.4 Mirroring 命令

镜像命令允许数据包被复制到某个特定端口来观察。需要注意的是，基于端口的镜像命令是由一个逻辑入口和一个逻辑出口来支持的（Note that port-based mirroring is supported on both an ingress and egress basis.）Mirroring 命令的使用如表 1-27 所示。

表 1-27 Mirroring 命令的使用

mirror mode=l2 port=ge0	Mirror all traffic from all ports to ge0
mirror mode=l2 port=ge7 IngressBitMap=ge0	All packets received by the ingress port of ge0 are mirrored to ge7. Note that this occurs even though ge7 and ge0 are not in the same VLAN
mirror mode=l2 port=ge7 EgressBitMap=ge0	All packets sending by the egress port of ge0 are mirrored to ge7
mir	Displays current settings of port mirroring

2.5.5 Filtering

Filtering 命令的作用是创建简单的过滤器，将数据包传向 CPU。如果你需要做一些大的改动，那么就需要直接编辑 GIRULE (or GIMASK)表。Filtering 命令的使用如表 1-28 所示。

表 1-28 Filtering 命令的使用

filter create dm=0x000102030405	Trap all packets with the destination MAC 00:01:02:03:04:05 to the CPU
filter show	Shows existing filter settings
mod girule.ge0 0 1 ACTION=0xc	Modifies the filter so that packets will go to CPU only

2.5.6 Spanning Tree Group Management

STG 命令用于管理生成树组和生成树的应用，它的使用如表 1-29 所示

表 1-29 STG 命令的使用

stg show	Shows the existing STP groups. Group 1 is the default one.
stg stp 1	Shows STP states for all ports in STG 1.
stp 1 start	Start STP for STG 1
stp 1 show details	Shows the STP state for STG 1
stp 1 root	Show the current root
stg stp 1 all forward	Show the current root

2.6 错误检测命令

2.6.1 Text

Text 命令帮助用户来验证芯片是否正常工作，它的使用如表 1-30 所示.

表 1-30 Text 命令的使用

TestList	Lists available tests
TestSelect +41	Adds test #41 to the list of selected tests
TestSelect -40	Removes test #40 from the list of selected tests
TestMode StopOnError	Chooses the mode to run tests
TestClear *	Clears statistics for all the tests
TestRun	Runs all the selected tests
TestRun 41	Runs only the specified test

2.6.2 CONFig

表 1-31 CONFig 命令的使用

config	Displays the configuration
config add pbmp_valid.1=0xf	Enables only ports 0-3 on unit 1. This will take effect only after the re-initialization of the driver
config pbmp_valid.1=0xff	Changes the value of the existing config variable
config delete pbmp_valid.1	Deletes the config variable (thus re-enabling all the ports).

3 Reference Documents 参考文献

- 1) BCM_CLI_Quick_Start_Guide.pdf
- 2) Generic SDK Introduction .pdf