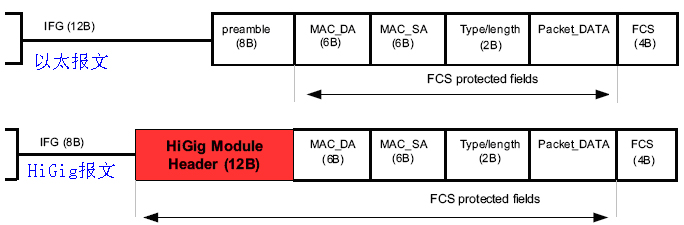
此文章已于 11:42:08 2016/8/4 发布到 MADHEX

HiGig/HiGig+/HiGig2

   HiGig(通常称为HiGigTM)是Broadcom公司的私有串行总线互联方案，于2001年推出，主要用于Broadcom公司StrataXGS系列芯片(如BCM5670/BCM5690等)之间的互联(也可以跟支持HiGig协议的NPU或ASIC连接)，既可用于板内连接，也可通过背板走线形式实现跨板连接。

HiGig总线是在以太网协议的基础上发展而来的，它在以太网二层报文中插入HiGig头，形成HiGig报文，通过HiGig头部携带的控制信息，来实现芯片端口的镜像、聚合、QOS等功能。



    如上图所示，将以太网二层报文的8Byte前导码和4个字节帧间隙(共12个字节帧间隙)替换成12个字节的HiGig报文头，这样，HiGig报文只有8个字节帧间隙，没有前导码。

HiGig接口支持的最大速率为10Gbps(共4对SerDes通道，每通道最大支持3.125Gbps，因为经过了8B/10B变换，所以有效带宽为2.5Gbps)，物理层电气特性如XAUI端口相同(详见IEEE802.3ae clause 47)。

Broadcom公司在其StrataXGS II系列产品上(如BCM5675/BCM5695等)推出了HiGig+总线，HiGig+只是在HiGig的基础了做了细微改进，将端口支持的最大速率从10Gbps提高到12Gbps(每个通道的最大速率从3.125Gbps提高到3.75Gbps),至于协议部分，没有做任何更改，与HiGig完全一样，所以对HiGig接口完全兼容。

隨着通信技术的发展，HiGig/HiGig+总线也暴露出了其自身的局限线，在对更高端的网络市场应用中显得力不从心，主要表现在以下几个方面：

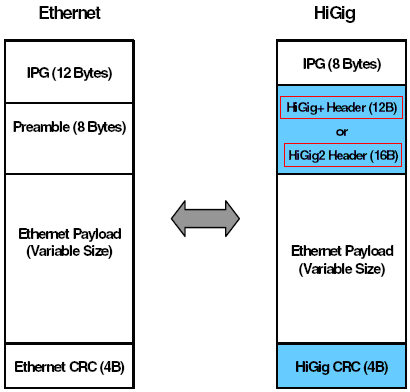
    1).地址空间和服务种类有限，满足不了更高级系统应用；

    2).头部结构不够灵活，满足不了未来技术发展；

    3).对流控和负载均衡技术支持不足。

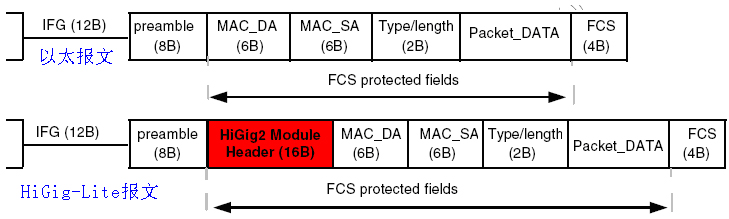
在此背景下，Broadcom公司于2005年在其StrataXGS III系列产品(如BCM56580/ BCM56700/BCM56800等)上推出了HiGig2总线，HiGig2总线在HiGig+总线的基础上，对端口速率和传输协议都进行了更改。

HiGig/HiGig+报文头为12个字节，而HiGig2报文头部增加到16个字节，使得HiGig2报文可以携带更多的信息来决定报文的处理方式，如镜像、重定向、流控和负载均衡等。如下图所示：

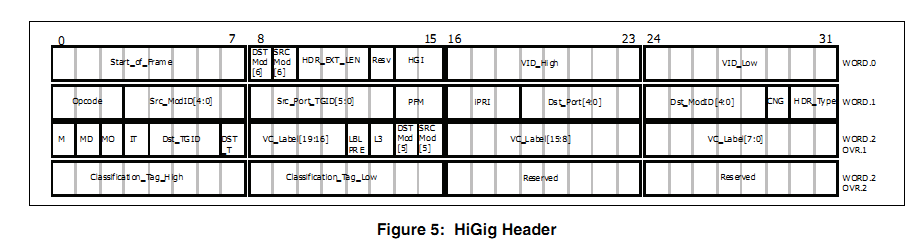


    说完HiGig/HiGig+/HiGig2后，不得不顺便说下另外一种非主流的HiGig类协议HiGig-Lite。由于HiGig-Lite不能与HiGig/HiGig+/HiGig2兼容，所以其只在很小范围内有使用(只局限在BCM5601x/BCM5620x/BCM5622x/BCM5371x几个系列芯片上)。HiGig-Lite端口只支持2.5Gbps速率，报文帧结构同XGMII类似。

HiGig-Lite协议不像HiGig/HiGig+/HiGig2那样将报文的前导码和部分帧间隙替换成HiGig/HiGig+/HiGig2头部，以求变换后的报文传输效率不变，而是保持原以太报文的前导码和帧间隙不变，只是在以太报文的前导码和目的MAC地址之间插入了一个16字节的HiGig-Lite报文头，如下图所示：



## HIGIG



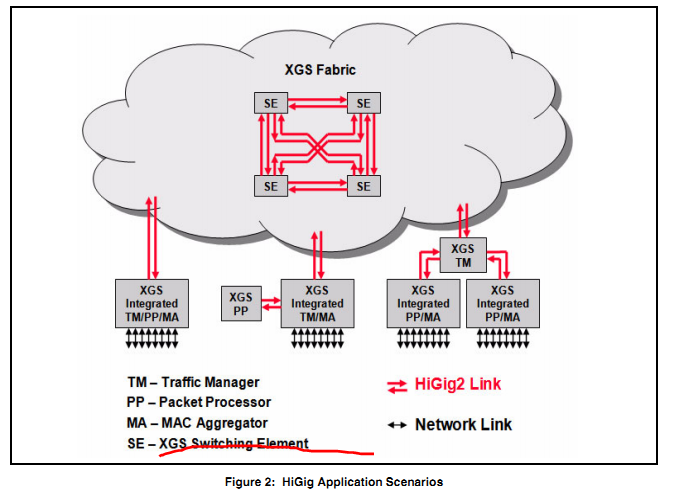
Higig（XAUI）接口有4个通道，higig header中byte0使用lane0传输，byte1使用lane1，byte2使用lane2，byte3使用lane3，byte4使用lane0，依次类推。

Higig header主要包括：START\_OF\_FRAME (SOF)、DST\_MODID、SRC\_MODID、HDR\_EXT\_LENGTH、VID\_HIGH、VID\_LOW、VID\_LOW、 SRC\_MODID、SRC\_PORT\_TGID、DST\_PORT、DST\_MODID 、HDR\_TYPE。其中HDR\_TYPE决定HDR\_TYPE之后4byte的格式。

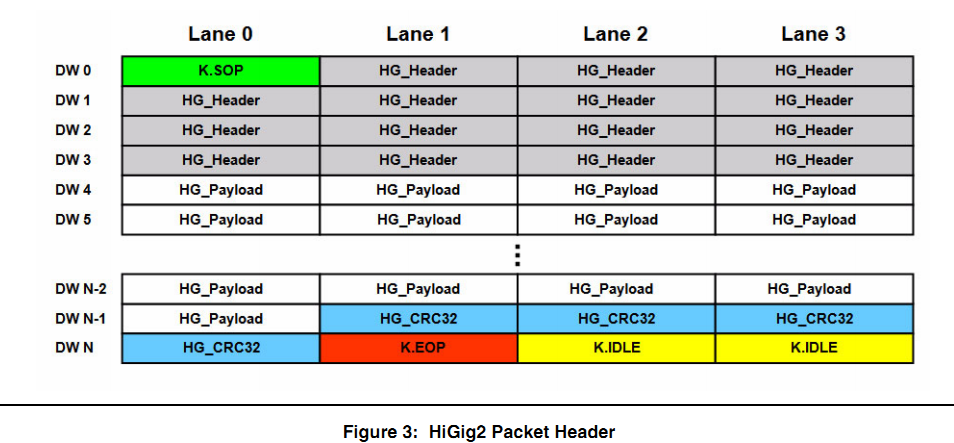
HDR\_TYPE：00 = Overlay 1 (default)，提供mirroring/trunking信息

01 = Overlay 2 (classification tag)，提供分类、过滤处理。

## HIGIG2



HIGIG2 PACKET HEADER



Higig2 Packet Header： 中包括， FRC和PPD。

1）K.SOP (0xFB): control character will always be aligned with Lane 0.即分隔符

2）K.EOP (0xFD): control character will be aligned depending on the length of HG\_Payload

3）HG\_Payload: will carry the Ethernet frame starting from the MACDA field

4）Fabric Routing Control (FRC)，占用7个字节。FRC中比较有用的是：

MCST（多播还是单播）、

TC [3:0]（用于qos）、

DST\_MODID [7:0] /MGID [15:8] （目的mode id）、

DST\_PID [7:0] /MGID [7:0]（目的端口id）、

SRC\_MODID [7:0]（源mode id）、

SRC\_PID [7:0]（源端口id）、

PPD\_TYPE（Packet Processing Descriptor Type）

5）Packet Processing Descriptor (PPD)，占用8个字节。主要是根据FRC中的PPD\_TYPE决定PPD的结构

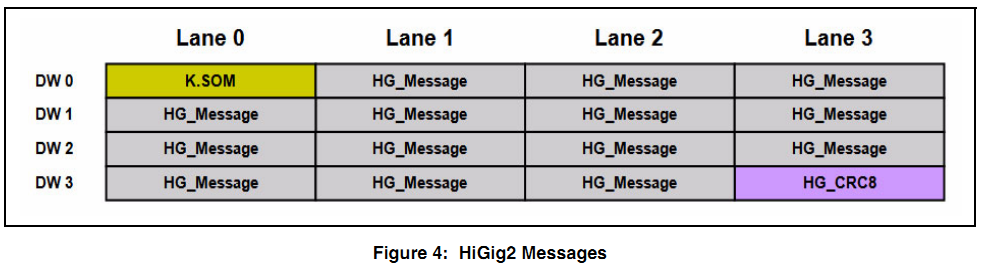
000: PPD Overlay1

001: PPD Overlay2

010~111: Reserved。

6) HG\_CRC32: will cover the bytes starting from the K.SOP to the last byte of the HG\_Payload for packet error protection.

## HIGIG2 MESSAGES

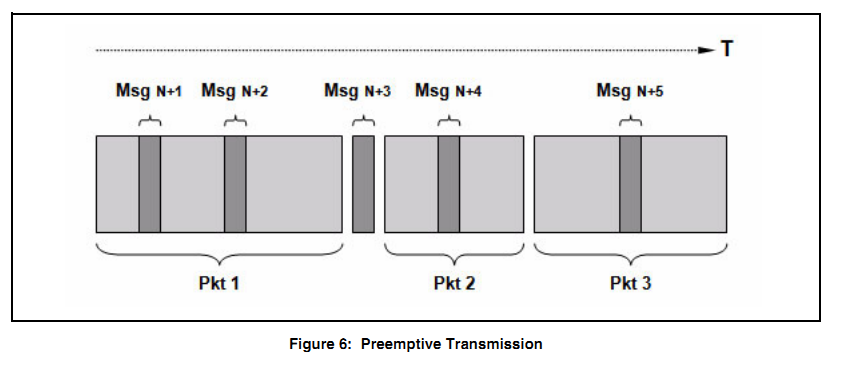


Higig2 messages占用16byte。

K.SOM:分隔符

HG\_CRC8：crc校验，从K.SOM到最后一个HG\_Message。

Higig2 messages可以出现在higig2数据包中，也可以独立成包。



Higig2 messages包括MSG\_TYPE（message类型）、FWD\_TYPE

MSG\_TYPE

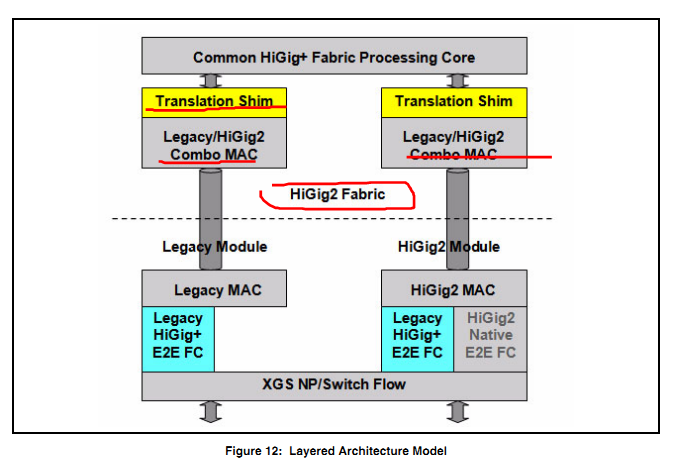
000000: Flow Control

FWD\_TYPE：

00: Link Level

Higig2能够兼容HiGig+，而HiGig+不能兼容higig2.

在支持higig2和higig+的结构，结构图如下



Combo MAC能够区分higig2 header和higig+ header.

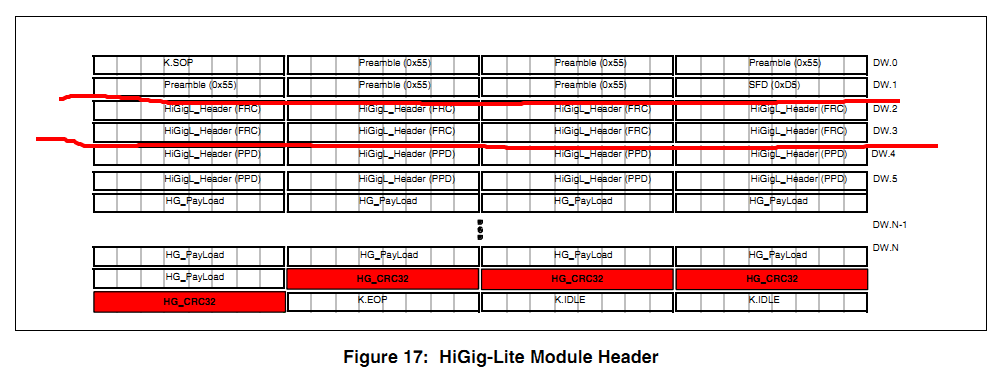
Translation shim负责转换为higig+ header。

Common HiGig+ Fabric Processing Core负责处理路由方案。

## HIGIG-LITE

HIGIG-LITE header 位于以太报文的前导码和目的MAC地址之间，占用16byte，HIGIG-LITE由两部组成：8byte的FRC和8byte的PPD。

HIGIG-LITE堆叠协议与基于HIGIG、HIGIG2的StrataXGS设备是不能兼容的，只有基于HIGIG-LITE的设备之间可以堆叠。

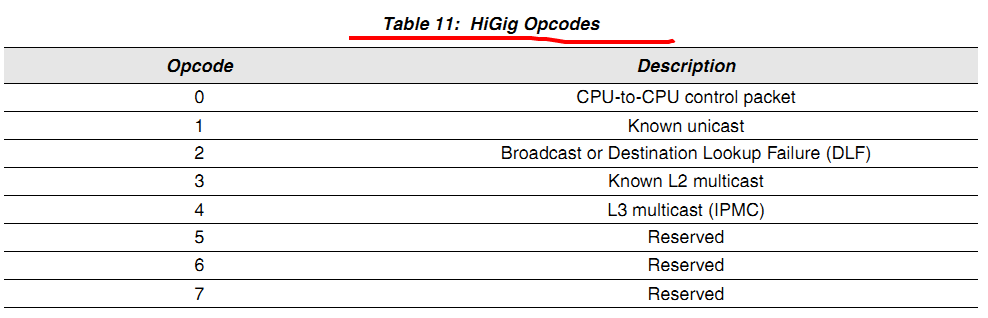


## 功能描述

每个交换设备都有唯一的一个mode id，用于转发数据包。Frabric设备没有mode id，因为他们只适用于扩展系统，而不是边沿设备，它只是需要通过mode id定位到相应的端口。

Mode id的数量是有限的，原来一个端口对应一个mode id，后来由于支持了重新映射功能，可以有多个端口对应一个mode id，通过端口数来区别。

当一个数据包需要通过higig口传递时，入口交换设备需要header域添加合适的信息，fabric设备和其他交换设备需要分析header中的opcode，来获知数据包的类型（CPU、broadcast、multicast、known unicast）。



Opcode=0：cpu控制报文。在大多数情况是发送给cpu的报文。

Opcode=1：known unicast packet。入口交换查找2层/3层 table，找到一个匹配的。在mode header中，dst\_modid 、dst\_port 制定了最后的出口。中间的Fabric、交换设备根据MH中内容来转化数据包，在目的mode，通过dst\_port来发送到对应的端口上。不需要转发表。

The ingress switch device will set opcode==1 for the following conditions:

1. Matching L2 entry

2. Matching L3 entry

3. FP match action==redirect

Opcode=2：用于广播，或者是在table中没有找到匹配的单播。Fabric、switch设备会将报文发到MH中制定的vlan中的其他成员。

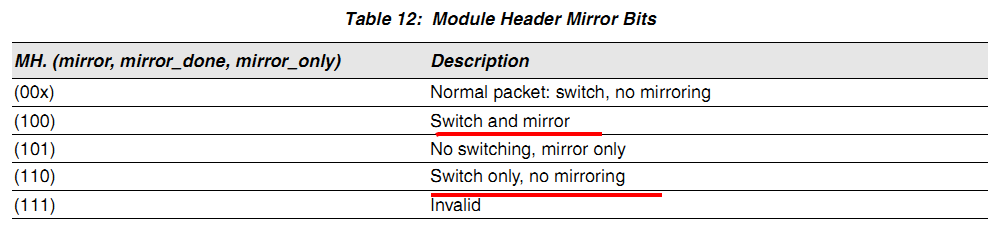
Opcode=3：KNOWN L2 MULTICAST PACKETS。入口交换查找L2/L2MC table，找到一个匹配的。在mode header中，dst\_modid 、dst\_port 指定了多播组id，中间的Fabric、交换设备根据MH中多播组id在L2/L2MC table查找，确定最后目的端口号。

Opcode=4：KNOWN L3 MULTICAST PACKETS。入口交换查找L3/L3MC table，找到一个匹配的entry，在mode header中，dst\_modid 、dst\_port指定了IPMC group ID, 中间的Fabric、交换设备将定位L3/L3MC table，使用MH中IPMC group ID，确定最后目的端口号。

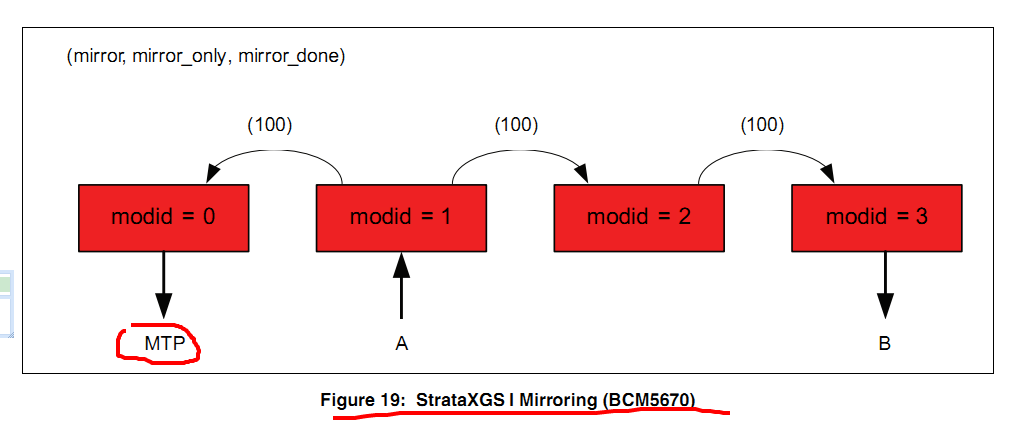
Opcode=5、6、7：无效的操作。

## 端口镜像

在StrataXGS I/II，端口镜像到镜像端口端、报文发送到目的端口，需要module header中的mirror, mirror\_done, mirror\_only确定。



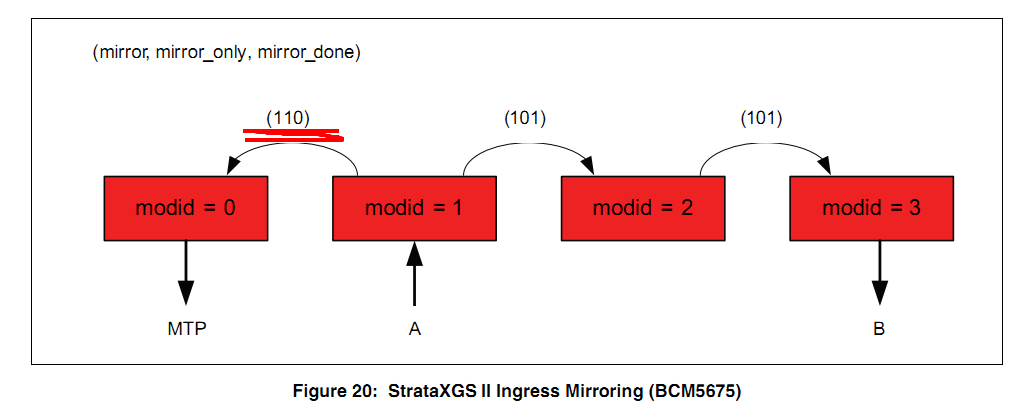
### STRATAXGS I MIRRORING (BCM5670)



(mirror, mirror\_done, mirror\_only)==(100)表示报文需要映像和交换。如果交换端口和镜像端口为同一个端口，那么只需要发送一个报文；如果不是同一个端口，需分别发送一个报文。无论是发送到交换端口还是镜像端口，报文module header域都不需要改变。

上图中MTP为镜像端口，B为交换端口

### STRATAXGS II INGRESS MIRRORING (BCM5675)



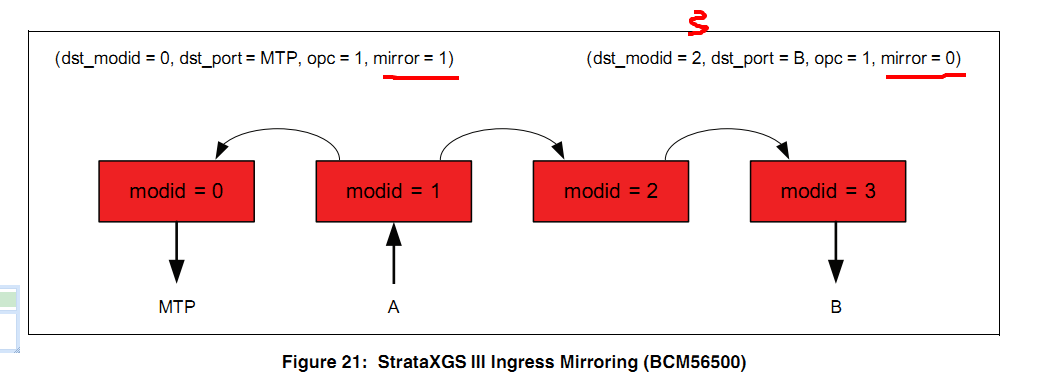
上图中MTP为镜像端口，B为交换端口。

如果映像端口与交换端口不是同一个端口，会发送两个数据包，两个端口各一个。

(mirror, mirror\_done, mirror\_only)==(110)表示报文只映射不交换

(mirror, mirror\_done, mirror\_only)==(101)表示报文值交换不映射

### STRATAXGS III INGRESS MIRRORING (BCM56500)



交换设备在经过higig会建立会有3份数据包拷贝： Switch copy、Ingress mirror copy 、Egress mirror copy

发送switch copy时，header会是MH.opcode==1 (known unicast), dst\_modid, dst\_port, mirror=0。

发送mirror copy时，header会是MH.opcode==1 (known unicast), dst\_modid, dst\_port, mirror=1。

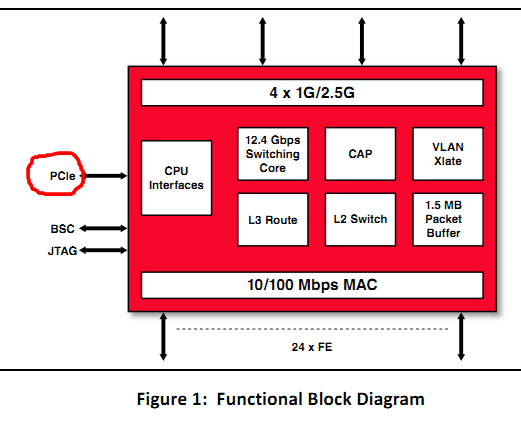
对于ingress mirror copy，报文是没有经过修改的原始报文。

### 2.5G HIGIG 模式

在2.5G higig模式下，只是用一个通道XAUI lane (0)，XAUI lane (1-3)没有使用。只有基于XAUI 的端口才支持2.5G higig模式

# 56146功能

### 56146简介



该款芯片有24个FE口，4个1G/2.5G口。

56146支持的交换功能：L2交换、l2多播、vlan、Double VLAN Tagging、VLAN Range-Based Double Tagging、端口过滤、速率风暴控制、生成树、链路汇聚、镜像、L3路由、IP多播、CFP、QOS、端口安全、DoS、Cpu协议包处理、堆叠、MIB等。

56146支持Ethernet/IEEE 802.3包长为64-1522B，jumbo数据最大为12KB，支持16K的mac地址学习，支持1K的2层多播组，支持4k vlan，链路汇聚支持128个trunk，每个端口有8个CoS队列，堆叠支持128个module，

56146的交换能力达到12.4Gps，24个FE口支持10M/100M，使用的接口是S3MII，BCM56146有2个HyperCores，每个HyperCores可以设置2个1G/2.5G port，所以有4个1G/2.5G端口，当设置为1G时，支持10/100/1000M full/half-duplex, 1000Base-X (fiber), 100Base-FX mode (fiber)模式，使用SGMII接口，当设置为2.5G时，支持HiGig-Lite或overclocked-Ethernet模式。

Cpu对56146的访问主要使用pcie接口。Port-based rate control with 8 Kbps granularity。

主要包括的表是：

1）端口表（Port Table）：每个端口在表中有一条目，记录l2学习、没用的端口、vlan处理、任务优先级等。

2）基于mac的vlan表（MAC-Based VLAN Table）：能容纳24K的mac地址，是基于源mac的vlan表，这个表与vlan转换表协同工作。

3）Vlan转换表（VLAN Translation table）：入口、出口分别2K，用于客户vlan与服务提供商vlan之间转换，与基于mac的vlan表协同工作。

4）基于协议的vlan表（Protocol-Based VLAN Table ）：每个端口16个。

5）Vlan表（VLAN Table）：容纳4K个vlan，显示每个vlan的端口和生成树组。

6）生成树组表（Spanning Tree Group Table）：容纳256个生成树组，显示每个生成树组中每个端口的生成树状态。

7）mac地址表（MAC Address Table）：容纳16K个mac地址，包括学习到的mac和编写的mac，表示目的端口和其mac地址属性（源、目的丢弃、阻塞、优先级、镜像等）

8）保留mac地址表（Reserved MAC Address Table）：包括64个条目，存储保留的mac地址。

9）mac阻塞表（MAC Block Table）：容纳32组。出口处的阻塞、洪泛的源mac地址组。

10）2层多播表（Layer 2 Multicast Table）：容纳1K组。存储2层多播组。

11）链路汇聚组表（Link Aggregation Group Table）：容纳128组。表示链路汇聚组端口成员与hash选择条件的关系。

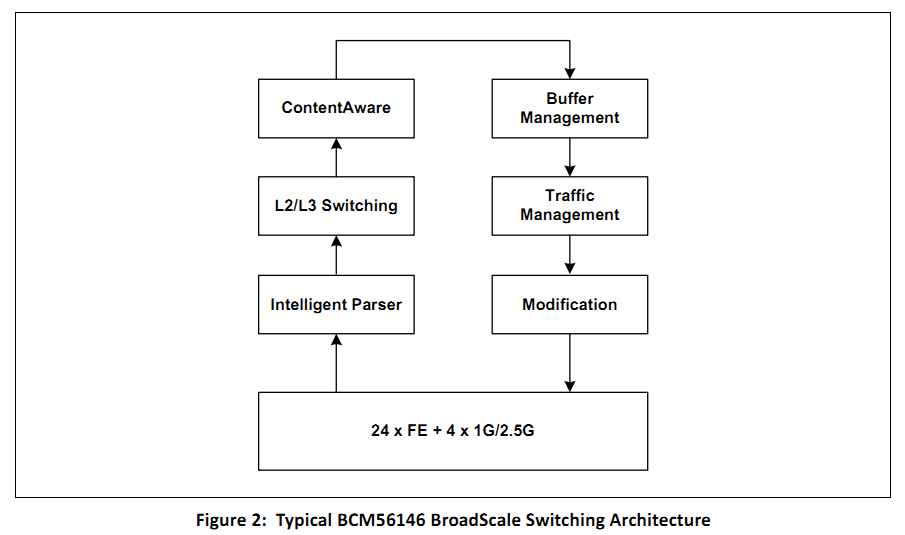
12）基于IP子网的vlan表（IP Subnet-Based VLAN Table）：容纳256个子网。

13）DSCP表（DSCP Table）：容纳1920个条目。重新映射入口、出口DSCP到新的DSCP和优先级

14）3层主机路由表（Layer 3 Host Route Table）：容纳512个IPV4主机，或256个IPV6主机。

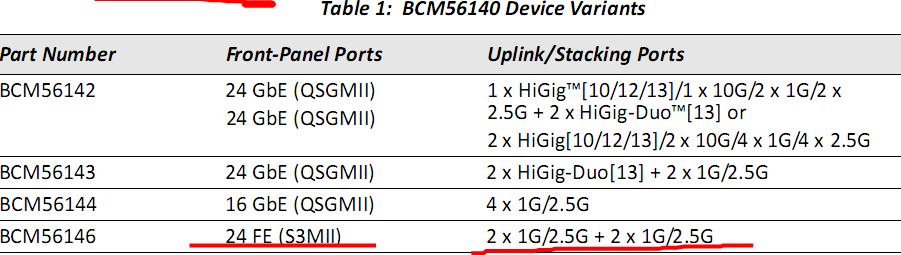
15）3层最长前向匹配路由表（Layer 3 LPM Route Table）：64 IPv4 路由或32个IPv6 路由

16）3层IP组播表（Layer 3 IP Multicast Table）：容纳64组。

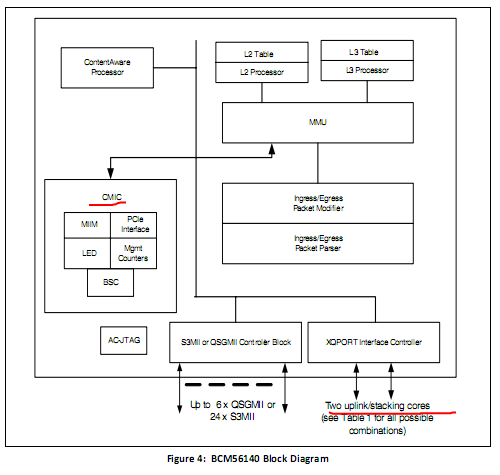
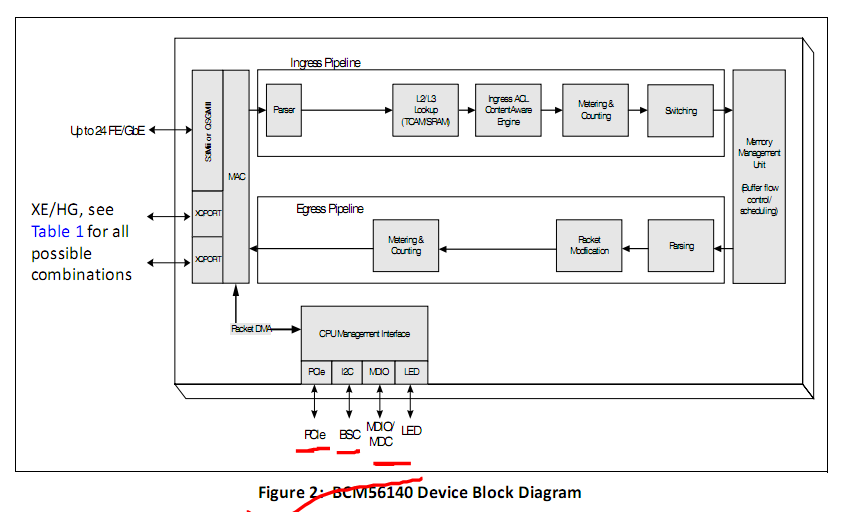
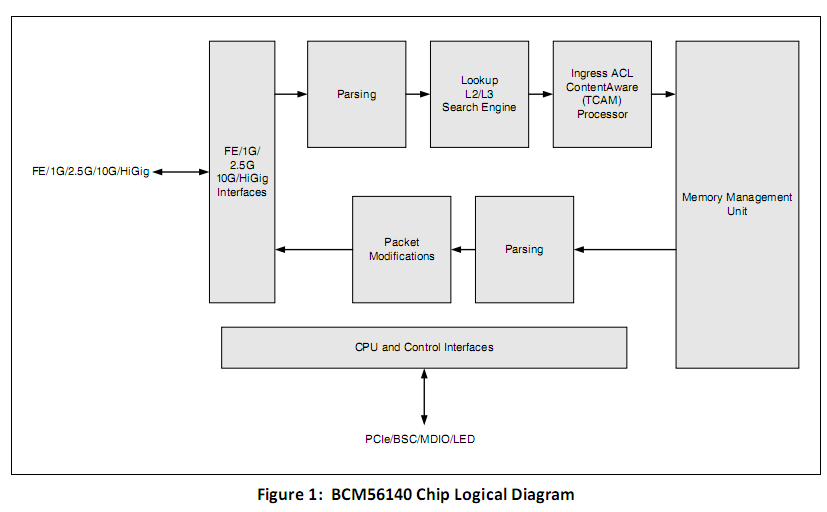


# 56140系列交换芯片功能

56140系列芯片主要包括包括：



他们符合StrataXGS® IV switching family



### Pipeline Blocks

1）Intelligent Parser：包括full parser 和HiGig+/HiGig2 parser。Full parser主要是分析入口处的数据包中前128字节的。Higig parser主要是分析HiGig+/HiGig2 header信息。

2）Security Engine：主要提供Early discard detection、Control Denial-of-Service, DoS, attack detection、Flow-based mirroring、Flow-rate metering。

3）L2 Switching：执行vlan、权限任务，寻找转发包的目的mac地址、源mac地址学习，包括VLAN type select、VLAN look-up、L2 unicast look-up、L2 multicast look-up。

4）L3 Routing：支持3层的ip协议。执行单播、多播数据包的源ip地址、目的ip地址查找。包括：L3 unicast look-up、L3 multicast look-up、Longest prefix match、Look-up switch logic、 Strict and loose uRPF checks。

5）ContentAware Processing：支持快速过滤处理、ACL、differentiated services、QoS。要分析这些内容：MACDA, MACSA, DIP, SIP, TCP, several others等等。

6）Buffer Management：提供cos、带宽保障、带宽限制、metering mechanisms。

7）Modification：数据包的修改可能是由于VLAN Translation、L3 routed packet modification等原因。

### Memory Management Unit

内存系统主要有CBP和transaction queue (XQ)组成，主要是管理cell buffer，设备支持1.5M cell buffer poll（CBP）。对于buffer的管理方式主要是：Ingress backpressure (IBP)、Head-of-line (HOL) prevention、 Ingress rate shaping, PAUSE metering

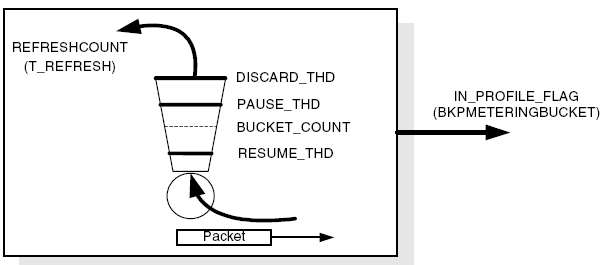
a) IBP主要用于入端口，缓解端口拥塞;尽可能减少数据包的丢失。 在Ingress Ports上有效管理buffer资源; 基于每端口设置IBP产生门限值与IBP消除门限；当到达的数据包数量满足达到门限，发送Pause帧; 当达到的数据包数量低于IBP的消除门限，停发Pause帧;

b) HOL主要用于出端口和cos。

* 当多个端口或多个流向一个端口发送数据包时有可能会产生阻塞，会引起输入端口到其它端口数据包的丢失,此种情况即为HOL(Head-of-Line Blocking，线路头阻塞)。
* 通过设置HOL门限值，当进入的数据包达到该门限,不再允许新的数据包进入Egress队列,在MMU丢弃这些数据包。HOL预防机制通过丢弃数据包达到目的; 与PAUSE Metering, Ingress Backpressure不同。
* 提供两种HOL预防机制：Cell-base。设置每个端口所用memory大小; Packet-based。设置每个端口输出队列的包的最大个数.

c) PAUSE Metering：主要是为入口提供实现入口速率整形。他会追踪每个入端口的带宽，当超过限制时，会发送PAUSE message。

* 每个端口有独立的漏桶，用于限制带宽与流量整形。
* 如下所示,每T\_REFRESH(7.8125 ms)周期从BUCKET中取出REFRESHCOUNT 个token (每token=0.5bit);
* 初始化是漏桶bit值为0(令牌为full)，当进入一个packets 时，相当于往BUCKET注入相应的token,每周期取出REFRESHCOUNT ；当进入的包大于PAUSE\_THD 时，PAUSE帧产生;这就是PAUSE Metering功能，用于IBP机制;



### Search Engines

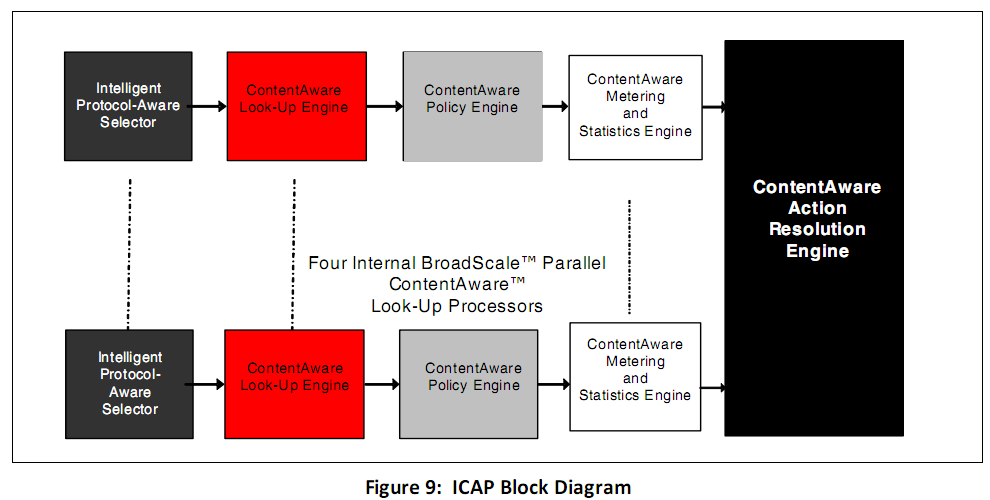
支持Hash search engine、CAM search engine。

Hash search engine (HSE)主要用在L2 MAC table (for look-up and learning), L3 host table (for IPv4 and IPv6 look-ups), MAC VLAN, IPMC searches (s,g) and (\*,g), Ingress VLAN translation, and egress VLAN translation中。

CAM search engine (CSE)主要是在使用ContentAware engine时，被Default routing/policy routing/ingress ACL或HiGig ACL触发。

### ContentAware Processor (CAP)

所谓ContentAware就是对packet的内容进行智能匹配的技术。主要是为ACL, DSCP, QoS等提供支持。



这5个部分分别是智能协议识别选择器、CAM查找引擎、策略引擎、meter和统计引擎、动作裁决引擎。在ingress端口，智能协议识别选择器对进来的包的前128bit按照协议字段进行选择和标记，CAM查找引擎按照用户给的key匹配协议选择器的内容，如果找到了，就执行策略引擎的动作，并可以实验meter和统计引擎进行限速、标记颜色和统计。

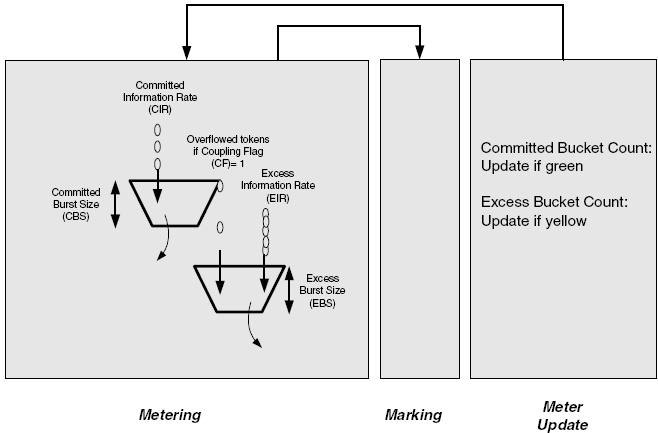
### MEF Policing

Meter用于监控传输带宽。SrTCM、TrTCM、modified TrTCM是meter的三种方案。有待定信息速率(CIR)和额外信息速率（EIR），他们有一定扩展范围，分别是CBS和EBS

- 当进入包数量在CBS之内时，报文标识为Green，可被转发的报文;

- 当进入包数量超出CBS但在EBS之内,超出CBS部分的包标识为YELLOW, YELLOW 包不保证可发,可应用丢弃/转发操作;

- 当进入的包数量超出EBS,超出EBS的包标识为RED,不会得到令牌转发;



Committed Information Rate (CIR)

Committed Burst Size (CBS)

Excess Information Rate (EIR)

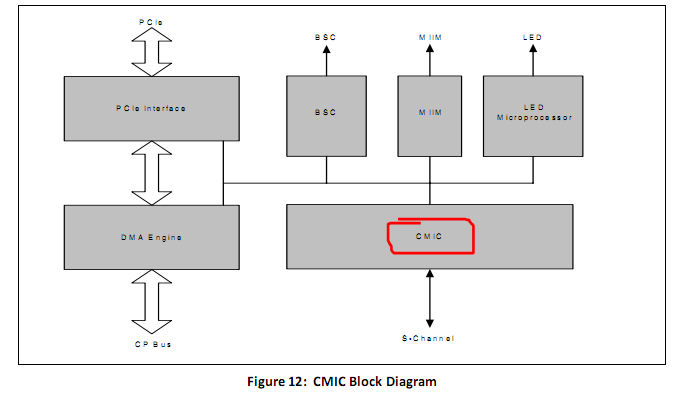
Excess Burst Size (EBS)

### CPU Management Interface Controller (CMIC)

CPU通过PCI接口与芯片的CMIC模块连接，并与芯片交互数据；通过PCIe接口，提供两种交互数据的机制：

- DMA 通道;用于交互大量数据块，比如收发数据包,CPU获取芯片的内存表数据;

- MESSAGING MECHANISM(消息机制)；用于交互小数据块，比如CPU读写芯片寄存器，读写内存表;



### Packet Flow

1）2层入口数据包流

数据包到达入口处时，会被分析，从数据包取出相关的内容，进行CAP处理，然后，判断数据包是否已经加了tag还是没有加tag。对于没有加tag的数据包，需要从VLAN\_XLATE table、 Subnet-based table、 Protocol-based table、 Port-based table (default)中指定vlan id。对于已经加tag的，可以从数据包中获得vlan id。通过vlan查找表，可以知道数据包是否属于这个vlan，如果属于这个vlan，将进行STP、vlan端口bitmap、PFM (port filtering mode)处理，如果这个数据包的vlan是无效的，将丢弃该数据包。

随后，开始了学习阶段。查看该数据的源mac地址和vlan id是否已经在L2 table中，如果有，说明已经学习过了，如果没有，学习的方法取决于CPU Managed Learning (CML)的设置，是通过硬件学习，还是传给cpu，还是丢弃。

下一阶段是目的mac地址查找。如果在L2\_USER\_ENTRY中找到，数据包的目的地址要基于BPDU设置。如果BPDU设置为0，数据包将根据目的module、目的端口进行转发；如果BPDU设置为，数据将被丢弃、或拷贝给cpu、或洪泛到vlan中。如果在L2\_USER\_ENTRY中没有找到，但在L2\_ENTRY table中找到了，将会转发到DST\_MODID 、DST\_PORT/TGID。否则，则洪泛到整个vlan中。

当数据包是多播包时，经过的路径与单播包一样。当多播包在L2\_ENTRY中找到，L2\_ENTRY的索引将用来确定L2MC table的索引。L2MC table用于映射出端口，当PFM为0时，将洪泛到整个vlan中，当PFM为1是，将转发给L2MC table中指定的端口。

对于多播包，经过的路径与单播包一样，只是在学习之后，将报文转发给该vlan的所有成员。

2）3层入口数据包流

如果查目的MAC地址表的时候发现L3bit置位了，就进入到L3转发流程。L3交换可以实现跨VLAN转发，而且它的转发依据不是根据目的MAC地址，而是根据目的IP。L3转发的流程是：

首先对L3头部进行校验，校验和错的包直接丢弃；然后进行原IP地址查找，如果主机路由表中没有找到，会上报给CPU，CPU会进行相应的处理，并更新接口表；

下一步进行目的IP地址查找，如果主机路由表中没有找到，就会在子网路由表中进行查找，在子网路由表中进行最长子网匹配的查找算法，如果在子网路由表中还没有找到，也送给CPU进行处理，如果在主机路由表或子网路由表中找到了，就会得到下一跳的指针。如果ECMP使能的话，会得到ECMP的指针和ECMP的个数，从而根据hash算法得到一个下一跳指针。下一条表项中包含了下一跳的MAC地址和接口表的索引。在包转发出去的时候，用下一跳的MAC地址替换掉包的目的MAC地址。用接口表中的MAC地址和VLAN替换掉包的原MAC地址和VLAN。

### L2 功能介绍

主要包括：

• “Learning” （mac地址学习）

• “L2 Address Aging” （2层地址老化）

• “L2 Address Learning Limits” （2层地址学习限制）

• “L2 Multicast” （2层多播）

• “L2 User Entry” （2层用户登记）

• “L2 Port Bridge” （2层端口桥）

• “Spanning Tree” （生成树）

1. Learning

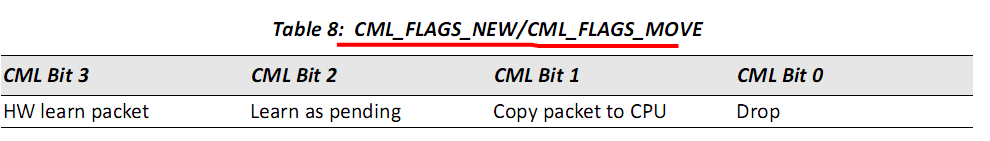
L2\_ENTRY表中包含三种不同的类型：

VLAN-based bridging (basic),

single-VLAN cross connect,

and double-VLAN cross connect.

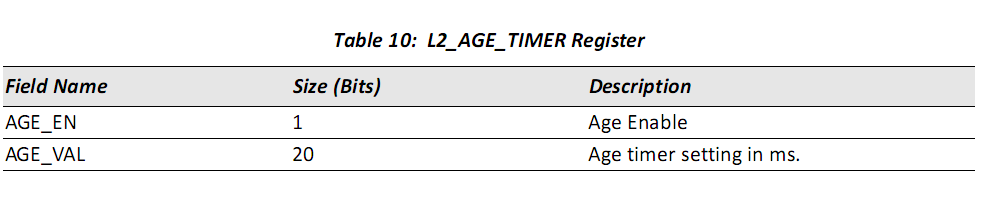
当CPU managed Learning (CML)模式设置时，硬件mac地址学习使能。通过CML\_FLAGS\_NEW和CML\_FLAGS\_MOVE控制CML，CML\_FLAGS\_NEW用来控制数据包中mac地址不认识的情况，CML\_FLAGS\_MOVE用来控制入口的数据包的mac地址与2层表中记录的该端口mac地址不相同的情况。有四种处理方式：HW学习包、pending学习、拷贝给cpu、丢弃。



L2\_MOD\_FIFO有128条记录，用于向cpu通知L2\_ENTRY 表发生了变化，L2\_MOD\_FIFO表中存储了L2\_ENTRY表的改变。

2）L2 Address Aging

通过L2\_AGE\_TIMER寄存器来使能老化，和设置老化时间。



3）L2 Address Learning Limits

* 设置每端口/每LAG，每VLAN与每芯片地址学习限制：

- 每芯片地址学习限制与当前学习的MAC数量统计

- 28个端口,每端口可设学习限制，当前学习的MAC统计；

- 128 个LAG，每LAG可设学习限制，当前学习MAC统计；

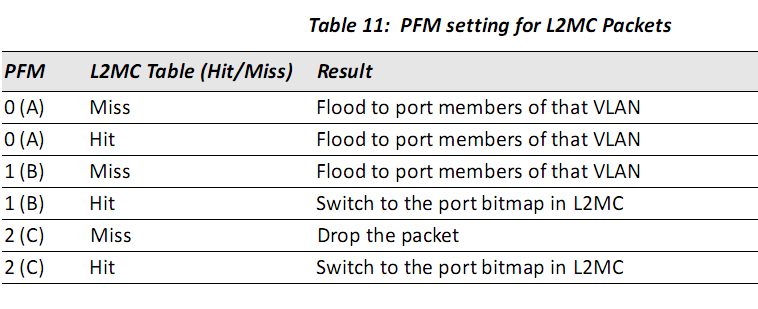
- 4095 VLAN，每VLAN可设学习限制，当前学习MAC统计；

* 支持使能地址限制Check功能：

数据包分别经过system limit，port/LAG limit, VLAN limit逐级检查，如限制门限达到，丢掉该数据包；也可选送CPU；

4）L2 Multicast

* 支持三种PFM模式 (port filtering modes)：
* PFM A, 所有组播包在VLAN域内广播；
* PFM B, 已知组播转发至组播表的pbmp端口;未知组播则在VLAN域内广播;
* PFM C,已知组播转发至组播表的pbmp端口;未知组播则丢弃;

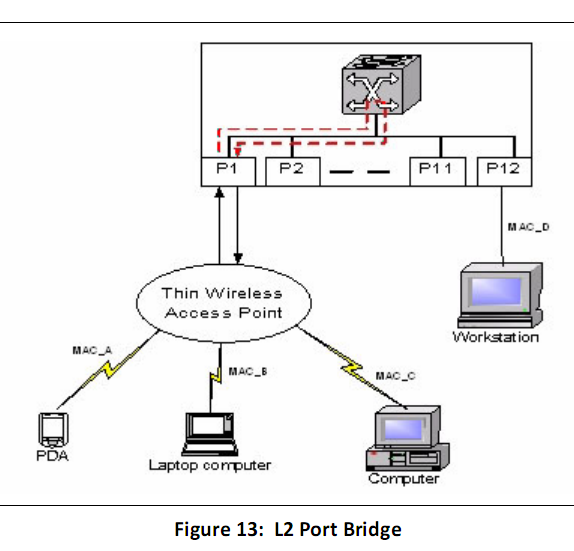


5）L2 User Entry

L2\_USER\_ENTRY表包含64条表项，关键值是MAC\_DA 、VLAN\_ID。

6）L2 Port Bridge

允许收发包是同一个端口。典型的应用是Wireless Access Point (WAP)。同一个端口可以对应多个mac。



7）Spanning Tree

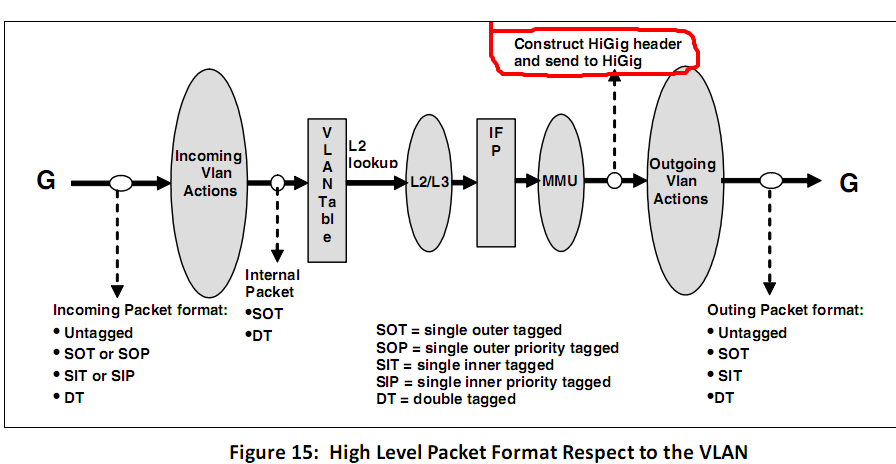
* 完全支持IEEE 802.1D(STP),IEEE 802.1s(MSTP), IEEE 802.1W(RSTP):

- 支持端口状态设置: disable, blocking, listening, learning , forwarding;

- 以上的2层功能，支持生成树必要的条件:学习,老化与MAC地址表的批量删除;

- 支持256 生成树表项。

### VLAN



Internal VLAN ID获取

* 提供并根据以下排列优先顺序的几种表用于获取Internal VLAN ID:
  + VLAN Translation。以数据包的Inner/Outer VID为索引；每端口可选设置时能;
  + MAC-Based。以数据包的源MAC地址为索引；每端口可选设置时能；
  + Subnet-Based。以数据包的源IP为索引；每端口可选设置时能；
  + Protocol-Based。以数据包的协议类型( ETHNET\_TYPE )为索引；每端口可选设置时能；
  + Port-Based。以端口配置的默认VLANID作为Internal VLAN；



* 对于Untagged与single priority-tagged packet ，查找的类型表以及优先顺序为MAC-Based - > Subnet-Based -> Protocol-Based ->Port-Base;
* 对tagged packet,查找顺序为 VLAN Translation -> MAC-Based - > Subnet-Based -> Protocol-Based ->Port-Base;

Ingress VLAN Action

* 查表结果，对数据包执行以下的VID转换规则：

- Add Internal OVID/IVID

- Replace incoming OVID/IVID with internal OVID/IVID

- Delete incoming OVID/IVID

- Do not modify

使用VLAN\_XLATE表

Egress VLAN Action

* 在出口端，根据OutVid/InnerVid 查找Egress VLAN Translation, 执行以下相应操作(根据untagged与tagged包类型):

- Add OVID/IVID

- Replace OVID/IVID with internal/outgoing OVID/IVID

- Delete OVID/IVID

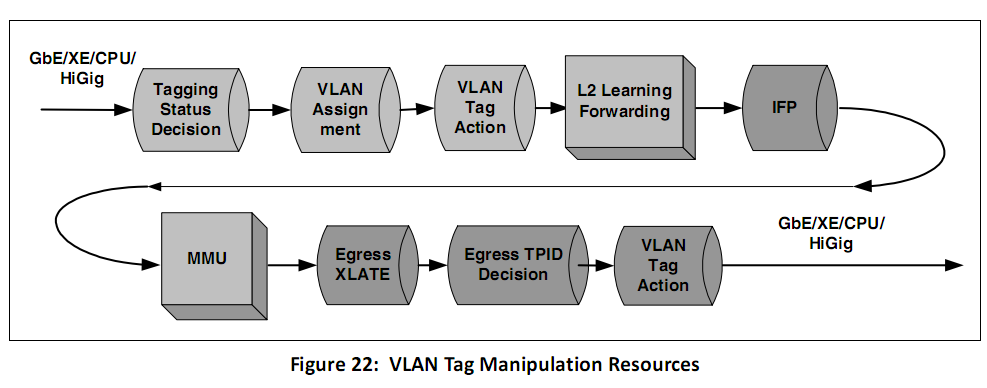
- Do not modify

* 通过EGR\_VLAN表配置，对数据包进行强制剥离外层VID;用于Double-tagged模式;

**VLAN Range Checking**

* 支持128个VLAN Range表。通过Range表的配置，将某段范围的userVID替换成一个OVID，节省VLAN表与其他资源VLAN Transparent表项的资源；
* VLAN Range执行优于Ingress VLAN Translation;

VLAN tag操作流程



转：http://blog.chinaunix.net/uid-11140746-id-3712645.html