sdk调试与丢包处理

sdk常用调试手段

sdk中所有端口描述都是从0开始

- 包转发情况查看

```
CDL_CLI(kgxx-sdk)# show packet fwd info
Forward: = {
                          InPort : 13,            //包入端口
                       PktLen(B) : 4a,            //包长 0x4a
                          FwdBmp : 2000000,       //转发端口情况16进制表示，每1bit代表一个转发口，上cpu时31口为1
                         FwdType : Ucast,         //转发类型：Ucast Mcast Bcast NAT
                        ProcType : Bridging,      //处理方式：Bridging、Routing
                          VlanID : b,             //转发的vlan（可以是经过vlan划分等功能之后的）
                        Priority : 0,             //内部优先级
                          L2Type : EthII,         //l2协议类型 "Nop"表示未识别
                          L3Type : Nop,           //l3协议类型
                          L4Type : Nop,           //l4协议类型
                     UDP-Payload : Nop,           //udp协议类型
                         VlanBmp : 0,             //标识ctag/stag生效 0untag或者未识别tag 1识别ctag 2识别stag 3识别stag&ctag
                         BrgProc : 0,             //l2转发使能  fdb转发
                       RouteProc : 0,             //route转发 主机路由，网段路由
                         AclProc : 0,             //acl enable
                      Ipv6AclLkp : 0,             //ipv6 acl表命中条目
                      Ipv4AclLkp : 0,             //ipv4 acl表命中条目
                          AclHit : 0,             //mac acl表命中条目
                       AclHitIdx : 0,             //被查表命中索引 用于查找TblAclQos action
                           SFlow : 0,             //使能acl log  TblAclQos randomLogEn
                        FlowSpan : 0,             //使能acl 流镜像 TblAclQos mirrorEn
                      BrgHitLeft : 0,             //fdb mac left hit
                     BrgHitRight : 0,             //fdb mac right hit
                       BrgHitIdx : 0,             //fdb mac 命中索引 用于查找TblMacLeft/Right action
                     HostHitLeft : 0,             //主机路由 route left hit
                    HostHitRight : 0,             //主机路由 route right hit
                      HostHitIdx : 0,             //网段路由命中(HostHitLeft/Right都没有命中) 查TblRoute action  主机路由命中索引 用于查找TblHostRouteLeft/Right action
                         LpmLkp : 0,              //主机路由表命中条目
                         LpmIdx : 0,              //网段路由表命中条目
                      NatHitIdx : 0,              //NAT命中查TblRoute action
}
//l2协议类型 "EthII", "SAP", "SNAP"
```

```
//l3协议类型 "Ipv4", "V4Arp", "MSRP", "1722", "V4Rarp", "Ipv6", "SLOW", "Eapol",
"LLDP", "MMRP", "MVRP", "PTP", "NCMT", "ISIS", "UDF"
//l4协议类型 "TCP", "UDP", "ICMP", "IGMP", "V6ICMP", "MLD", "ND", "L4UDF"
//udp协议类型 "RTP", "RTCP", "SNMP", "PTP", "BFD"
```

- 丢包情况查看

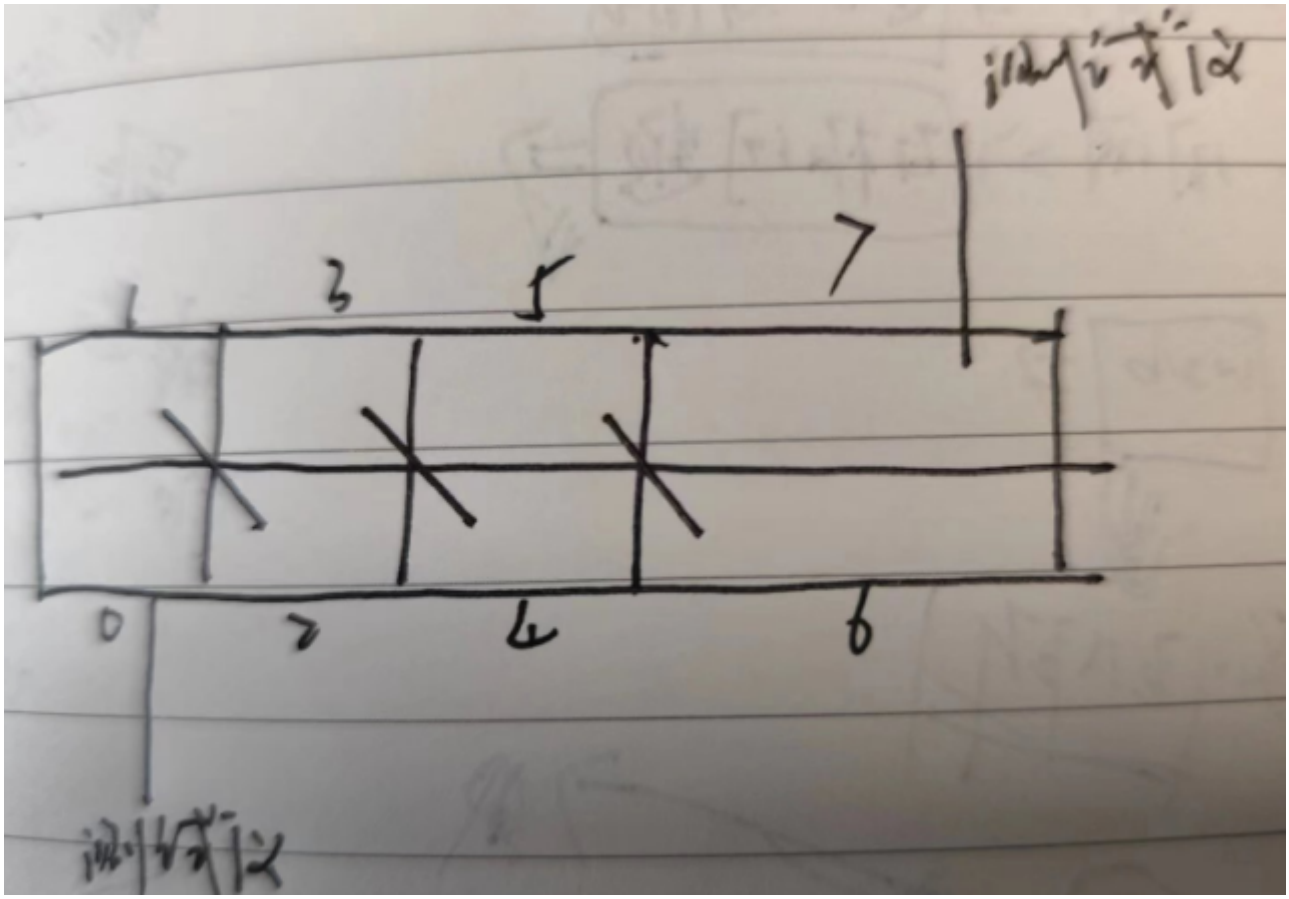
```
CDL_CLI(kgxx-internal)# show packet discard info
该命令只能在完全丢包情况下有效，部分丢包，大部分情况看不到丢包情况。
Discard: = {
                      L2PrsError : 0,           //二层头解析错误
                       MacSaEqDa : 0,           //macSaEqMacDaDrop  包MacSa ==
MacDa
                     TcpCtl0Seq0 : 0,           //tcpFlagsCtl0Seq0En tcp Control
flags == 0 sequence number == 0
                      IpsaEqIpda : 0,           //sipEqDipDrop Ipsa == IpDa
                       UdpSpEqDp : 0,           //udpSportEqDportDrop udp source
port == dest port
                       TcpSpEqDp : 0,           //tcpSportEqDportDrop tcp source
port == dest port
                      BothSynFin : 0,           //tcpFlagSynFinDrop 同时设置TCP标志
位Syn和Fin
                    FinUrgPshSeq0 : 0,          //tcpFlagFinUrgPshSeq0Drop 同时设
置TCP标志Urg、Fin和Psh，且序号为0
                      FragOfset1 : 0,           //tcpFragOffset1Drop tcp fragment
offset == 1
                      PartialTcp : 0,           //icmpFragDrop 第一个分片TCP报文，
第4层报头长度小于最小报头长度，且不是分片报文
                        FragICMP : 0,           //TblDosControl ICMP fragment
packet
                     OverLenPing : 0,           //icmpV4MaxSize/icmpV6MaxSize ping
包长大于设置的最大值
                       HWDiscard : 0,           //说明丢包来自mac
                      802.1xDrop : 0,           //TblPhyPort dot1xEn=1 dot1xDrop=1
                     LoopPktDrop : 0,           //check source mac fail
                  PtpPkt!PtpPort : 0,           //?? TblClanClassifyLog ptpDiscard
                         AftDrop : 0,           //AcceptFrameType vlan tag过滤
                      McVlan!Mc : 0,            //?? TblIntfMapLog mcVlanDrop
                        AclDrop :  0,           //acl hit deny
                      RouteDrop : 0,            //route hit drop
                      RouteExcp : 0,            //route未命中drop
                       BrgToSrc : 0,            //fdb action source mac discard
                       BrgForce : 0,            //fdb action dest mac discard
                        StpFail : 0,            //stp drop eg:block
                      StormDrop : 0,            //设置风暴抑制
                        LrnLock : 0,            //mac学习锁定 TblPortLearnCtrl
lock = 1
                      LrnExceed : 0,            //mac学习限制生效丢弃
TblPortLearnCtrl irnNumExceedDiscard=1
                      MeterDrop : 0,            //设置policer
                   MeterDropCnt : 0,            //policer丢包个数
                  IgrVlanFilter : 0,            //TblSrcPort ingressFilteringEn=1
```

```
入端口vlan过滤
                    EgrPortFilter : 0,           //TblSrcPortEgressMask
portEgressMask 设置mask中没有对应的端口
                    EgrVlanFilter : 0,           //TblFwdEgressFilter
egrFilteringEn=1
                        LAGFilter : 0,           //TblMcastPort applyVlanMask=1
Multicast Vlan Filtering
                    DestLostInCfg : 0,           //?? TblFwdLog destLostInCfg
                       EgrStpFail : 0,           //stp egress drop
                          EgrDrop : 0,           //?? TblEgrLog egrDiscard
                          EgrExcp : 0,           //?? TblEgrLog egrExcpType
}
```

丢包测试

- 测试方式 sdk丢包测试一般采用蛇形方式，测试步骤如下：

1. 如下图所示对板子进行蛇形连接，1与2、3与4、5与6分别用网线直连。



2. 打开sdk，切换到exec_node模式，执行snake脚本 `CDL_CLI# load config file ./config/bridge_snake.cfg`

3. 0口打流7口接收，也可双向打流，流格式默认即可。

4. 观察测试仪包统计情况，是否存在丢包。

丢包调试方式

1. 包进mac情况调试 包进mac调试主要看包是否可进入mac，进入mac的数量是否对，以及包是rx出问题还是tx出问题了。

   若是能稳定复现的丢包，建议打burst流来统计流情况。 建议打流测试之前，使用清空命令对info进行清理，防止其他的一些干扰。

包统计情况

```
CDL_CLI(kgxx-sdk)# show mib all Gmac0 info
---------------------------------Gmac0-------------------------------------
|ItemName                |EMac Frames Counter     |EMac Bytes Counter      |PMac
Frames Counter     |PMac Bytes Counter      |
-------------------------------------------------------------------------------
|ReceivedGoodUCast       |10                      |1280                    |0
|0                       |
|ReceivedGoodMCast       |0                       |0                       |0
|0                       |
|ReceivedGoodBCast       |0                       |0                       |0
|0                       |
|ReceivedGoodPause       |0                       |0                       |0
|0                       |
|ReceivedGoodControl     |0                       |0                       |0
|0                       |
|ReceivedJabber          |0                       |0                       |0
|0                       |
|ReceivedCollisionFrag   |0                       |0                       |0
|0                       |
|ReceivedFcsError        |0                       |0                       |0
|0                       |
|Reserve                 |0                       |0                       |0
|0                       |
|Reserve                 |0                       |0                       |0
|0                       |
|ReceivedGoodOversize    |9                       |13680                   |0
|0                       |
|ReceivedGoodUndersize   |9                       |522                     |0
|0                       |
|ReceivedGood63B         |9                       |522                     |0
|0                       |
|ReceivedBad63B          |0                       |0                       |0
|0                       |
|ReceivedGood1519B       |9                       |13680                   |0
|0                       |
|ReceivedBad1519B        |0                       |0                       |0
|0                       |
|ReceivedGoodJumbo       |0                       |0                       |0
|0                       |
|ReceivedBadJumbo        |0                       |0                       |0
|0                       |
|Received64B             |0                       |0                       |0
|0                       |
|Received65to127B        |0                       |0                       |0
```

```
|0                        |
|Received128to255B        |10                       |1280                     |0
|0                        |
|Received256to511B        |0                        |0                        |0
|0                        |
|Received512to1023B       |0                        |0                        |0
|0                        |
|Received1024to1518B      |0                        |0                        |0
|0                        |
|TransmittedUCast         |0                        |0                        |0
|0                        |
|TransmittedMCast         |0                        |0                        |0
|0                        |
|TransmittedBCast         |0                        |0                        |0
|0                        |
|TransmittedPause         |0                        |0                        |0
|0                        |
|TransmittedControl       |0                        |0                        |0
|0                        |
|TransmittedLessThan64B   |0                        |0                        |0
|0                        |
|Transmitted64B           |0                        |0                        |0
|0                        |
|Transmitted65to127B      |0                        |0                        |0
|0                        |
|Transmitted128to255B     |0                        |0                        |0
|0                        |
|Transmitted256to511B     |0                        |0                        |0
|0                        |
|Transmitted512to1023B    |0                        |0                        |0
|0                        |
|Transmitted1024to1518B   |0                        |0                        |0
|0                        |
|Transmitted1519BtoMTU    |0                        |0                        |0
|0                        |
|TransmittedJumbo         |0                        |0                        |0
|0                        |
|TransmittedMacUnderrun   |0                        |0                        |0
|0                        |
|TransmittedFcsError      |0                        |0                        |0
|0                        |
|TxExcessiveDeferral      |0                        |0                        |0
|0                        |
|TxLateCollision          |0                        |0                        |0
|0                        |
|TxExcessiveCollision     |0                        |0                        |0
|0                        |
|TxOneCollision           |0                        |0                        |0
|0                        |
|TxMultipleCollision      |0                        |0                        |0
|0                        |
|TransmittedDeferral      |0                        |0                        |0
|0                        |
-------------------------------------------------------------------------------
```

参数解释：
RxSum //rx统计情况
ReceivedGoodPause 收到pause帧的统计
ReceivedJabber 收到超时传输帧统计情况
ReceivedCollisionFrag 端口半双工出现冲突帧的统计
ReceivedFcsError 检验出错的帧统计情况
ReceivedGoodOversize 小于64长度的帧统计
ReceivedGoodUndersize 大于1518长度的帧统计

RxRange //rx基于包长的详细统计
ReceivedGood63B 收到小于64长度正确帧的统计
ReceivedBad63B 收到小于64长度错误帧的统计
ReceivedGood1519B 收到大于1518长度正确帧的统计
ReceivedBad1519B 收到大于1518长度错误帧的统计
ReceivedGoodJumbo 收到好的巨型帧统计
ReceivedBadJumbo 收到坏的巨型帧统计
Received64B
Received65to127B
Received128to255B
Received256to511B
Received512to1023B
Received1024to1518B

TxSum //tx统计情况
TransmittedUCast
TransmittedMCast
TransmittedBCast
TransmittedPause
TransmittedControl

TxRange //tx基于包长的详细统计
TransmittedLessThan64B
Transmitted64B
Transmitted65to127B
Transmitted128to255B
Transmitted256to511B
Transmitted512to1023B
Transmitted1024to1518B
Transmitted1519BtoMTU
TransmittedJumbo

TxError //tx错误情况统计
TransmittedMacUnderrun 主机无法以足够快的速度提供发送数据
TransmittedFcsError 发送校验出错的帧统计
TxExcessiveDeferral 发送MAC在超过两个最大以太帧的时间内仍无发送机会
TxLateCollision 发送完512bit后才检测冲突
TxExcessiveCollision 检测到冲突超过15次
TxOneCollision
TxMultipleCollision
TransmittedDeferral

b. 清包方式

```
CDL_CLI(kgxx-sdk)# clear Gmac0 mibinfo
```

2. 包在mac内部转发调试

a. 包转发情况查看

```
CDL_CLI(kgxx-sdk)# show packet fwd info

CDL_CLI(kgxx-sdk)# show packet discard info
该命令只能在完全丢包情况下有效，部分丢包，大部分情况看不到丢包情况。

CDL_CLI(kgxx-sdk)# show inout port Gmac0 info
-----------------------------------------------Interface Frames Statistic---------
-------------------------------------
Gmac0:  |RxESop  |RxEEop  |RxEByte |RxEErro |        |RxPSop  |RxPEop  |RxPByte
|RxPErro |        |TxESop  |TxEEop  |TxEByte |TxEErro |        |TxPSop  |TxPEop
|TxPByte |TxPErro |        |sync
  |Spd     |CErr    |Merg    |
0       |60      |60      |eb      |70      |        |0       |0       |0       |0
|        |de      |de      |0       |0       |        |0       |0       |0
|0       |        |
1       |1       |a7d9    |0       |
-------------------------------------------------------------------------------------
-------------------------------------
该命令可以用来查看当前状态下mac的进包情况，一般用来debug包是否进mac，由于不可清除，不建议用RxPErro|RxEErro|TxPErro|TxEErro作为帧出错的标志，需要看是否有错帧建议使用上面说的mibinfo.
```

b. 包转发寄存器情况查看

```
//用于调试当前包的入队列情况
CDL_CLI(kgxx-sdk)# inspectentry TmAdmInOutCnt 0
TmAdmInOutCnt : {
"tmWriteInCnt" : "0x8250",
"igrInCnt" : "0x8250",
"tmRepOutCnt" : "0x8250",
"tmSchOutCnt" : "0x8250",
"tmAdmAllDropCnt" : "0x0",
"tmAdmIgrDropCnt" : "0x0", //入队列丢包情况
"tmAdmEgrDropCnt" : "0x0", //出队列
"tmAdmInCntPri0" : "0x50", //代表包进入队列0
"tmAdmInCntPri1" : "0x0",
"tmAdmInCntPri2" : "0x0",
"tmAdmInCntPri3" : "0x0",
"tmAdmInCntPri4" : "0x0",
"tmAdmInCntPri5" : "0x0",
"tmAdmInCntPri6" : "0x0",
"tmAdmInCntPri7" : "0x0"
```

```
}

//用于调试当前包的出队列情况
CDL_CLI(kgxx-sdk)# inspectentry TmSchDebug 0
TmSchDebug : {
"tmSchInCnt" : "0x8251",
"tmSchOutCnt" : "0xeed1",
"tmSchFreeListCnt" : "0x1000",
"tmSchEnqPri0" : "0x51",  //代表包进入队列0
"tmSchEnqPri1" : "0x0",
"tmSchEnqPri2" : "0x0",
"tmSchEnqPri3" : "0x0",
"tmSchEnqPri4" : "0x0",
"tmSchEnqPri5" : "0x0",
"tmSchEnqPri6" : "0x0",
"tmSchEnqPri7" : "0x0",
"tmSchDeqPri0" : "0xd1", //代表包从队列0出去
"tmSchDeqPri1" : "0x0",
"tmSchDeqPri2" : "0x0",
"tmSchDeqPri3" : "0x0",
"tmSchDeqPri4" : "0x0",
"tmSchDeqPri5" : "0x0",
"tmSchDeqPri6" : "0x0",
"tmSchDeqPri7" : "0x0",
"tmSchEnqBmp" : "0x2000000",
"admEgrDropLeftCnt" : "0x0"
}

//代表包转发log
CDL_CLI(kgxx-sdk)# inspectentry CtlFwdLog 0
CtlFwdLog : {
"fwdBitmap" : "0x2000000", //转发到端口
"fwdBitmapHi" : "0x0",
"vlanDiscard" : "0x0",
"mcastFlood" : "0x0",
"destMap" : "0x0",
"opCode" : "0x0",
"ucastFlood" : "0x0",
"criticalPacket" : "0x1",
"forbidEdit" : "0x0",
"redirPtp" : "0x0",
"egrPortFilted" : "0x0",
"egrVlanFilted" : "0x0",
"lagFilted" : "0x0",
"isPtp" : "0x0",
"ptp2Cpu" : "0x0",
"addRtag" : "0x0",
"tsnGateId" : "0x0",
"tsnCycle" : "0x0",
"cpuPktType" : "0x0",
"destLostInCfg" : "0x0"
}
```

增加sgmii外环设置用于在mac收包直接转到phy，屏蔽mac影响 CDL_CLI(tsn_v5-sdk)# port 8 loopback-sgmii-out enable