



# 盛科 CTC8180 机架优化方案介绍

版本 R1.0  
日期 2020-12-04

版权所有 © 盛科网络（苏州）有限公司。保留一切权利。

未经盛科网络（苏州）有限公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式和任何方法传播。



盛科商标，服务标志和其他盛科标志均为盛科网络（苏州）有限公司拥有商标。盛科交换机系列产品和芯片系列产品的标志均为盛科网络（苏州）有限公司商标或注册商标。未经盛科书面授权，不允许使用这些标志。

本文档提及的其他所有商标和商业名称，由各自的所有人拥有。

## 注意

您购买的产品、服务或特性等应受盛科网络商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，本公司对本文档内容不做任何明示或默示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

## 盛科网络（苏州）有限公司

地址 江苏省苏州市工业园区星汉街 5 号（腾飞新苏工业坊）B 幢 4 楼 13/16 单元

电话 86-512-62885358

传真 86-512-62885870

网址 <http://www.centecnetworks.com>

邮箱 [support@centecnetworks.com](mailto:support@centecnetworks.com)

# 内容目录

1 简介.....	5
1.1 适用对象.....	5
1.2 缩略词.....	5
1.3 参考.....	5
2 概要介绍.....	6
2.1 主要改动介绍.....	6
3 详细介绍.....	7
3.1 CFlexHeader 结构扩展.....	7
3.1.1 功能描述.....	7
3.1.2 CFlexHeaderBasic.....	8
3.1.3 CFHeaderExtEgrEdit.....	8
3.1.4 CFHeaderExtCid.....	8
3.1.5 CFHeaderExtLearning.....	9
3.1.6 CFHeaderExtOam.....	9
3.2 组播流量 stacking trunk Port 选路.....	9
3.2.1 组播流量 stacking trunk Port 选路实现.....	9
3.2.2 组播流量 stacking trunk Port 选路限制.....	10
3.3 CFlex Port 备份端口.....	10
3.3.1 CFlex Port 备份端口的实现.....	10
3.3.2 CFlex Port 备份端口的限制.....	11
4 附录.....	12
4.1 CFlexHeader 格式.....	12
4.1.1 CFHeaderBasic 格式.....	12
4.1.2 CFHeaderExtCid 格式.....	13
4.1.3 CFHeaderExtEgrEdit 格式.....	14
4.1.4 CFHeaderExtOam 格式.....	15
4.1.5 CFHeaderExtLearning 格式.....	16

## 图形目录

图 3-1 CFlexHeader .....	8
图 3-2 CTC8180 CFlex Trunk 示意图.....	10

# 1 简介

本文档在《盛科交换芯片机架方案介绍》文档基础上，介绍盛科 CTC8180 相对 CTC7148/7132 芯片在分布式系统的转发上新增的一些特性，或者与其它芯片实现有差异的地方。如果需要了解详细的分布式转发原理和实现，请参考《盛科交换芯片机架方案介绍》。

## 1.1 适用对象

对使用盛科 TransWarp™ 系列交换芯片搭建机架系统感兴趣的用户。

## 1.2 缩略词

缩略词	定义
LC/LineCard	线卡，用于提供业务接口，也叫做业务卡，或者接口卡
GB	盛科 GreatBelt 系列交换芯片
GG	盛科 GoldenGate 系列交换芯片
D2	盛科 Duet2 系列交换芯片，或者称为 CTC7148
TM	CTC7132
TM.MX	CTC8180

## 1.3 参考

无

# 2 概要介绍

## 2.1 主要改动介绍

- CFlexHeader 结构扩展
- 组播流量 trunk port 选路方法
- CFlex Port 备份端口

# 3 详细介绍

## 3.1 CFlexHeader 结构扩展

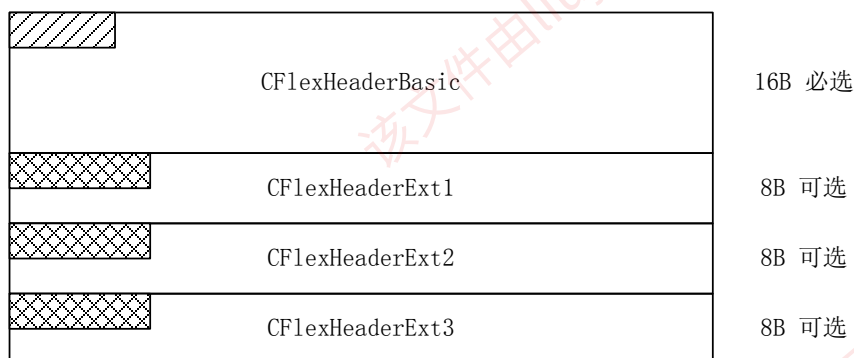
### 3.1.1 功能描述


从 CTC7132 开始，在保留原有 CFlexHeader 结构基础上，新增一种全新的 CFlexHeader 结构。新增的 CFlexHeader 结构有两部分组成。

一部分是固定的 16 字节 CFlexHeaderBasic，这部分是必选 CFlexHeader，通常应用只要这个头部就足够了。

另外一部分是有多种类型扩展头部组成，每种类型的扩展头部支持一种或者一部分业务特性。每个扩展头长度为 8 字节。扩展头部不是必选，可以根据业务按需携带到 CFlexHeaderBasic 后面。目前支持四种类型的扩展头部，分别是：

- CFlexHeaderExtEgrEdit: egressEdit 所需信息
- CFlexHeaderExtLearning: macSa 学习所需信息
- CFlexHeaderExtOam: oam 所需信息
- CFlexHeaderExtCid: 其他业务所需信息



 extHeaderLen[2:0], 扩展头长度, 以8字节为单位


 extHeaderType[3:0], 扩展头类型

图3-1 CFlexHeader

上图为 CFlexHeader 结构示意图。一个完整的 CFlexHeader。为了减小开销, CFHeaderExtLearning 和 CFHeaderExtOam 两种扩展头部不会同时携带, 一旦需要携带 CFHeaderExtOam, 则自动忽略 CFHeaderExtLearning。

CTC8180 相对于 CTC7132 对 CFlex header 的改进相对较小, 主要是在原有的字段上进行了添加或者扩展字段宽度。

### 3.1.2 CFlexHeaderBasic

CFlexHeaderBasic 总共 16 字节, 为必选字段。增加了以下几个字段

- macSalookupEn, 控制在 CFlex macsa 查找, 和原有的 learning 解耦, 便于已经学习过的报文刷新老化标记位。
- fidHigh, fid 的位宽增加了一个 bit, 由原来的 14bits 变成了 15 bits
- SrcVlanPtrHigh, 位宽增加了一个 bit, 由原来的 14bits 变成了 15 bits

### 3.1.3 CFHeaderExtEgrEdit

CFHeaderExtEgrEdit 是可选扩展头部, 总共 8 字节, 如果报文做 egress edit, 需要携带这个头部。增加了以下字段:

- aclDscpValid, 在 Egress chip 编辑模式下, 控制 ACL 修改 DSCP 在出口芯片不被 nexthop 的编辑行为覆盖。

### 3.1.4 CFHeaderExtCid

CFHeaderExtCid 是可选扩展头部, 总共 8 字节, 携带 categoryId, stacking 拓扑发现, 报文截断等相关的属性。增加了以下字段:

- l2eSrcCidHigh[7:0], cid 位宽扩展, 由原有的 8bits 变成 16bits, 表示 cid[15:8],
- queMapType[2:0], 控制报文到远端出口芯片入队列的方法,
- serviceId[9:0], 用于在出口芯片控制通过 serviceid 入队列。



### 3.1.5 CFHeaderExtLearning

CFHeaderExtLearning 是可选扩展头部， 总共 8 字节， 携带 MAC learning 相关的属性。字段没有变化。

### 3.1.6 CFHeaderExtOam

CFHeaderExtOam 是可选扩展头部， 总共 8 字节， 这里面包括一些 OAM 相关的属性。增加了以下字段：

- microbfdMacDaValid[3:0],用于控制 MicroBFD 在非 up 状态下发送的 MACDA 为特定的组播 MACDA 地址。

## 3.2 组播流量 stacking trunk Port 选路

### 3.2.1 组播流量 stacking trunk Port 选路实现

原有 CTC7132 在处理去远端 chip 流量的时候，单播会根据 destchip 得到 trunkId，组播会从组播表 DsMet 中得到 trunkId。得到 trunkId 之后，会使用 trunkId 索引 trunk group 表 DsLinkAggregateChannelGroup，trunk group 表中包含成员端口表 DsLinkAggregateChannelMember 的 base 和 number。

最好通过报文的 CFlexHash 来选择成员表 DsLinkAggregateChannelMember = base + Hash%number。

DsLinkAggregateChannelMember 中的内容为相应的出口 trunk port。

在 CTC8180 上，单播报文的处理逻辑和之前相同。对于组播报文 trunk port 的选路方法和之前不同，具体的实现方案如下：

1.组播表中 DsMet 中得到 trunkId，使用 trunk id 索引 trunk port 的 bitmap 表 DsMetNonUcCflexMemberBitmap，DsMetNonUcCflexMemberBitmap 表中的成员 portBitmap[255:0]表示 trunk group 中的端口。

2.CFlexHash 用来索引 hash 值对应的 mask 表项 DsMetNonUcLagBlockMask；通过 MetFifoCtl.mcastToCflexIndexSelMode 控制支持的 mask 表的模式，

- 0：全局支持一个 trunk 组，每个组支持所有 hash 值选 mask bit
- 1：全局支持 64 个 Cflex trunk 组，每个组支持 16 个 hash 选 mask bit
- 2：全局支持 32 个 Cflex trunk 组，每个组支持 32 个 hash 值选 mask bit

3. 两个 bitmap 运算获取到第一个成员端口。

Portbitmap[255:0] = DsMetNonUcCflexMemberBitmap.bitmap[255:0] & (~DsMetNonUcLagBlockMask.blockPbmMask[255:0])

### 3.2.2 组播流量 stacking trunk Port 选路限制

由于组播 LAG trunk 选路的变化，带来了以下影响：

- 1.组播流量到 Cflex trunk 不支持 DLB，RR 等模式
- 2.Cflex hash 在用来索引 mask 表的时候， Cflex hash 的有效值被截短，可能导致组播的 load balance 不太均衡。

## 3.3 CFlex Port 备份端口

### 3.3.1 CFlex Port 备份端口的实现

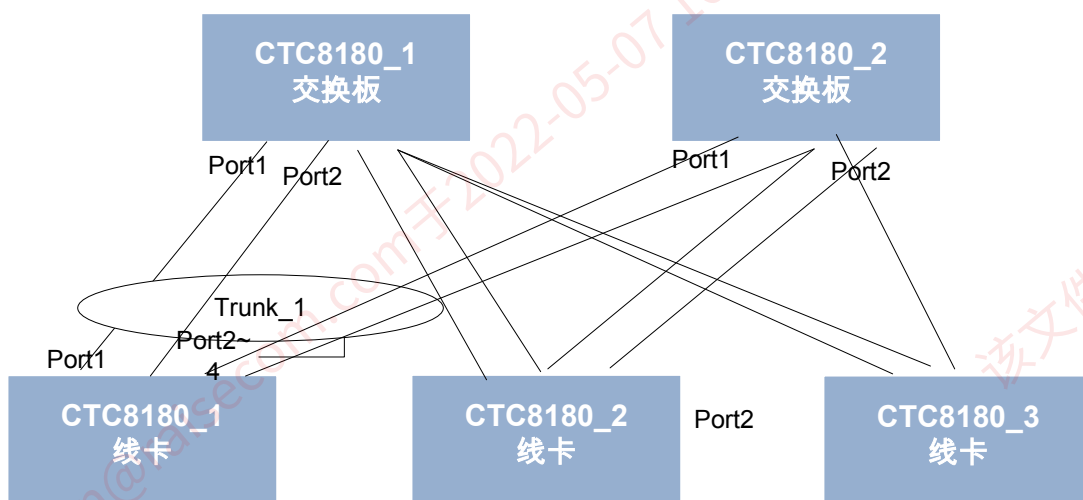


图3-2 CTC8180 CFlex Trunk 示意图

在机架方案中，线卡和交换板在逻辑上通过一个 Cflex trunk 连接，在 Cflex trunk 内线卡会连接到不同的交换板，当交换网板被拔掉的时候，流量需要从断开的物理端口切换到 trunk 内的其他端口，如图 3-2 中线卡 1 上 trunk 1 中 Port1/2 对应的端口交换板 CTC8180 拔掉的情况下，线卡 1 上的端口 1，2 的流量需要切换到 Port3，4。为了快速切换需要为每个端口配置备份端口，例如端口 1 的备份端口为 3，4；端口 2 的备份端口为 3，4。

在 CTC8180 上可是实现硬件检查并切换到备份端口的功能。

对于单播流量的处理流程如下：

当报文根据 destchip 选择到 Cflex trunk 之后，获取  
DsLinkAggregateChannelGroup.IshEn, channelLinkAggMemBase,  
channelLagMemNum;

同时会根据 Cflex hash 选择 DsLinkAggregateChannelMember = base + hash%num

在 DsLinkAggregateChannelMember 中的 channelId 是 trunk 的目的端口，在 LshEn 的情况下，会用 channelId 索引端口 link 状态表 LagEngineLinkState，和端口备份表 CflexLagLinkSelfHealingSet。

当 LagEngineLinkState.linkState 为 1 的时候，表示端口 down，

$LshSetMemberSel = CFlexhash[3:0] \% (CflexLagLinkSelfHealingSet.setSize[2:0] + 1)$

用 LshSetMemberSel 重新选择

CflexLagLinkSelfHealingSet.gMember[LshSetMemberSel] 得到新的 destchannel。

对于组播流量处理的流程相似：

在根据 3.2.1 章节中 Cflex trunk 中组播选路之后，使用选路之后的出口 trunk port 去索引 DsMetPortLagLinkSelfHealingSet，已经端口 link 状态表 MetFifoLinkStatus，

当 DsMetPortLagLinkSelfHealingSet 中 nonUcastToLagLinkSelfHealingEn 使能并且 MetFifoLinkStatus.linkStats 为 1（端口 down）的情况下，

$LshSetMemberSel = CFlexhash[3:0] \% (DsMetPortLagLinkSelfHealingSet.setSize[2:0] + 1)$

用 LshSetMemberSel 重新选择

DsMetPortLagLinkSelfHealingSet.gMember[LshSetMemberSel] 得到新的 destchannel

### 3.3.2 CFlex Port 备份端口的限制

- 1.每个 CFlex Port 的备份端口支持最大为 8.
- 2.在端口 link 恢复之后，硬件无法加回原有端口

# 4 附录

## 4.1 CFlexHeader 格式

### 4.1.1 CFHeaderBasic 格式

Layout	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	fid								srcVlanPtr								tbd															
4									logicSrcPort								sourcePortIsolateld				fid											
8									destMap												headerHash											
c									sourcePort												prio											

Field Name	Offset	Bits	Description
tbd	0	4:0	保留字段
fromCpu	0	5	CPU 下发的报文标志
isDebuggedPkt	0	6	Debugged 报文标志
macLearningEn	0	7	表示是否需要做 MACSA 的查找
srcVlanPtr	0	20:8	用于出口 vlan 编辑
operationType	0	23:21	operationType
fid	0 4	31:24 5:0	用于 MAC 地址学习
sourcePortIsolateld	4	12:6	源端口的端口隔离组号
fromCpuOrOam	4	13	CPU 或者 OAM 下发的报文标志
logicSrcPort	4	29:14	逻辑源端口号
headerHash	4 8	31:30 5:0	Hash 值用于端口聚合选择成员端口
bridgeOperation	8	6	二层操作标志
macKnown	8	7	表示在 Ingress 芯片二层查找时是否匹配到 VLAN 默认转发条目，1 表示匹配了具体的二层转发表项，0 表示匹配到了 VLAN 默认转发条目。对于非二层转发的报文，macKnown 默认置 1

destMap	8	29:8	表示目的地信息。如果目的地是组播, 包括 {isMcast, 5' b0, destId[15:0]}, isMcast 置 1, destId[15:0]表示组播组 Id; 如果目的地是单播, {isMcast, 4' b0, isToCpu, destChipId[6:0], destId[8:0]}, isMcast 置 0, isToCpu 表示该报文是否是送到 CPU 的, destChipId[6:0]表示目的芯片号, destId[8:0]表示单播目的端口号
packetType	8 c	31:30 0	报文类型
color	c	2:1	报文颜色
prio	c	6:3	报文转发优先级
fromLag	c	7	表示报文来自 LAG 口
sourcePort	c	23:8	报文进入系统的原始的源端口号 globalSrcPort
outerVlanIsCVlan	c	24	用于指示 egress chip 如何解些携带两层 Tag 的报文。如果 outerVlanIsCVlan 置 1, 则外层 Tag 会被解析成 C-Tag
svlanTpidIndex	c	26:25	SVLAN 的 TPID 索引值
outerVlanOperType	c	27	是否使能 Dot1Q 操作
extHeaderLen	c	30:28	扩展头长度(不含必选头部长度), 以 8 字节为单位。比如后续有 2 个扩展头, extHeaderLen 为 2
bypassAll	c	31	Bypass 所有操作

### 4.1.2 CFHeaderExtCid 格式

Layout	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
0					portolateType																	i2eSrcCid											
4	exHeaderType				tbd																												

Field Name	Offset	Bits	Description
cnAction	0	1:0	传递 ecn mark 处理逻辑
terminateCidHdr	0	2	置 1 表示出方向无需携带 categoryId Tag
pktWithCidHeader	0	3	置 1 表示原始报文中携带这 categoryId Tag
i2eSrcCid	0	11:4	源 categoryId
i2eSrcCidValid	0	12	置 1 表示源 categoryId 字段有效
pbbCheckDiscard	0	13	pbbCheckDiscard

sourcePortExtender	0	14	端口扩展
portMacSaEn	0	15	使能报文携带端口上的 mac
truncateLenProflId	0	19:16	DsTruncationProfile 表的索引，用于对报文做截断
criticalPacket	0	20	critical 报文标志
isSpanPkt	0	21	表示报文是否是被镜像出来的
isLeaf	0	22	用于水平分割检查
logicPortType	0	23	logicPort 类型
bypassCFlexSrcCheck	0	24	置 1 表示在 Stacking 环上转发的时候，该报文需要跳过防环检测。通常只有 CPU 下发的报文才可能会被置上这个 bit
portIsolateType	0	27:25	portIsolate 类型
ptpApplyEgressAsymmetryDelay	0	28	ptp egressAsymmetry 补偿
isCFlexUpdateResidenceTime	0	29	ptp 更新驻留时间
c2cCheckDisable	0	30	破坏时，是否对 c2c 报文做检查
neighborDiscovery	0	31	用于 stacking 拓扑发现。通常只有 CPU 下发的报文才可能会被置上这个 bit，普通数据业务跨芯片转发的时候不会置这个 bit
noDot1AeEncrypt	4	0	不做 Dot1Ae 编码
tbd	4	27:1	保留字段
extHeaderType		31:28	扩展头部类型，CFHeaderExtEgrEdit 扩展头类型为 2

### 4.1.3 CFHeaderExtEgrEdit 格式

Layout	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	nextHopPtr														srcDscp				ecmpHash													
4	extHeaderType		tbd																ttl													

Field Name	Offset	Bits	Description
ecmpHash	0	7:0	用于新头部编辑的时候，映射 VxLAN 的 UDP source port, 或者 NvGRE 的 GRE key
srcDscp	0	13:8	进来报文的 DSCP 值
nextHopPtr	0	31:14	用于读取 egress chip 上 DsNextHop 表
ttl	4	7:0	原始报文的 TTL 值



egressEditEn	4	8	如果置 1，表示 egress edit，报文编辑将在出接口所在芯片上进行
tbd	4	27:9	保留字段
extHeaderType	4	31:28	扩展头部类型，CFHeaderExtEgrEdit 扩展头类型为 1

#### 4.1.4 CFHeaderExtOam 格式

Layout	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	localPhyPort										mepIndex										oamPacketOffset											
4	extHeaderType								oamType																							
	oamType										dmOffset										tbd											

Field Name	Offset	Bits	Description
oamPacketOffset	0	7:0	报文头部偏移量，用于 OAM 引擎解析 OAM 报文
mepIndex	0	21:8	用于读取 MEP 属性的索引
localPhyPort	0	30:22	入端口号
tbd	0 4	31 5:0	保留字段
dmOffset	4	13:6	DM 头偏移
mipEn	4	14	使能 MIP
dmEn	4	15	使能 DM
useOamTtl	4	16	编辑使用 OAM 引擎携带的 TTL
galExist	4	17	OAM 引擎发送的报文是否包含 GAL
linkOam	4	18	Link OAM
isUp	4	19	CFM UP MEP
oamType	4	23:20	OAM 报文类型： ETH_OAM : 1 IP_BFD : 2 MPLS_OAM: 6 MPLS_BFD: 7 ACH_OAM: 8
rxOam	4	24	表示需要上送到 OAM 引擎处理
oamTunnelEn	4	25	透过 OAM 识别上送到 OAM 引擎，正常转发
fromCpuLmUpDisable	4	26	CPU 下发报文是否需要被 UP MEP LM 统计
fromCpuLmDownDisable	4	27	CPU 下发报文是否需要被 Down MEP LM 统计

extHeaderType	4	31:28	扩展头部类型, CFHeaderExtEgrEdit 扩展头类型为 4
---------------	---	-------	-------------------------------------

#### 4.1.5 CFHeaderExtLearning 格式

Layout	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	macAddr																															
4	extHeaderType		tbd														macAddr															

Field Name	Offset	Bits	Description
macAddr	0 4	31:0 15:0	用于 MAC 地址学习
tbd	4	27:16	保留字段
extHeaderType	4	31:28	扩展头部类型, CFHeaderExtEgrEdit 扩展头类型为 3