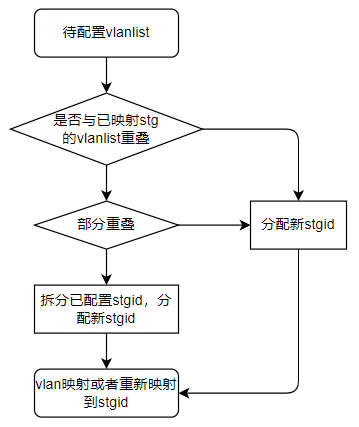
**stg分配逻辑**

stg的分配逻辑是每条命令行配置下发的vlanlist需要映射到不同的stg上，如果前后两条命令行配置的vlanlist有重叠部分，那么对应的需要将这两条命令行的vlanlist映射的stg进行拆分处理。

比如vlan 1-100命令行配置映射到stg 1，再下发vlan 50-150的配置时，因为有vlan 50-100的重叠，所以要根据vlan 1-49，50-100，101-150分段，分别映射到不同的stg，删除其中一条配置时将多余的stg释放，这时候又需要将stg资源进行合并。驱动层代码的功能就是实现stg的这个分配逻辑。

检查vlanlist，分配stgid的逻辑框图如下：



分配好硬件stgid，并将vlanlist映射到各自的stgid后，再设置接口在各个stgid中的状态。

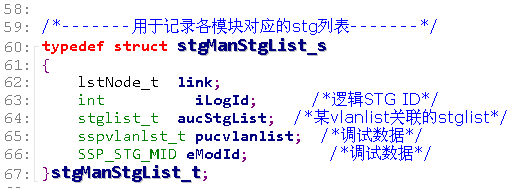
之后保护倒换时就只需要根据端口设置端口在各个stg中的状态即可。比如，端口1/1/2在stg 1和2中设置成block，但是在stg 3-127中仍然保持forwarding状态。

**主要数据结构**

**m\_StgManStg**



链表类型，保存的数据结构类型：



主要用来保存每一个配置命令使用到的逻辑stgid，实际分配的硬件stgid，涉及到的vlan，以及该该配置是哪个模块下发的。

各成员变量：

**iLogId**

用来保存软件配置的逻辑stgid，与实际下发芯片使用的硬件stgid不同，主要是记录配置使用的逻辑id，比如命令行配置两条erps环命令，使用的是两个不同的逻辑id，但是可能会因为有重复的vlan（第一条命令是vlan 1-100，第二条是vlan 50-150），在实际下发到芯片时会做拆分处理，出现芯片实际使用了三个stg id，vlan 1-49使用一个stg，vlan 50-100使用一个stg,vlan 101-150使用一个stg。

**aucStgList**

数组类型，使用bit位保存stgid，数组长度64，总共512个bit：



配置使用到的所有硬件stg id，比如第一条命令的vlanlist是1-100，在全局变量m\_StgManStg中新增一个链表节点，分配的硬件stgid为1，那么这个变量aucStgList的bit1置位；第二条命令使用到的vlanlist是50-150，与第一条命令的vlanlist有重叠，那么经过拆分，分配到两个新的硬件stgid为2和3，那么在全局变量m\_StgManStg中新增一个链表节点，变量aucStgList的bit2和bit3置位。

**pucvlanlist**

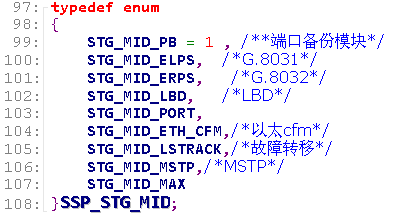
数组类型，使用bit位保存涉及到的vlan：



比如创建一条erps环配置时，需要block vlan 1-100，传入的vlanlist参数也是通过数组的bit位标识，然后处理保存到本变量。

**eModId**

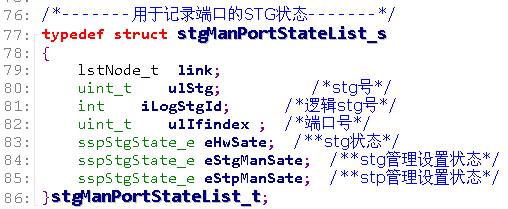
使用本stg的模块，是erps环网配置，还是stp协议的配置等：



**m\_StgManPortState**



链表类型，保存的数据结构：



主要用来保存接口对应的stgid，以及在该stg中的状态是转发还是丢弃。

**ulStg**

硬件stgid，实际下发到芯片的。

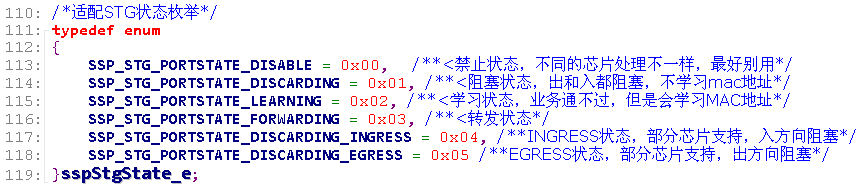
**iLogStgId**

逻辑stgid，用来保存软件配置，与实际下发芯片使用的硬件stgid不同，主要是记录配置使用的逻辑id，参考m\_StgManStg中的iLogId说明。

**eHwSate**

下发的端口硬件状态。

**eStgManSate/eStpManSate**



软件保存的端口状态，在设置端口状态时会用来做一些检查，一般在函数\_stg\_port\_state\_set中用到。

**m\_StgManPortVlanListMode**



链表类型，保存的数据结构：



主要用来保存同一个端口下，相同vlanlist使用到的stgid，避免不同配置产生冲突，互相影响。例：

同一个端口下，vlan 1-100分配了硬件stgid 1，状态为forwarding，那么该端口下需要再配置一条vlan 50-150时，会通过这个变量进行检查，vlan有重叠时返回错误。

如果不检查该条件，那么vlan 50-100的vlan可能会分配新的硬件stgid，并且设置端口在这个新的stgid上是discarding状态，导致第一条命令的vlan 50-100出现异常，原本转发的报文会突然被丢弃。

一般都是在\*\*\*\_take和\*\*\*\_give函数中使用。

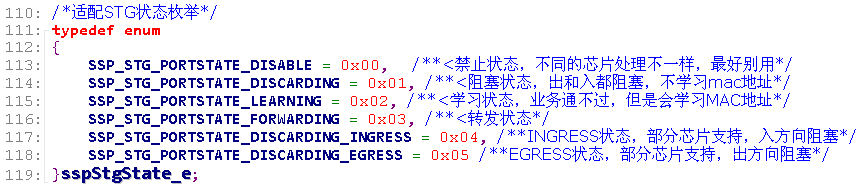
**pucvlanlist**

数组类型，使用bit位保存涉及到的vlan：



比如创建一条erps环配置时，需要block vlan 1-100，传入的vlanlist参数也是通过数组的bit位标识，然后处理保存到本变量。

**eStpManSate**

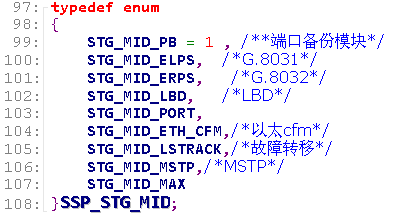


**ulStgTakeCnt/ulStpTakeCnt**

同一个模块使用同一个端口的次数，比如stp协议多次使用到同一个端口。注意如果是stg类型，一般计数只有一次，因为不能有vlan重叠，而同一个端口下不重叠的vlan会在本全局变量m\_StgManPortVlanListMode链表中新增一个节点。

**eModId/eStpModId**

使用本全局变量的模块，是erps环网配置，还是stp协议的配置等。如果是stg类型，那么eStpModId设置为0，如果是stp类型，那么eModId设置为0。



**m\_aucStgManStgUsed**



无符号字符类型，双指针全局变量，一层指针用来区分对应的芯片id，ctc设备一般最大支持两个芯片，第二层指针用来表示对应的硬件stgid（注意不是软件记录的stgid），初始化为512个stgid空间。

通过函数\_stg\_stg\_hw\_create申请分配硬件stgid，申请成功后保存到该全局变量。

通过函数\_stg\_stg\_hw\_del释放硬件stgid，该全局变量对应的数组下标置零。

**m\_aucStgManVlanToStg**



整型类型，双指针全局变量，一层指针用来区分对应的芯片id，ctc设备一般最大支持两个芯片，第二层指针用来表示vlan所在的硬件stgid（注意不是软件记录的stgid），初始化为4094个vlan空间。

初始化vlan对应的stgid为0，后续创建vlan时会设置vlan所在stgid为1，所以平台下发配置通过\_stg\_stg\_vlan\_hw\_set分配的硬件stgid保存到本全局变量，在删除配置时会将本全局变量置为1。

**m\_ulStgManStgUsed**



每次通过函数\_stg\_stg\_hw\_create分配硬件stgid后该全局变量加1，通过函数\_stg\_stg\_hw\_del释放硬件stgid时该全局变量减1。

主要作用是在分配硬件stgid时检查硬件资源是否够用。

**m\_aucStgManStgRef**



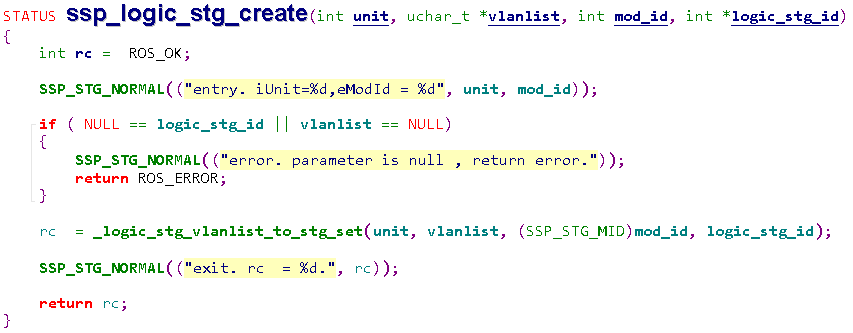
无符号字符类型，双指针全局变量，一层指针用来区分对应的芯片id，ctc设备一般最大支持两个芯片，第二层指针用来表示每个stg被复用的次数。

在删除命令释放一个stgid时，将其他与本stgid有重叠的配置合并到其他stgid。比如，两条配置，vlan 1-100配置时分配了硬件stgid 1，再配置vlan 50-150时，会将vlan 1-49拆分到stgid 1，vlan 50-100拆分到stgid 2，vlan 101-150拆分到stgid 3，然后删除第二条vlan 50-150的配置时，会删除stgid 2和3，将拆分到stgid 2的vlan 50-100合并到原先的stgid 1，这样恢复到只配第一条命令vlan 1-100时分配的stgid 1。

合并的操作主要在\_logic\_stg\_vlanlist\_to\_stg\_del函数中处理。相关全局变量m\_StgManStg。

**主要函数**

**分配stgid**



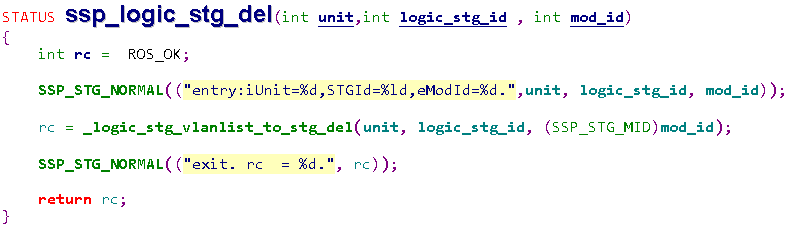
所有需要使用stg功能的模块都通过这个对外接口获取stgid。可以看到stg的分配只与vlan相关，mod\_id对应的是使用stg的模块id；logic\_stg\_id则是软件的stg索引，与实际下发芯片的硬件stgid不同，软件索引用来标识对应的命令行配置。比如创建两条ERPS环网配置，软件索引一定是不同的，但是硬件stgid则有可能相同。

其中调用的函数



会根据全局变量m\_aucStgManVlanToStg和需要的新stgid个数来判断是否需要进行拆分。

**删除stgid**

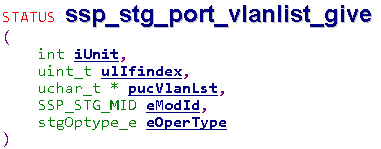


**port+vlan占用标识**



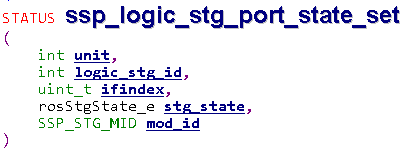
得到新的stgid后，要通过本函数设置软件变量，来标识指定端口和vlan已经被某模块使用，无法再进行配置。如果检查到已被占用，则需要通过ssp\_logic\_stg\_del删除已经创建的stgid。

**删除port+vlan占用标识**



命令行删除配置，释放stgid时清除软件的端口加vlan占用标识，下次配置时可以重新使用。

**设置端口状态**



设置端口在stg中的状态，是forwarding或者discarding。

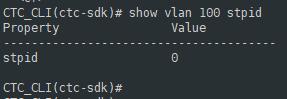
通过软件的stg索引logic\_stg\_id在保存的全局变量m\_StgManStg中找到保存的硬件stgid，然后设置端口在这些stgid中的状态。

**sdk相关**

**sdk相关命令**

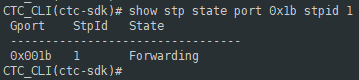
**查看vlan所在stp实例**

show vlan VLAN\_ID stpid



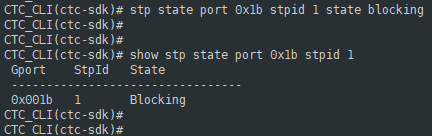
**查看接口stp状态**

show stp state port GPHYPORT\_ID stpid STP\_ID



**设置接口stp状态**

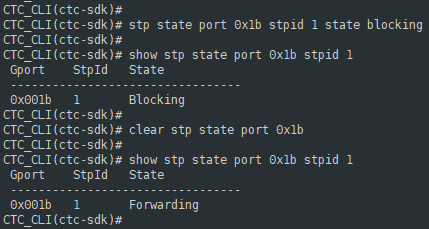
stp state port GPHYPORT\_ID stpid STP\_ID state (forwarding | blocking | learning)



设置接口在指定STP\_ID中的状态是forwarding，blocking或者learning。

**清除接口stp状态**

clear stp state port GPHYPORT\_ID



命令设置接口为默认forwarding状态。