INTERNATIONAL TELECOMMUNICATION UNION

TELECOMMUNICATION STANDARDIZATION SECTOR

STUDY PERIOD 2022-2024

SG15-TD181/WP3

STUDY GROUP 15

Original: English

Question(s): 10/15 — Geneva, 17 - 28 April 2023

TD

Source: Editor G.8121/Y.1381

Title: Draft Amendment 1 to Recommendation ITU-T G.8121/Y.1381

Contact: Yuji Tochio
Fujitsu
Japan

Tel: +81 80 2281 8390
E-mail: tochio@fujitsu.com

Abstract: This document provides Draft Amendment 1 to G.8121/Y.1381.

This document provides Draft Amendment 1 to G.8121/Y.1381.

Amendment 1:

- Provides new Annex A “Mapping MPLS-TP packets to OTN using IMP”
- Replaces Maintenance Communication Channel (MCC) by Management Communication Channel (MCC) the term “Maintenance Communication Channel (MCC)” by “Management Communication Channel (MCC)”
- Makes editorial corrections in Figure 11-37 to Figure 11-40 (replacing ETH_FP by ETH_AP) and Table 11-17
- Updates the publication dates in References

International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

G.8121/Y.1381

Amendment 1
(xx/202x)~~(11/2018)~~

SERIES G: TRANSMISSION SYSTEMS AND MEDIA, DIGITAL SYSTEMS AND NETWORKS

Packet over Transport aspects – MPLS over Transport aspects

SERIES Y: GLOBAL INFORMATION INFRASTRUCTURE, INTERNET PROTOCOL ASPECTS, NEXT-GENERATION NETWORKS, INTERNET OF THINGS AND SMART CITIES

Internet protocol aspects – Transport

Characteristics of MPLS-TP equipment functional blocks
Amendment 1

Recommendation ITU-T G.8121/Y.1381

ITU-T G-SERIES RECOMMENDATIONS

TRANSMISSION SYSTEMS AND MEDIA, DIGITAL SYSTEMS AND NETWORKS

INTERNATIONAL TELEPHONE CONNECTIONS AND CIRCUITS	G.100–G.199
GENERAL CHARACTERISTICS COMMON TO ALL ANALOGUE CARRIER-TRANSMISSION SYSTEMS	G.200–G.299
INDIVIDUAL CHARACTERISTICS OF INTERNATIONAL CARRIER TELEPHONE SYSTEMS ON METALLIC LINES	G.300–G.399
GENERAL CHARACTERISTICS OF INTERNATIONAL CARRIER TELEPHONE SYSTEMS ON RADIO-RELAY OR SATELLITE LINKS AND INTERCONNECTION WITH METALLIC LINES	G.400–G.449
COORDINATION OF RADIOTELEPHONY AND LINE TELEPHONY	G.450–G.499
TRANSMISSION MEDIA AND OPTICAL SYSTEMS CHARACTERISTICS	G.600–G.699
DIGITAL TERMINAL EQUIPMENTS	G.700–G.799
DIGITAL NETWORKS	G.800–G.899
DIGITAL SECTIONS AND DIGITAL LINE SYSTEM	G.900–G.999
MULTIMEDIA QUALITY OF SERVICE AND PERFORMANCE – GENERIC AND USER-RELATED ASPECTS	G.1000–G.1999
TRANSMISSION MEDIA CHARACTERISTICS	G.6000–G.6999
DATA OVER TRANSPORT – GENERIC ASPECTS	G.7000–G.7999
PACKET OVER TRANSPORT ASPECTS	G.8000–G.8999
Ethernet over Transport aspects	G.8000–G.8099
MPLS over Transport aspects	**G.8100–G.8199**
Synchronization, quality and availability targets	G.8200–G.8299
Service Management	G.8600–G.8699
ACCESS NETWORKS	G.9000–G.9999

For further details, please refer to the list of ITU-T Recommendations.

Recommendation ITU-T G.8121/Y.1381

Characteristics of MPLS-TP equipment functional blocks

Amendment 1

Summary

Recommendation ITU-T G.8121/Y.1381 specifies both the functional components and the methodology that should be used in order to specify multi-protocol label switching – transport profile (MPLS-TP) layer network functionality of network elements; it does not specify individual MPLS-TP network equipment as such.

Amendment 1:

- Provides new Annex A "Mapping MPLS-TP packets to OTN using IMP"
- Replaces Maintenance Communication Channel (MCC) by Management Communication Channel (MCC) the term "Maintenance Communication Channel (MCC)" by "Management Communication Channel (MCC)"
- Makes editorial corrections in Figure 11-37 to Figure 11-40 (replacing ETH_FP by ETH_AP) and Table 11-17
- Updates the publication dates in References

History

Edition	Recommendation	Approval	Study Group	Unique ID*
1.0	ITU-T G.8121/Y.1381	2006-03-29	15	11.1002/1000/8785
1.1	ITU-T G.8121/Y.1381 (2006) Cor. 1	2006-12-14	15	11.1002/1000/9007
1.2	ITU-T G.8121/Y.1381 (2006) Amd. 1	2007-10-22	15	11.1002/1000/9178
2.0	ITU-T G.8121/Y.1381	2012-09-21	15	11.1002/1000/11517
2.1	ITU-T G.8121/Y.1381 (2012) Amd. 1	2012-12-22	15	11.1002/1000/11800
3.0	ITU-T G.8121/Y.1381	2013-11-06	15	11.1002/1000/12020
4.0	ITU-T G.8121/Y.1381	2016-04-13	15	11.1002/1000/12804
4.1	ITU-T G.8121/Y.1381 (2016) Cor. 1	2016-11-13	15	11.1002/1000/13100
4.2	ITU-T G.8121/Y.1381 (2016) Amd. 1	2017-08-13	15	11.1002/1000/13317
5.0	ITU-T G.8121/Y.1381	2018-11-29	15	11.1002/1000/13759
5.1	ITU-T G.8121/Y.1381 (2018) Amd. 1	202x-m-d	15	

* To access the Recommendation, type the URL http://handle.itu.int/ in the address field of your web browser, followed by the Recommendation's unique ID. For example, http://handle.itu.int/11.1002/1000/11830-en.

Keywords

Atomic functions, equipment functional blocks, multi-protocol label switching – transport profile, MPLS-TP, MPLS-TP layer network.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at http://www.itu.int/ITU-T/ipr/.

Table of Contents

Page

Recommendation ITU-T G.8121/Y.1381

Characteristics of MPLS-TP equipment functional blocks[1]

Amendment 1

Editorial note – This is a complete-text publication. Modifications introduced by this Amendment are shown in revision marks relative to Recommendation ITU-T G.8121/Y.1381 (2018).

1 Scope

This Recommendation describes both the functional components and the methodology that should be used in order to describe multi-protocol label switching – transport profile (MPLS-TP) layer network functionality of network elements; it does not describe individual MPLS-TP network equipment as such.

This Recommendation provides a representation of the MPLS-TP technology using the methodologies that have been used for other transport technologies (e.g., synchronous digital hierarchy (SDH), optical transport network (OTN) and Ethernet).[2]

This Recommendation forms part of a suite of Recommendations covering the full functionality of network equipment. These Recommendations are [ITU-T G.705], [ITU-T G.783], [ITU-T G.798], [ITU-T G.806] and [ITU-T G.8021]. This Recommendation also follows the principles defined in [ITU-T G.805].

These Recommendations specify a library of basic building blocks and a set of rules by which they may be combined in order to describe digital transmission equipment. The library comprises the functional building blocks needed to specify completely the generic functional structure of the MPLS-TP layer network. In order to be compliant with this Recommendation, equipment needs to be describable as an interconnection of a subset of these functional blocks contained within this Recommendation. The interconnections of these blocks should obey the combination rules given.

Not every atomic function defined in this Recommendation is required for every application. Different subsets of atomic functions may be assembled in different ways according to the combination rules given in this Recommendation to provide a variety of different capabilities.

---

[1] Cisco Systems has expressed concerns that in the event of a difference between this ITU-T Recommendation and any of the normatively referenced IETF RFCs, interoperability issues may arise. To prevent interoperability issues, the behaviour defined in the IETF RFCs must be maintained, and any such differences must be resolved in coordination with the IETF in a timely manner.

Orange has expressed concerns that in the event of a difference between this ITU-T Recommendation and any of the normatively referenced IETF RFCs, interoperability issues may arise. To prevent interoperability issues, the behaviour defined in the IETF RFCs must be maintained, and any such differences must be resolved in coordination with the IETF in a timely manner.

Verizon Communications has expressed concerns that in the event of a difference between this ITU-T Recommendation and the behavior defined in the normatively referenced IETF RFCs, interoperability issues may arise. There is no guidance in this ITU-T Recommendation that describes how to address differences in behaviour between the Recommendation and the normatively referenced IETF RFCs. Verizon Communications feels that any difference should be resolved in coordination with the IETF in a timely manner and until the issue is resolved, the behavior defined in the IETF RFCs should be maintained.

[2] This ITU-T Recommendation is intended to be aligned with the IETF MPLS RFCs normatively referenced by this Recommendation.

Network operators and equipment suppliers may choose which functions must be implemented for each application.

Figure 1-1 presents the set of atomic functions associated with the traffic signal transport. The functions are based on the functional architecture as described in [ITU-T G.8110.1]. It is noted that this Recommendation only defines Ethernet for the client of MPLS-TP as multi-protocol label switching - transport profile / Ethernet MAC layer network (MT/ETH) adaptation function.

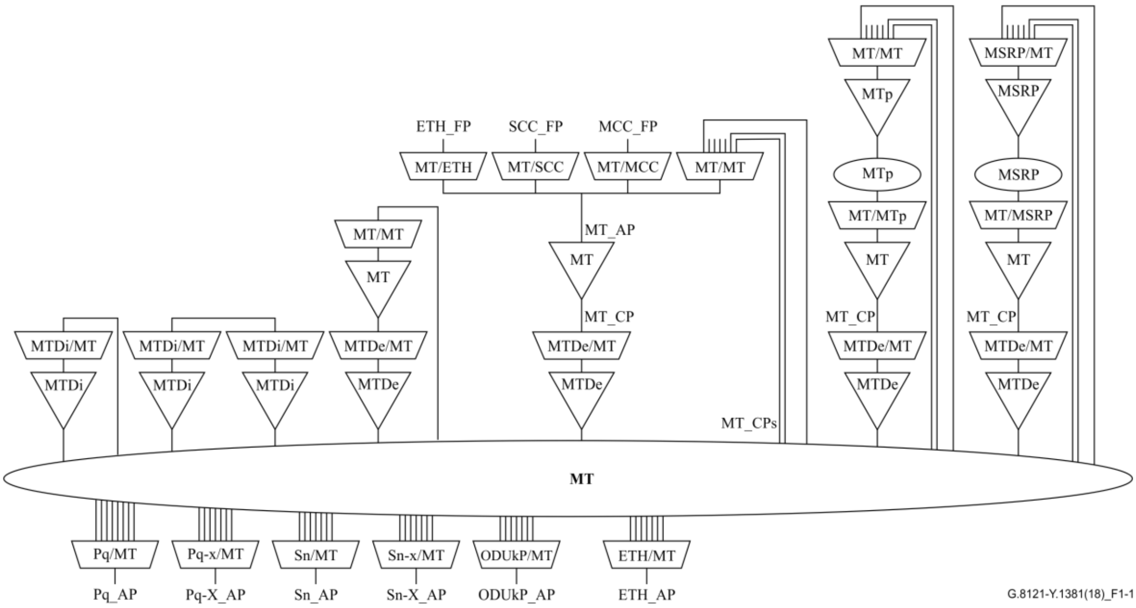


Figure 1-1 – MPLS-TP atomic functions

2 References

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

[ITU-T G.704] Recommendation ITU-T G.704 (1998), *Synchronous frame structures used at 1544, 6312, 2048, 8448 and 44 736 kbit/s hierarchical levels.*

[ITU-T G.705] Recommendation ITU-T G.705 (2000), *Characteristics of plesiochronous digital hierarchy (PDH) equipment functional blocks*.

[ITU-T G.707] Recommendation ITU-T G.707/Y.1322 (2007), *Network node interface for the synchronous digital hierarchy (SDH).*

[ITU-T G.709] Recommendation ITU-T G.709/Y.1331 (2016), *Interfaces for the optical transport network.*

[ITU-T G.783] Recommendation ITU-T G.783 (2006), *Characteristics of synchronous digital hierarchy (SDH) equipment functional blocks*.

[ITU-T G.798] Recommendation ITU-T G.798 (~~2017~~2023), *Characteristics of optical transport network hierarchy equipment functional blocks.*

[ITU-T G.805] Recommendation ITU-T G.805 (2000), *Generic functional architecture of transport networks.*

[ITU-T G.806] Recommendation ITU-T G.806 (2012), *Characteristics of transport equipment – Description methodology and generic functionality.*

[ITU-T G.832] Recommendation ITU-T G.832 (1998), *Transport of SDH elements on PDHnetworks – Frame and multiplexing structures.*

[ITU-T G.7041] Recommendation ITU-T G.7041/Y.1303 (2016), *Generic framing procedure*.

[ITU-T G.7043] Recommendation ITU-T G.7043/Y.1343 (2004), *Virtual concatenation of plesiochronous digital hierarchy (PDH) signals.*

[ITU-T G.7044] Recommendation ITU-T G.7044/Y.1347 (2011), *Hitless adjustment of ODUflex(GFP).*

[ITU-T G.7712] Recommendation ITU-T G.7712/Y.1703 (2019~~10~~), *Architecture and specification of data communication network.*

[ITU-T G.8021] Recommendation ITU-T G.8021/Y.1341 (2022~~18~~), *Characteristics of Ethernet transport network equipment functional blocks.*

[ITU-T G.8040] Recommendation ITU-T G.8040/Y.1340 (2005), *GFP frame mapping into Plesiochronous Digital Hierarchy (PDH).*

[ITU-T G.8110.1] Recommendation ITU-T G.8110.1/Y.1370.1 (2011), *Architecture of the Multi-Protocol Label Switching transport profile layer network.*

[ITU-T G.8131] Recommendation ITU-T G.8131/Y.1382 (2014), *Linear protection switching for MPLS transport profile.*

[ITU-T G.8251] Recommendation ITU-T G.8251 (2010), *The control of jitter and wander within the optical transport network (OTN).*

[ITU-T Y.1415] Recommendation ITU-T Y.1415 (2005), *Ethernet-MPLS network interworking – User plane interworking*.

[IETF RFC 3031] IETF RFC 3031 (2001), *Multiprotocol Label Switching Architecture.*

[IETF RFC 3032] IETF RFC 3032 (2001), *MPLS Label Stack Encoding.*

[IETF RFC 3270] IETF RFC 3270 (2002), *Multi-Protocol Label Switching (MPLS) Support of Differentiated Services.*

[IETF RFC 4448] IETF RFC 4448 (2006), *Encapsulation Methods for Transport of Ethernet over MPLS Networks*.

[IETF RFC 4720] IETF RFC 4720 (2006), *Pseudowire Emulation Edge-to-Edge (PWE3) – Frame Check Sequence Retention*.

[IETF RFC 5332] IETF RFC 5332 (2008), *MPLS Multicast Encapsulation*.

[IETF RFC 5462] IETF RFC 5462 (2009), *Multiprotocol Label Switching (MPLS) Label Stack Entry: "EXP" Field Renamed to "Traffic Class" Field.*

[IETF RFC 5586] IETF RFC 5586 (2009), *MPLS Generic Associated Channel*.

[IETF RFC 5718] IETF RFC 5718 (2010), *An In-Band Data Communication Network For the MPLS Transport Profile*.

[IETF RFC 6371] IETF RFC 6371 (2011), *Operations, Administration and Maintenance Framework for MPLS-Based Transport Networks.*

[IETF RFC 6374] IETF RFC 6374 (2011), *Packet Loss and Delay Measurement for MPLS Networks*.

[IETF RFC 6435] IETF RFC 6435 (2011), *MPLS Transport Profile Lock Instruct and Loopback Functions*, plus Errata 3429 (2013)

3 Definitions

3.1 Terms defined elsewhere

This Recommendation uses the following terms defined elsewhere:

3.1.1 **access point**: [ITU-T G.805]

3.1.2 **adapted information**: [ITU-T G.805]

3.1.3 **associated channel header**: [IETF RFC 5586]

3.1.4 **bottom of stack**: [IETF RFC 3032]

3.1.5 **characteristic information**: [ITU-T G.805]

3.1.6 **client/server relationship**: [ITU-T G.805]

3.1.7 **connection**: [ITU-T G.805]

3.1.8 **connection point**: [ITU-T G.805]

3.1.9 **explicitly TC-encoded-PSC LSP**: [IETF RFC 5462]

3.1.10 **G-ACh label**: [IETF RFC 5586]

3.1.11 **generic associated channel**: [IETF RFC 5586]

3.1.12 **label**: [IETF RFC 3031]

3.1.13 **label-only-inferred PSC LSP**: [IETF RFC 3270]

3.1.14 **label stack**: [IETF RFC 3031]

3.1.15 **label switched path**: [IETF RFC 3031]

3.1.16 **label value**: [IETF RFC 3032]

3.1.17 **layer network**: [ITU-T G.805]

3.1.18 **matrix**: [ITU-T G.805]

3.1.19 **MPLS label stack**: [IETF RFC 3031]

3.1.20 **network**: [ITU-T G.805]

3.1.21 **network connection**: [ITU-T G.805]

3.1.22 **per-hop behaviour**: [IETF RFC 3270]

3.1.23 **reference point**: [ITU-T G.805]

3.1.24 **subnetwork**: [ITU-T G.805]

3.1.25 **subnetwork connection**: [ITU-T G.805]

3.1.26 **termination connection point**: [ITU-T G.805]

3.1.27 **time-to-live**: [IETF RFC 3031]

3.1.28 **traffic class**: [IETF RFC 5462]

3.1.29 **trail**: [ITU-T G.805]

3.1.30 **trail termination**: [ITU-T G.805]

3.1.31 **transport**: [ITU-T G.805]

3.1.32 **transport entity**: [ITU-T G.805]

3.1.33 **transport processing function**: [ITU-T G.805]

3.1.34 **unidirectional connection**: [ITU-T G.805]

3.1.35 **unidirectional trail**: [ITU-T G.805]

3.2 Terms defined in this Recommendation

None.

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

ACH	Associated Channel Header
AI	Adapted Information
AIS	Alarm Indication Signal
AP	Access Point
APC	Automatic Protection Coordination
APS	Automatic Protection Switching
APSb	APS packet with incompatible "PT"
APSc	APS packet with incompatible Capabilities TLV
BWR	Bandwidth Resize
CC	Continuity Check
CC-V	Continuity Check and Connectivity Verification
CC/CV	Continuity Check or Connectivity Verification
CI	Characteristic Information
CII	Common Interworking Indicator
CoS	Class of Service
CP	Connection Point
CSF	Client Signal Fail
CSP	Client Specific Process
CV	Connectivity Verification
CW	Control Word
DCI	Detect Clearance Indication
DEG	Degraded Signal
DM	Delay Measurement
DP	Drop Precedence
DT	Diagnostic Test

ECC	Embedded Communication Channel
EMF	Equipment Management Function
ETH	Ethernet MAC layer network
ETY	Ethernet PHY layer network
E-LSP	Explicitly TC-encoded-PSC LSP
FDI	Forward Defect Indication
FP	Flow Point
FTP	Flow Termination Point
G-ACh	Generic Associated Channel
GAL	G-ACh Label
GFP	Generic Framing Procedure
HAO	Hitless Adjustment of ODUflex
IMP	Idle Mapping Procedure
iPHB	incoming PHB
L-LSP	Label-only-inferred PSC LSP
LCAS	Link Capacity Adjustment Scheme
LCK	Lock
LER	Label Edge Router
LKI	Lock Instruct
LKR	Lock Report
LM	Loss Measurement
LOC	Loss of Continuity
LOS	Loss of Signal
LSE	Label Stack Entry
LSP	Label Switched Path
LSR	Label Switching Router
LStack	Label Stack
MAC	Media Access Control
MCC	Management ~~Maintenance~~ Communication Channel
MEG	Maintenance Entity Group
MEL	Maintenance Entity group Level
MEP	Maintenance entity group (MEG) End Point
MI	Management Information
MIP	Maintenance entity group (MEG) Intermediate Point
MP	Management Point
MMG	Mismerge

MPLS	Multi-Protocol Label Switching
MPLS-TP	Multi-Protocol Label Switching – Transport Profile
MSRP	MPLS-TP Shared Ring Protection
MT	Multi-Protocol Label Switching – Transport Profile
MTDe	MPLS-TP MEP Diagnostic function
MTDi	MPLS-TP MIP Diagnostic function
NCM	Network Connection Monitoring
NCS	Network Connectivity Status
NE	Network Element
NMS	Network Management System
OAM	Operation, Administration and Maintenance
ODU	Optical channel Data Unit
ODUk	Optical channel Data Unit – order k
oPHB	outgoing PHB
OPU	Optical Payload Unit
OPUk	Optical Payload Unit of level k
OTH	Optical Transport Hierarchy
OTN	Optical Transport Network
PDU	Protocol Data Unit
PFI	Payload Frame check sequence Indication
PHB	Per Hop Behaviour
PLM	Payload Mismatch
PM	Performance Monitoring
POH	Path Overhead
PSC	PHB Scheduling Class
PSI	Payload Structure Indication
PT	Payload Type
PTI	Payload Type Identifier
PW	Pseudowire
P11s	1 544 kbit/s PDH path layer with synchronous 125 µs frame structure according to [ITU-T G.704]
P12s	2 048 kbit/s PDH path layer with synchronous 125 µs frame structure according to [ITU-T G.704]
P31s	34 368 kbit/s PDH path layer with synchronous 125 µs frame structure according to [ITU-T G.832]
P32e	44 736 kbit/s PDH path layer with frame structure according to [ITU-T G.704]
RCOH	Resize Control Overhead

RDI	Remote Defect Indication
RES	Reserved overhead
RFC	IETF Request For Comments
RI	Remote Information
RP	Remote Point
RPS	Ring Protection Switch
RT	Route Tracing
SCC	Signalling Communication Channel
SCCType	Signalling Communication Channel Type
SCN	Signalling Communication Network
SD	Signal Degrade
SDH	Synchronous Digital Hierarchy
SDL	Specification and Description Language
SLM	Synthetic Loss Measurement
SNC	Subnetwork Connection
SNCP	Subnetwork Connection Protection
SNC/S	SNCP with Sublayer monitoring
SPME	Sub-Path Maintenance Element
SSD	Server Signal Degrade
SSF	Server Signal Fail
TC	Traffic Class
TCM	Tandem Connection Monitoring
TCP	Termination Connection Point
TFP	Termination Flow Point
TH	Throughput
TLV	Type Length Value
TSD	Trail Signal Degrade
TSF	Trail Signal Fail
TTL	Time-To-Live
TTSI	Trail Termination Source Identifier
UNC	Unexpected CoS
UNM	Unexpected MEP
UNP	Unexpected Period
UPI	User Payload Identifier
VLI	Virtual concatenation/Link capacity adjustment scheme Information
WTR	Wait-To-Restore

5 Conventions

The diagrammatic convention for connection-oriented layer networks described in this Recommendation is that of [ITU-T G.805].

6 Supervision

The generic supervision functions are defined in clause 6 of [ITU-T G.806]. Specific supervision functions for the MPLS-TP network are defined in this clause.

6.1 Defects

6.1.1 Summary of entry/exit conditions for defects

The defect entry and exit conditions are based on events. Occurrence or absence of specific events may raise or reset specific defects.

In the following:

Valid means a received value is *equal* to the value configured via the management information (MI) input interface(s).

Invalid means a received value is *not equal* to the value configured via the MI input interface(s).

The events defined for this Recommendation are summarized in Table 6-1 as a quick overview. Events, other than the protection switching events, are generated by processes in the MT_TT_Sk function as defined in clause 9.2. These processes define the exact conditions for these events; Table 6-1 only provides a quick overview.

Table 6-1 – Overview of events

Event	Meaning
unexpMEG	Reception of a CC-V packet with an invalid maintenance entity group (MEG) value.(Note 1) NOTE – Clause 5.1.1 of [IETF RFC 6371] describes the conditions when a received CC-V packet is considered to have invalid MEG and maintenance entity group end point (MEP) values. In case (1) a CC packet is received by a sink MEP monitoring the MEG for CC and CV functions, or (2) a CV packet is received by a sink MEP monitoring the MEG for CC-only function; the received CC-V packet is considered as having an invalid MEG value (thus triggering the unexpMEG event)
unexpMEP	Reception of a CV packet with an invalid MEP value, but with a valid MEG value.
unexpPeriod	Reception of a CC-V packet with an invalid Periodicity value, but with valid MEG and MEP values.
unexpCoS	Reception of a CC-V packet with an invalid traffic class (TC) value, but with valid MEG and MEP values.
expCC-V	Reception of a CC-V packet with valid MEG and MEP values.
RDI=x	Reception of a CC-V packet for the peer MEP with the remote defect indication (RDI) information indicate to x; where x=0 (remote defect clear) and x=1 (remote defect set).
LCK	Reception of a locked (LCK) packet. (Note 2)
AIS	Reception of an alarm indication signal (AIS) packet.

Table 6-1 – Overview of events

Event	Meaning
BS	Bad second, a second in which the lost frame ratio exceeds the bad second threshold (BS_THR).
CSF-LOS	Reception of a client signal fail (CSF) packet that indicates "Client Loss of Signal".
CSF-FDI	Reception of a CSF packet that indicates "Client Forward Defect Indication".
CSF-RDI	Reception of a CSF packet that indicates "Client Reverse Defect Indication".
expAPS	Reception of a valid APS packet. (Note3)
APSw	Reception of an automatic protection switching (APS) packet from the working transport entity.
APSb	Reception of an APS packet with incompatible "PT" field value.
APSc	Reception of an APS packet with incompatible "Capabilities TLV" information.
APSr	Reception of an APS packet with incompatible "Data Path" value. (Note 4)

NOTE 1 – According to [IETF RFC 6371], a CC-V packet is either a CC packet or a CV packet. A CV packet performs both CC and CV OAM functions. A CC packet performs only CC OAM function.

NOTE 2 – IETF uses this term LCK as lock report (LKR) and lock instruct (LKI) in [IETF RFC 6371]

NOTE 3 – For the term APS, [ITU-T G.8131] conventionally uses the term automatic protection coordination (APC) for the protocol of MPLS-TP linear protection.

NOTE 4 – One way to detect this event is to detect that the transmitted "Data Path" and the received "Data Path" values differ, for example in case traffic switching occurs due to a local request.

The occurrence or absence of these events may detect or clear a defect. An overview of the conditions is given in Table 6-2. The notation "#event=x (K × period)" is used to indicate the occurrence of x events within the period as specified between the brackets.

Table 6-2 gives a quick overview of the types of defects for MPLS-TP layer and the raising and clearing conditions for these defects as described in [IETF RFC 6371].

Table 6-2 – Overview of detection and clearing conditions

Defect	Defect detection	Clearing condition
dLOC	#expCC-V==0 (K × CC_Period)	expCC-V
dUNC	unexpCoS	#unexpCoS==0 (K × CC-V_Period)
dMMG	unexpMEG	#unexpMEG==0 (K × CC-V _Period)
dUNM	unexpMEP	#unexpMEP==0 (K × CC-V_Period)
dUNP	unexpPeriod	#unexpPeriod==0 (K × CC-V_Period)
dRDI	RDI==1	RDI==0
dAIS	AIS	#AIS==0 (K × AIS_Period)
dLCK	LCK	#LCK==0 (K × LCK_Period)
dCSF-LOS	CSF-LOS	#CSF-LOS == 0 (K × CSF_Period or CSF-DCI)
dCSF-FDI	CSF-FDI	#CSF-FDI == 0 (K × CSF_Period or CSF-DCI)

Table 6-2 – Overview of detection and clearing conditions

Defect	Defect detection	Clearing condition
dCSF-RDI	CSF-RDI	#CSF-RDI == 0 (K × CSF_Period or CSF-DCI)
dDEG	#BS==DEGM (DEGM × 1second)	#BS==0 (M × 1second)
dFOP-PMb	APSb	expAPS
dFOP-PMc	APSc	expAPS
dFOP-NR	APSr continues more than 50ms	expAPS
dFOP-CM	APSw	#APSw == 0 (K × normal APS Period)
dFOP-TO	#expAPS==0 (K × long APS interval)	expAPS

6.1.2 Continuity supervision

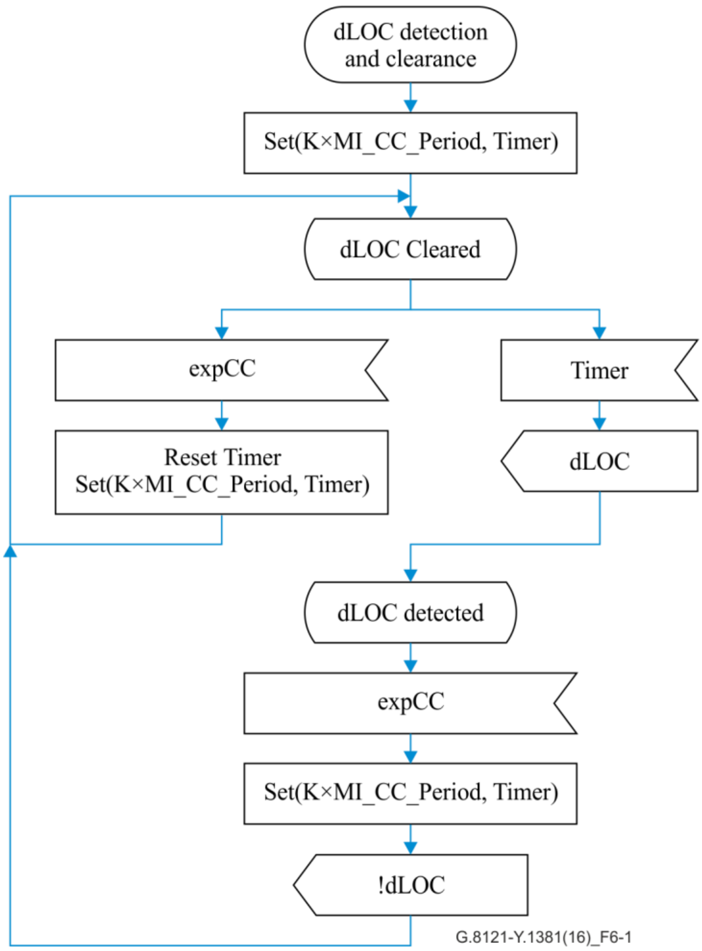


Figure 6-1 – dLOC detection and clearance process

6.1.2.1 Loss of continuity defect (dLOC)

The loss of continuity defect is calculated at the MT layer. It monitors the presence of continuity in MT trails.

Its detection and clearance are defined in Figure 6-1. The 'period' in Figure 6-1 is set to K×MI_CC_Period, where MI_CC_Period corresponds to the configured CC Period and K is such that 3.25≤K≤3.5.

6.1.3 Connectivity supervision

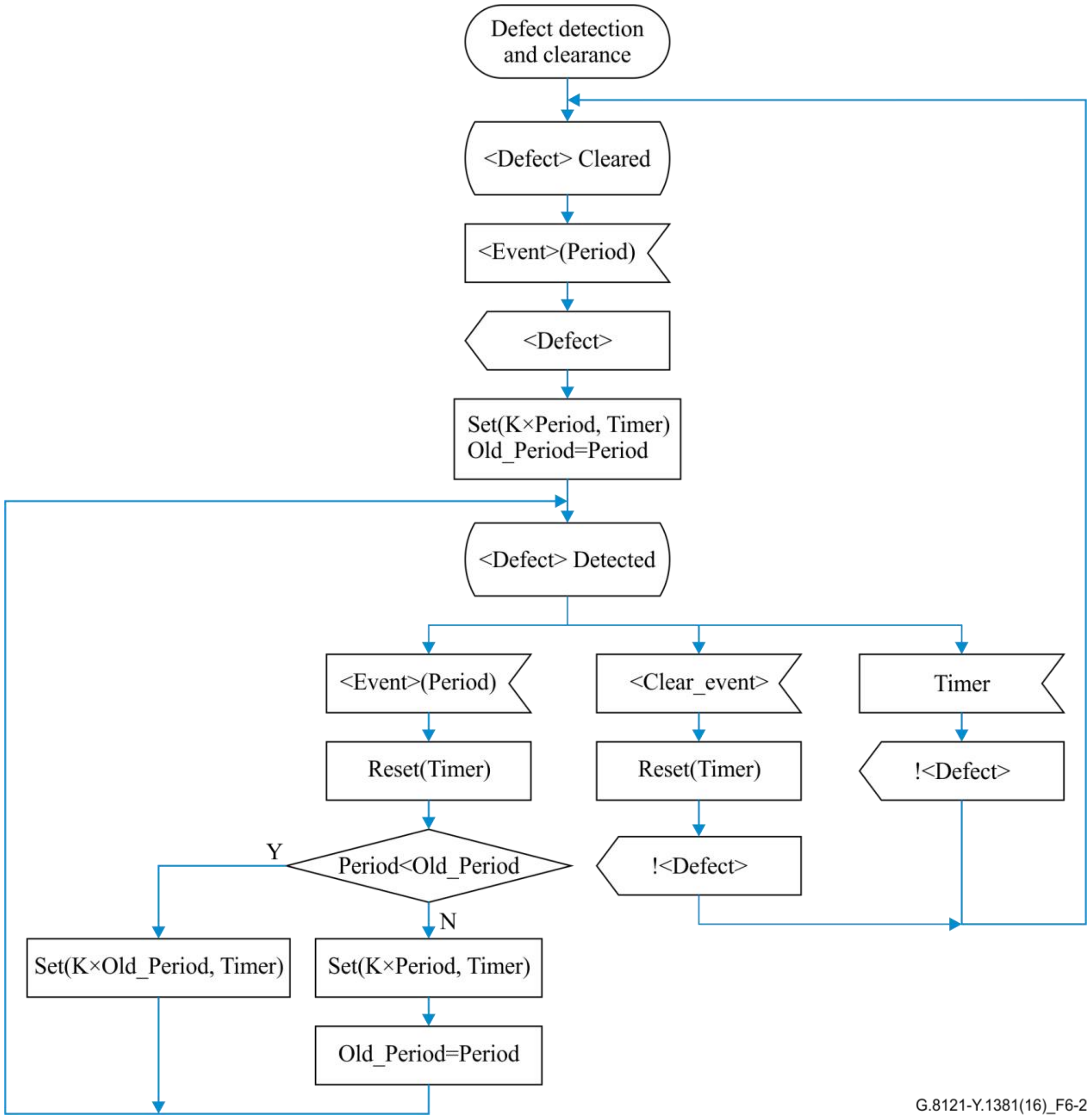


Figure 6-2 – Defect detection and clearance process for dMMG, dUNM, dUNP, dUNC, dAIS, dLCK and dCSF

Figure 6-2 shows a generic state diagram that is used to detect and clear the dMMG, dUNM, dUNP, dUNC, dAIS, dLCK and dCSF (dCSF-LOS, dCSF-FDI, and dCSF-RDI) defects. In this diagram <Defect> needs to be replaced with the specific defect and <Event> with the specific event related to this defect. Furthermore, in Figure 6-2, 3.25 ≤ K ≤ 3.5.

Figure 6-2 shows that the Timer is set based on the last received period value, unless an earlier operation, administration and maintenance (OAM) packet triggering <Event> (and therefore the detection of <Defect>) carried a longer period. As a consequence, clearing certain defects may take more time than necessary.

6.1.3.1 Mismerge defect (dMMG)

The mismerge defect detect is calculated at the MT layer. It monitors the connectivity in a maintenance entity group (MEG).

Its detection and clearance are defined in Figure 6-2. The <Defect> in Figure 6-2 is dMMG. The <Event> in Figure 6-2 is the unexpectedMEG event and the "Period" is the period carried in the CV packet that triggered the event, unless an earlier CV packet triggering an unexpectedMEG event carried a greater period.

6.1.3.2 Unexpected MEP defect (dUNM)

The unexpected MEP defect is calculated at the MT layer. It monitors the connectivity in a maintenance entity group.

Its detection and clearance are defined in Figure 6-2. The <Defect> in Figure 6-2 is dUNM. The <Event> in Figure 6-2 is the unexpectedMEP event and the Period is the Period carried in the CV packet that triggered the event, unless an earlier CV packet triggering an unexpectedMEP event carried a greater period.

6.1.3.3 Degraded signal defect (dDEG)

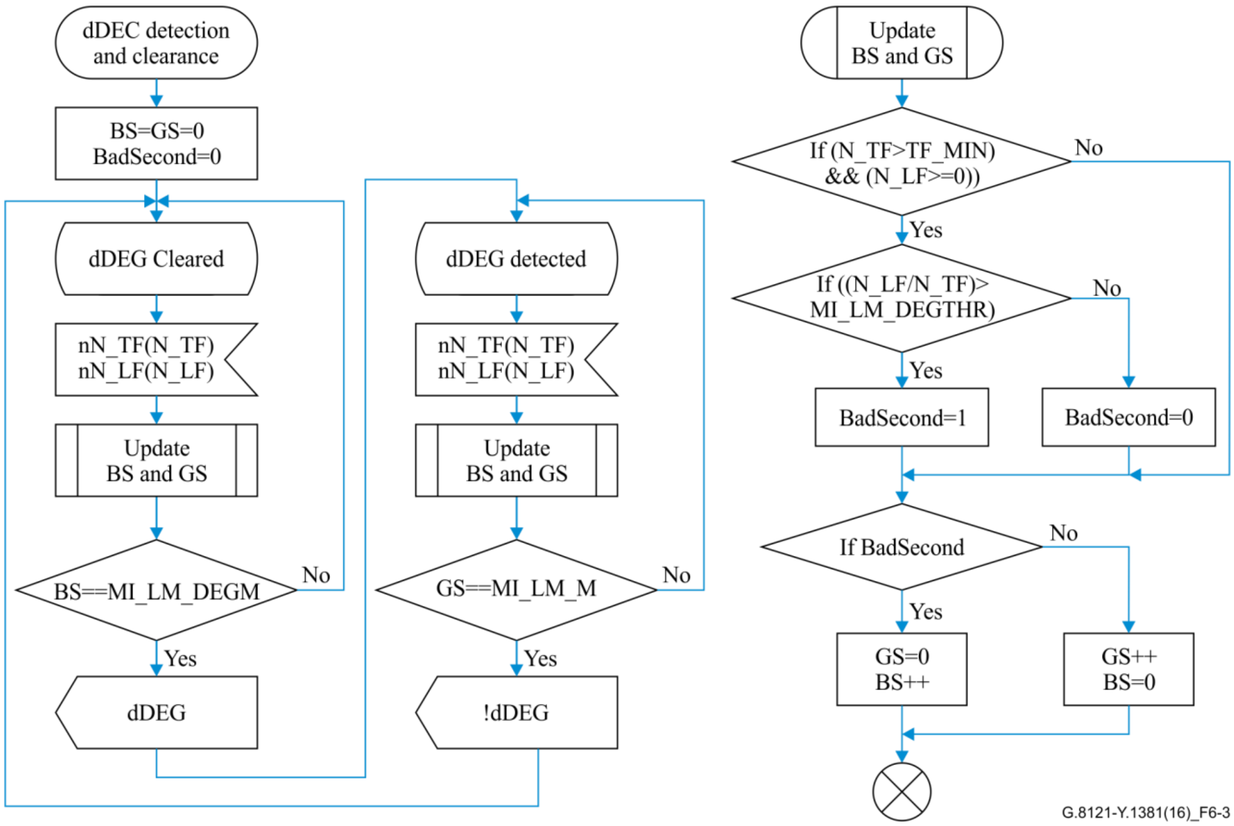


Figure 6-3 – dDEG detection and clearance process

The degraded signal defect (dDEG) is calculated at the MT layer. It monitors the connectivity of a MT trail.

Its detection and clearance are defined Figure 6-3.

Every second the state machine receives the 1 second counters for the near end received and transmitted frames and determines whether the second was a bad second. The defect is detected if there are MI_LM_DEGM consecutive bad seconds and cleared if there are MI_LM_M consecutive good seconds.

In order to declare a bad second, the number of transmitted frames must exceed a threshold (TF_MIN). If this is true, then a bad second is declared if either the frame loss is negative (i.e., there are more frames received than transmitted) or the frame loss ratio (i.e., lost frames/transmitted frames) is greater than MI_LM_DEGTHR.

6.1.4 Protocol supervision

6.1.4.1 Unexpected periodicity defect (dUNP)

The unexpected periodicity defect is calculated at the MT layer. It detects the configuration of different periodicities at different MEPs belonging to the same MEG.

Its detection and clearance are defined Figure 6-2. The <Defect> in Figure 6-2 is dUNP. The <Event> in Figure 6-2 is the unexpectedPeriod event and the "Period" is the period carried in the CC-V packet that triggered the event, unless an earlier CC-V packet triggering an unexpectedPeriod event carried a greater period.

6.1.4.2 Unexpected CoS defect (dUNC)

The unexpected class of service (CoS) defect is detected at the MT layer. It detects the configuration error of different CoS at different MEPs belonging to the same MEG.

Its detection and clearance are defined Figure 6-2.

The <Defect> in Figure 6-2 is dUNC. The <Event> in Figure 6-2 is the unexpectedCoS event and the "Period" is the period associated with the CC-V packet that triggered the event, unless an earlier CC-V packet triggering an unexpectedCoS event carried a greater period.

6.1.4.3 Protection protocol supervision

For linear protection, the failure of protocol defects are applied in the case of a bidirectional protection switching operation, where the coordination between the two ends is needed.

6.1.4.3.1 Linear protection failure of protocol provisioning mismatch on bridge type (dFOP-PMb)

The failure of protocol provisioning mismatch – bridge type defect is calculated at the MT layer. It monitors the provisioning mismatch on bridge type:

– By comparing bridge type (i.e., selector bridge or permanent bridge) of the transmitted and the received APS packets. The reception of an APS packet with incompatible bridge type results in APSb event as defined in Table 6-1.

– In case of APS protocol defined in [ITU-T G.8131], the bridge type is signalled in the Protection type field as defined in clause 8 of [ITU-T G.8131]. If the value of the PT field of one side is 2 (i.e., selector bridge) and the value of PT field of the other side is 1 or 3 (i.e., permanent bridge), then this will result in an APCb event as defined in Table 6-1.

dFOP-PMb defect detection and clearing condition are defined in Table 6-2.

dFOP-PMb is detected on the receipt of an APSb event and is cleared on the receipt of an expAPS event.

The detection and clearance events of the defects are generated by the subnetwork connection protection process (clause 9.1.1).

6.1.4.3.2 Linear protection failure of protocol provisioning mismatch on Capabilities TLV (dFOP-PMc)

The failure of protocol provisioning mismatch – Capabilities TLV defect is calculated at the MT layer. It monitors the provisioning mismatch on Capabilities TLV as defined in clause 8.1 of [ITU-T G.8131]:

– by comparing the 12 octet Capabilities TLV information of the received APS packets with the default values defined in clause 8.1 of [ITU-T G.8131].

dFOP-PMc defect detection and clearing condition are defined in Table 6-2.

dFOP-PMc is detected on the receipt of an APSc event and is cleared on the receipt of an expAPS event.

The detection and clearance events of the defects are generated by the subnetwork connection protection process (clause 9.1.1).

6.1.4.3.3 Linear protection failure of protocol no response (dFOP-NR)

The failure of protocol no response defect is calculated at the MT layer. It monitors incompletion of protection switching by comparing the transmitted "Data Path" and the received "Data Path" values in the APS protocol.

dFOP-NR defect detection and clearing condition are defined in Table 6-2.

dFOP-NR is detected when an APSr event continues for more than 50ms and it is cleared on the receipt of the expAPS event.

The detection and clearance events of the defects are generated by the subnetwork connection protection process (clause 9.1.1).

6.1.4.3.4 Linear protection failure of protocol configuration mismatch (dFOP-CM)

The failure of protocol configuration mismatch defect is calculated at the MT layer. It monitors working and protection configuration mismatch by detecting the receipt of the APS packet from the working transport entity.

dFOP-CM defect detection and clearing condition are defined in Table 6-2.

dFOP-CM is detected on the receipt of an APSw event and is cleared on the receipt of no APSw event during K times the long APS transmission period defined in [ITU-T G.8131], where $3.25 \le K \le 3.5$.

The detection and clearance events of the defects are generated by the subnetwork connection protection process (clause 9.1.1).

6.1.4.3.5 Linear protection failure of protocol time out (dFOP-TO)

The failure of protocol time out defect is calculated at the MT layer. It monitors the time out defect by detecting the prolonged absence of expected APS packets.

dFOP-TO defect detection and clearing condition are defined in Table 6-2.

dFOP-TO is detected on the receipt of no expAPS event during K times the long APS interval defined in [ITU-T G.8131] (where K >= 3.5). dFOP-TO is cleared on the receipt of an expAPS event.

The detection and clearance events of the defects are generated by the subnetwork connection protection process (clause 9.1.1).

6.1.5 Maintenance signal supervision

6.1.5.1 Remote defect indicator defect (dRDI)

The remote defect indicator defect (dRDI) is detected at the MT layer. It monitors the presence of the RDI maintenance signal.

dRDI is detected on the receipt of the RDI=1 event and is cleared on receipt of the RDI=0 event.

6.1.5.2 Alarm indication signal defect (dAIS)

The alarm indication signal defect (dAIS) is detected at the MT layer. It monitors the presence of the AIS maintenance signal.

dAIS defect detection and clearance process are illustrated in Figure 6-2. The <Defect> in Figure 6-2 is dAIS. The <Event> in Figure 6-2 is the AIS event and the "Period" is the period associated with the AIS packet unless an earlier AIS packet was associated with a greater period.

6.1.5.3 Locked defect (dLCK)

The locked defect (dLCK) is detected at the MT layer. It monitors the presence of the locked maintenance signal.

dLCK defect detection and clearance process are illustrated in Figure 6-2. The <Defect> in Figure 6-2 is dLCK. The <Event> in Figure 6-2 is the LCK event and the "Period" is the period associated with the LCK packet unless an earlier LCK packet was associated with a greater period.

6.1.5.4 Client signal fail defect (dCSF)

The CSF (i.e., CSF-LOS, CSF-FDI and CSF-RDI) defect is detected at the MT layer. It monitors the presence of the CSF maintenance signal.

dCSF defect detection and clearance process conditions are illustrated in Figure 6-2. The <Defect> in Figure 6-2 is dCSF-LOS, dCSF-FDI or dCSF-RDI. The <Event> in Figure 6-2 is the CSF event and the "Period" is the period associated with the CSF packet unless an earlier CSF packet was associated with a greater period

The <Clear_event> in Figure 6-2 is the CSF event which indicates detect clearance indication (DCI).

6.2 Consequent actions

For generic consequent actions, see [ITU-T G.806]. For the specific consequent actions applicable to MPLS-TP, refer to the specific atomic functions.

6.3 Defect correlations

For the defect correlations, see the specific atomic functions.

6.4 Performance filters

For further study.

7 Information flow across reference points

Information flow for MPLS-TP functions is defined in clause 9. A generic description of information flow is defined in clause 7 of [ITU-T G.806].

8 MPLS-TP processes

This clause defines the specific processes for the MPLS-TP network. Generic processes are defined in clause 8 of [ITU-T G.806].

8.1 G-ACh process

8.1.1 Overview

In order to ensure proper operational control, MPLS-TP network elements exchange OAM packets that strictly follow the same path as user traffic packets; that is, OAM packets are subject to the exact same forwarding schemes (e.g., fate sharing) as the user traffic packets. These OAM packets can be distinguished from the user traffic packets by using the G-ACh and G-ACh label (GAL) constructs.

The G-ACh is a generic associated control channel mechanism for "sections", label switched paths (LSPs) and pseudowires (PWs), over which OAM and other control messages can be exchanged. The GAL is a label based exception mechanism to alert label edge routers/label switching routers (LERs/LSRs) of the presence of an associated channel header (ACH) after the bottom of the stack.

The format of GAL and ACH is described in [IETF RFC 5586].

8.1.2 G-ACh insertion process

Figure 8-1 describes G-ACh insertion process.

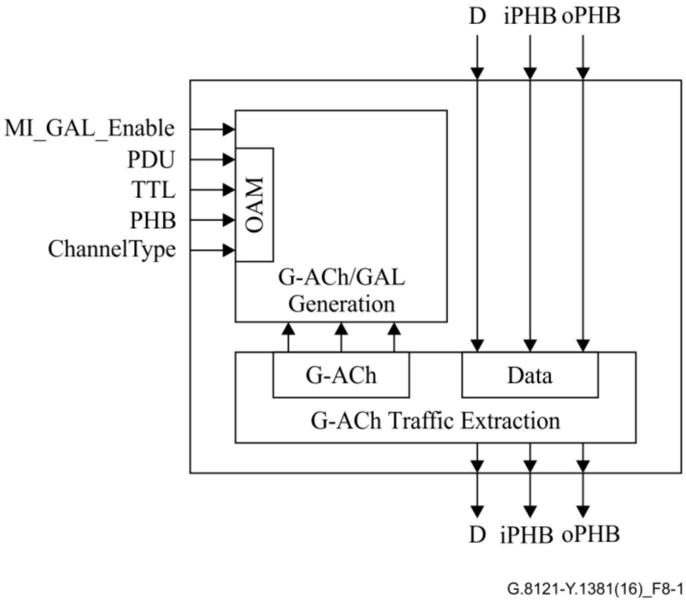


Figure 8-1 – G-ACh insertion process

The G-ACh insertion process encapsulates OAM packets and multiplexes them with the data packets. The data packets are passed through unchanged, while the OAM packets are encapsulated as follows:

A G-ACh header is prepended to the OAM PDU, with the channel type set to the specified value. If MI_GAL_Enable is true, the process then further prepends a G-ACh label (GAL) as described in [IETF RFC 5586]. If the time-to-live (TTL) signal is not specified, the TTL field in the MI_CI_D is set to 255; otherwise, it is set to the value in the TTL signal. If a GAL is inserted, the TTL field in the GAL label stack entry (LSE) is set as defined in [IETF RFC 5586].

NOTE 1 – Certain OAM packets can be addressed to a MIP and thus need to be inserted with a specific TTL to ensure that the TTL expires at the target MIP. OAM packets addressed to a MEP have the TTL set to 255.

NOTE 2 – MI_GAL_Enable must be set to true on LSPs and to false on PWs. Setting it to true for PWs is for further study.

8.1.3 G-Ach extraction process

Figure 8-2 describes G-ACh extraction process.

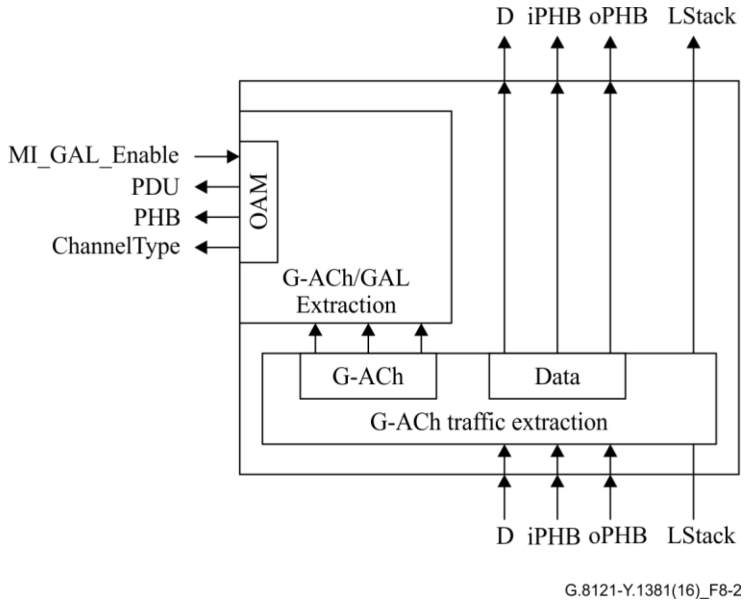


Figure 8-2 – G-ACh extraction process

The G-ACh traffic unit will be extracted if it includes GAL and ACH in the incoming data when MI_GAL_Enable is set.

8.2 TC/Label processes

The TC/Label processes can operate in two modes, as described in clause 7.3 of [ITU-T G.8110.1]. Mode 1 is the default mode and is mandatory. Mode 2 is optional and is only used for section monitoring. The mode is set as MI_Mode by the equipment management function (EMF); it is expected that the EMF sets this automatically to the appropriate value depending on whether the MEP is monitoring a section or a PW/LSP. It is not expected that this MI is exposed to the user.

NOTE – MI_Mode is always set to mode 1 when the TC/Label process is within a MT/MT_A function, since a MPLS section is the lowest monitoring level and is only applicable to monitoring directly over the (non-MPLS) server layer. In a Server/MT_A function, the TC/Label process may operate in mode 1 (if there is no section MEP) or mode 2 (if there is a section MEP).

8.2.1 TC/Label source processes

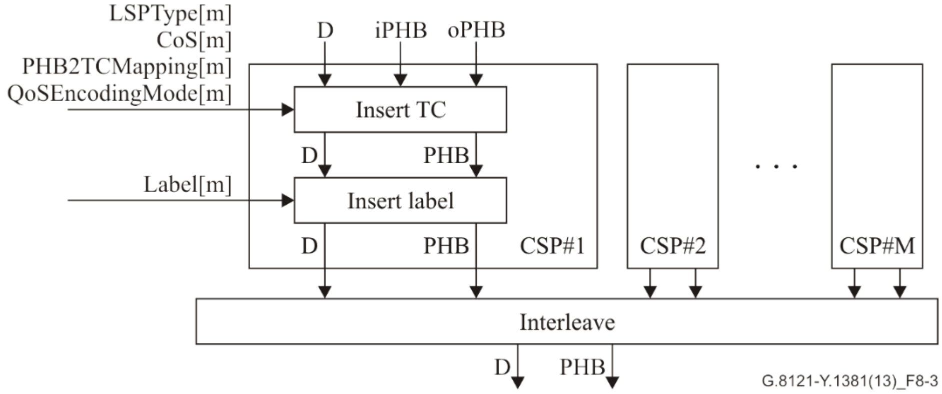


Figure 8-3 – TC/Label source processes in mode 1

Figure 8-3 shows the TC/Label source processes when MI_Mode is set to mode 1. These processes are performed on a packet-per-packet basis.

Client specific processes: The function supports M ($M \leq 2^N - 16$, with N = 20 for MPLS label) client specific processes (CSP#1 to CSP#M), each connected to a single MPLS-TP connection point. CSP#m ($1 \leq m \leq M$) is active when Label[m] has a value in the range of 16 to $2^N - 1$.

TC insertion process: Insert the TC field, encoding the PHB information according to the following rules:

– If LSPType[m] = L-LSP, the drop precedence (DP) information is encoded into the TC field according to [ITU-T G.8110.1] and CoS[m].

– If LSPType[m] = E-LSP, the PHB information is encoded into the TC field according to the 1:1 mapping configured in the PHB2TCMapping[m].

NOTE – E-LSP and L-LSP are referred to in [ITU-T G.8110.1]

The PHB information to map into the TC field is selected according to the following rules:

– If QoSEncodingMode[m] = A, the iPHB information is mapped into the TC field.

– If QoSEncodingMode[m] = B, the oPHB information is mapped into the TC field.

Label insertion process: Insert the 20-bit MPLS Label field with the value provided via Label[m].

Interleave process: Interleave the MPLS-TP traffic units from the client specific processes into a single stream.

When MI_Mode is set to mode 2, the TC/Label source process simply deletes the TTL and S fields from each MPLS traffic unit.

8.2.2 TC/Label sink processes

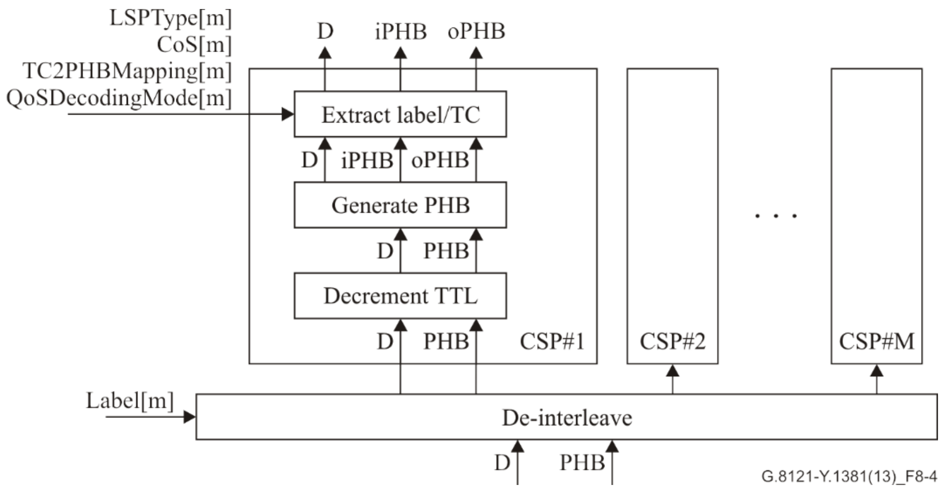


Figure 8-4 – TC/Label sink processes in mode 1

Figure 8-4 shows the TC/Label sink processes when MI_Mode is set to mode 1. These processes are performed on a packet-per-packet basis.

Deinterleave process: Deinterleaves the MPLS-TP traffic units and forwards each of its client specific process #m based on the value in the Label field of the traffic unit. Relation between CSP and MPLS label value is provided by Label[1..M].

Traffic units received with a label value identifying a non-active CSP are dropped.

Client specific processes: The function supports M (M ≤ 2^N − 16, with N = 20 for MPLS label) client specific processes (CSP#1 to CSP#M), each connected to a single MPLS-TP connection point. CSP#m (1 ≤ m ≤ M) is active when Label[m] has a value in the range of 16 to 2^N − 1.

Label and TC extraction process: Extracts the MPLS label and the TC fields from the traffic unit.

TTL decrement process: Decrements the TTL. If the MPLS-TP CP is not a TCP and the decremented TTL is less than or equal to zero, the traffic unit is dropped silently.

NOTE – MIPs and MEPs compound functions are connected to the Server/MT_A (or MT/MT_A) functions via a MPLS-TP TCP.

PHB generation process: Processes the TC field.

The iPHB signal is generated according to the following rules:

– If LSPType[m] = L-LSP, the CoS information is equal to the CoS[m] while the DP information is decoded from the TC field according to [ITU-T G.8110.1] and the CoS[m].

– If LSPType[m] = E-LSP, the PHB information is decoded from the TC field according to the 1:1 mapping configured in the TC2PHBMapping[m].

NOTE – E-LSP and L-LSP are referred to in [ITU-T G.8110.1]

The CI_oPHB is generated according to the following rule:

– If QoSDecodingMode = A, the oPHB is equal to the generated iPHB.

– If QoSDecodingMode = B, the oPHB is equal to the received PHB.

When MI_Mode is set to mode 2, the TC/Label sink process simply inserts a TTL and a S field on to each MPLS traffic unit, with the values set to 254 and 0 respectively.

8.2.3 Label stack copy process

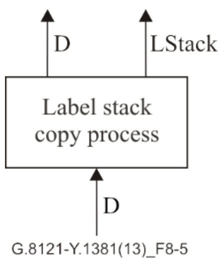


Figure 8-5 – Label stack copy process

Figure 8-5 shows Label stack copy process. It passes through the CI_D unchanged and copies from the CI_D traffic unit the complete label stack.

The LStack information is lost at the input of Server/MT and MT/MT source adaptation function and at the input of MT/Client sink adaptation function.

8.3 Queuing process

The Queuing process buffers received MPLS packets for output according to the CI_oPHB. Figure 8-6 shows Queuing process. The details of the Queuing process implementation are out of the scope of this Recommendation.

The Queuing process is also responsible for dropping packets if their rate at the MT_CI is higher than the <Srv>_AI_D can accommodate. Performance monitor counters are for further study.

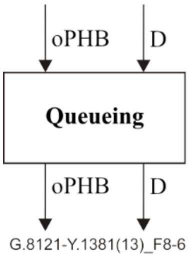


Figure 8-6 – Queuing process

8.4 MPLS-TP-specific GFP-F processes

8.4.1 MPLS-TP-specific GFP-F source processes

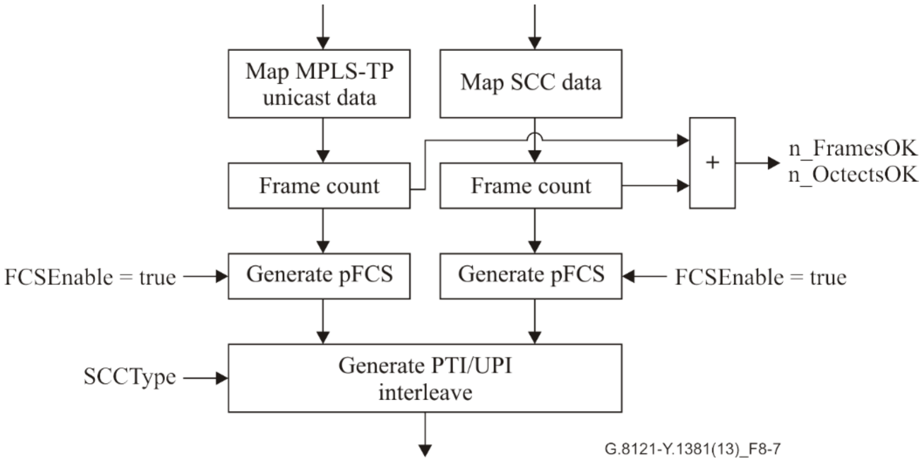


Figure 8-7 – MPLS-TP-specific GFP-F source process

Figure 8-7 shows the MPLS-TP-specific GFP-F source processes. These processes are performed on a packet-per-packet basis.

Mapping of MPLS-TP data: The MPLS-TP packet is inserted into the client payload information field of the generic framing procedure (GFP) frame as defined in clause 7.6 of [ITU-T G.7041]. One MPLS-TP packet results in one GFP frame.

Mapping of SCC data: The signalling communication channel (SCC) frame is inserted into the client payload information field of the GFP frame as defined in clause 7 of [ITU-T G.7041]. One SCC packet results in one GFP frame.

Frame count: It counts the number of frames (n_FramesOK) and of octets (n_OctetsOK) that passes through.

pFCS generation: See clause 8.5.4.1.1 of [ITU-T G.806]. GFP FCS is always enabled (FCSEnable=true).

Generate PTI and UPI, interleave: The payload type identifier (PTI) field of the GFP type header is set fixed to "000". The user payload identifier (UPI) field of the GFP type header is set to:

– the MPLS UPI (as defined in Table 6-3 of [ITU-T G.7041]), for frames coming from the Map MPLS-TP data process;

– the SCC UPI according to SCC type for frames coming from the Map SCC data process.

The frames are then interleaved to form a single stream.

NOTE – GFP client management frames are not defined for MPLS-TP over GFP-F mapping.

8.4.2 MPLS-TP-specific GFP-F sink processes

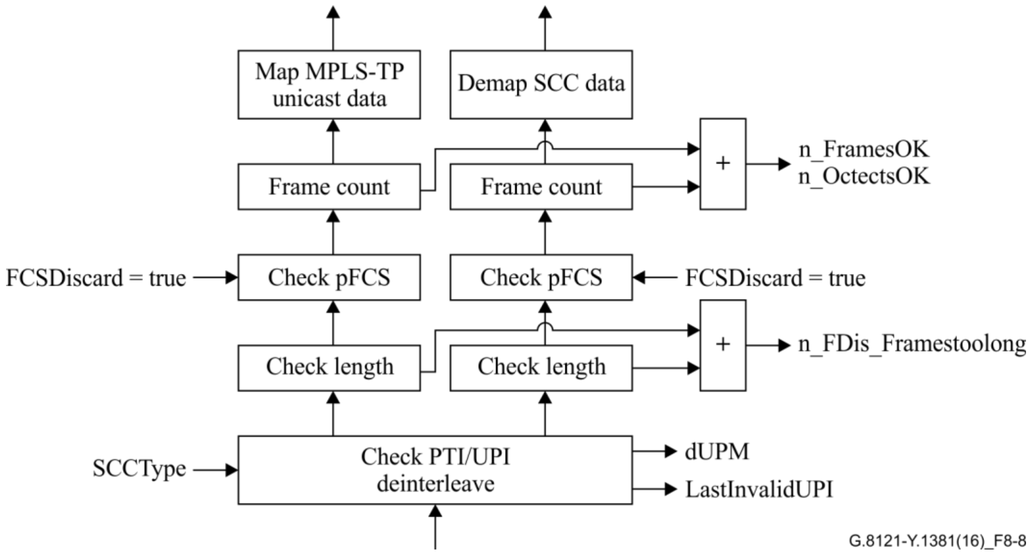


Figure 8-8 – MPLS-TP-specific GFP-F sink process

Figure 8-8 shows the MPLS-TP-specific GFP-F sink processes. These processes are performed on a packet-per-packet basis.

Check PTI and UPI, deinterleave: GFP frames with an accepted PTI (AcPTI, see clause 8.5.1.1 of [ITU-T G.806]) of "000" are client data frames. All GFP frames with an accepted PTI (AcPTI, see clause 8.5.1.1 of [ITU-T G.806]) value other than "000" shall be discarded.

The UPI of client data frames is checked to generate dUPM as follows:

– a "valid-UPI frame" is a frame with a UPI that equals either the MPLS UPI (as defined in Table 6-3 of [ITU-T G.7041]) or the SCC UPI according to SCCType. All other frames are "invalid-UPI frames".

– dUPM is raised as soon as one "invalid-UPI frame" is received.

– dUPM is cleared if no "invalid-UPI frames" have been received for the last Tclear seconds.

Tclear is for further study. If dUPM is active, the latest received invalid UPI is available at LastInvalidUPI. If dUPM is not active, LastInvalidUPI is "n/a".

The UPI of client data frames is further used to deinterleave the frames:

– "valid-UPI frames" with UPI equalling the MPLS UPI (as defined in Table 6-3 of [ITU-T G.7041]) are sent towards the "Demap MPLS-TP data" process.

– "valid-UPI frames" with UPI equalling the SCC UPI according to SCCType (as defined in Table 6-3 of [ITU-T G.7041]) are sent towards the "Demap SCC data" process.

– "invalid-UPI frames" are discarded.

GFP-F frame length: It checks whether the length of the GFP-F frame is allowed. Frames longer than GFP_Length bytes are dropped and counted (n_FramesTooLong).

NOTE – GFP_Length is for further study.

pFCS supervision: See clause 8.5.4.1.2 of [ITU-T G.806]. The discarding of errored frames is always enabled (FCSdiscard=true). If the accepted payload frame check sequence indication.

(PFI) is 0, the frame is dropped and counted (n_FDis_PFI).

Frame count: It counts the number of frames (n_FramesOK) and of octets (n_OctetsOK) that pass through.

Demapping of SCC data: The SCC packet is extracted from the client payload information field of the GFP frame as defined in clause 7 of [ITU-T G.7041]. One GFP frame results in one SCC frame.

Demapping of unicast MPLS-TP data: The MPLS-TP packet is extracted from the client payload information field of the GFP frame as defined in clause 7.6 of [ITU-T G.7041]. One GFP frame results in one MPLS-TP packet.

8.5 Control word (CW) processes

This function performs the control word (CW) processing as described in [IETF RFC 4448]. The CW is known as the common interworking indicators (CII) in [ITU-T Y.1415].

8.5.1 CW insertion process

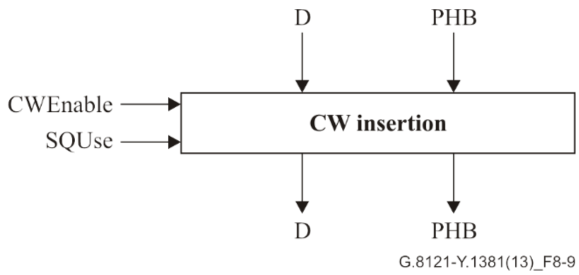


Figure 8-9 – CW insertion process

Figure 8-9 shows CW insertion process. This function should generate and insert the CW as described in [IETF RFC 4448] if the indication CWEnable is true. Otherwise, no insertion should be performed. If the indication SQUse is false, the sequence number field should be set at all zeroes.

8.5.2 CW extraction process

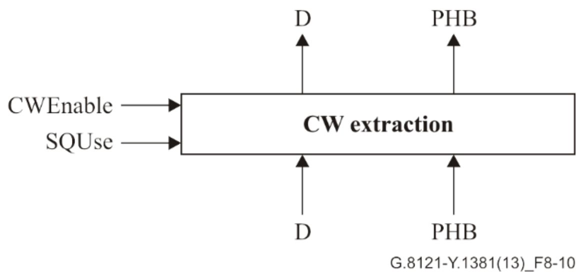


Figure 8-10 – CW extraction process

Figure 8-10 shows CW extraction process. This function should process and remove the CW as described in [IETF RFC 4448], if the indication CWEnable is true. In this case, if the indication SQUse is true, the sequence number field should be processed and out-of-sequence packets dropped (no reordering is performed by this process).

8.6 OAM related processes used by server adaptation functions

8.6.1 Selector process

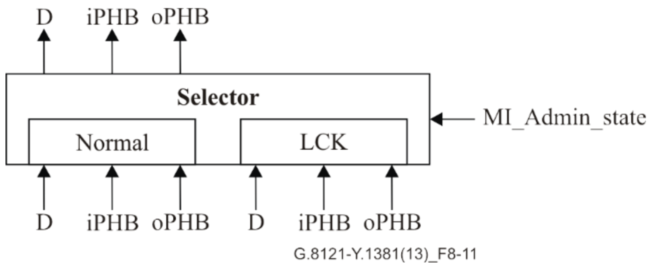


Figure 8-11 – Selector process

Figure 8-11 shows the Selector process symbol. The Selector process selects the valid signal from the input of the normal MT_CI signal or the MT_CI LCK signal (as generated by the LCK generation process in clause 8.6.3). The normal signal is blocked if MI_Admin_State is LOCKED. The behaviour is illustrated in Figure 8-12.

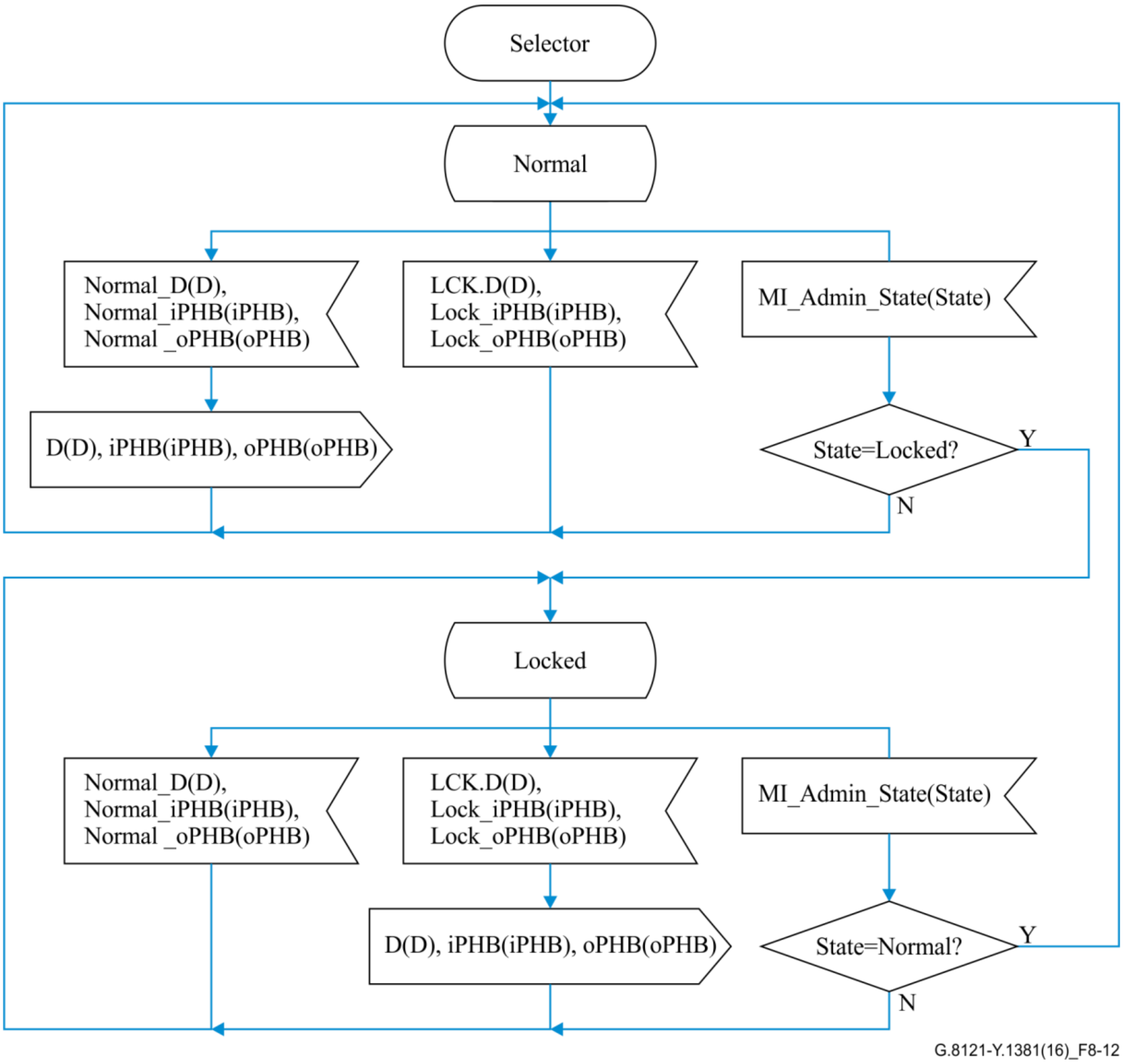


Figure 8-12 – Selector behaviour

8.6.2 AIS insert process

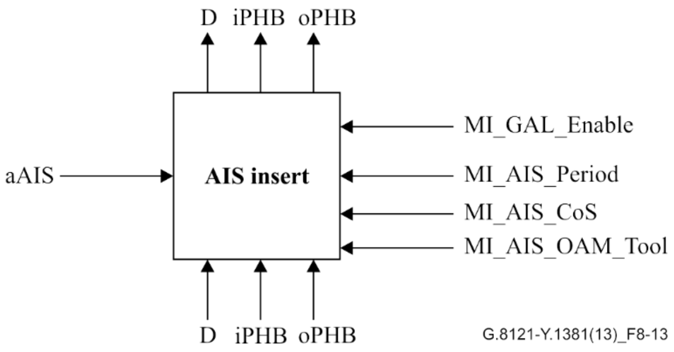


Figure 8-13 – AIS insert process

Figure 8-13 shows the AIS insert process symbol. The generated AIS traffic units are inserted in the incoming stream, i.e., the output stream contains the incoming traffic units and the generated AIS traffic units.

The period between consecutive AIS traffic units is determined by the MI_AIS_Period parameter. The format of the AIS traffic units is defined by the MI_AIS_OAM_Tool parameter. The generated AIS traffic units are G-ACh encapsulated as described in clause 8.1 which includes GAL or not depending on MI_GAL_Enable.

The value of the MT_CI_ iPHB and MT_CI_oPHB signals associated with the generated AIS traffic units is the PHB with the lowest drop precedence within the CoS defined by the MI_AIS_CoS input parameter. As described in [IETF RFC 6371], AIS packets are transmitted with the "minimum loss probability PHB".

8.6.3 LCK generation process

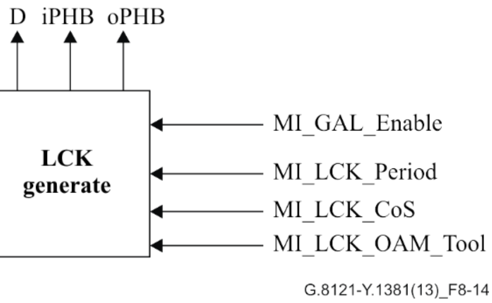


Figure 8-14 – LCK generation process

Figure 8-14 shows the LCK[3] generation process symbol. The LCK generation process generates MT_CI traffic units where the MT_CI_D signal contains the LCK signal. Figure 8-15 defines the behaviour of the LCK generation process.

3 IETF uses the term LKR for this function.

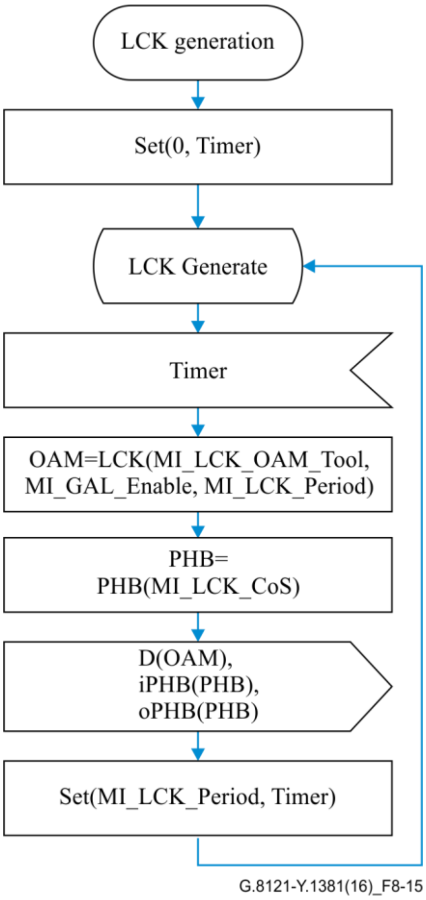


Figure 8-15 – LCK generation behaviour

The LCK generation process continuously generates LCK traffic units. The period between consecutive LCK traffic units is determined by the MI_LCK_Period parameter.

The LCK(LCK_OAM_Tool, GAL_Enable, Period) function generates a LCK traffic unit, whose format is defined by the LCK_OAM_Tool parameter, that encodes the period information defined by the value of the Period parameter. The generated traffic unit is G-ACh encapsulated, as described in clause 8.1, with or without the GAL depending on the GAL_Enable parameter.

The value of the MT_CI_iPHB and MT_CI_oPHB signal associated with the generated LCK traffic units is the PHB with the lowest drop precedence within the CoS defined by the MI_LCK_CoS input parameter. The PHB(MI_LCK_CoS) function generates such PHB information. As described in [IETF RFC 6371], LCK packets are transmitted with the "minimum loss probability PHB".

8.7 OAM related processes used by adaptation functions

8.7.1 MCC and SCC mapping and demapping

As defined in [ITU-T G.7712], an embedded communication channel (ECC) provides a logical operations channel between network elements (NEs) that can be utilized by various applications. A managementmaintenance communication channel (MCC) is an ECC dedicated for management plane communications. A signalling communication channel (SCC) is an ECC dedicated for control plane communications.

The MCC mapping and demapping processes are provided to support the MT to MCC adaptation function for accessing the MCC. The SCC mapping and demapping processes are provided to support the MT to SCC adaptation function for accessing to the SCC. The mapping and demapping processes for MCC is very similar to that of the SCC. In the following description given in this clause and in clause 8.7.2, the term ECC will be used, which applies to both MCC and SCC.

8.7.1.1 ECC mapping

The ECC mapping process is associated with the MT/MCC_A_So and MT/SCC_A_So functions, which are described in clauses 10.2.2.1 and 10.2.1.1 respectively.

This process shall map the incoming ECC packet into G-ACh encapsulated ECC traffic unit (i.e., a MT_AI_D traffic units carrying an ECC packet).

The ECC traffic units generated by this process are encapsulated into the G_ACh, as defined in [IETF RFC 5718], using or not the GAL depending on the MI_GAL_Enable configuration parameters. The value of the MT_AI_PHB associated with the generated ECC traffic units is defined by the MI_ECC_CoS input parameter.

8.7.1.2 ECC demapping

The ECC Demapping process is associated with the MT/MCC_A_Sk and MT/SCC_A_Sk functions, which are described in clauses 10.2.2.2 and 10.2.1.2 respectively.

This process shall extract the ECC packet from the G-ACh encapsulated ECC traffic unit (i.e., MT_AI_D traffic units carrying ECC packets).

The criteria for selecting ECC traffic units are based on the values of the fields within the MT_AI_D signal:

– GAL is included to the MT_AI_D if GAL usage is enabled via MI_GAL_Enable.

– The Channel type of G-ACh indicates a MCC packet (in MT/MCC_A_Sk) or a SCC packet (in MT/SCC_A_Sk), as defined in [IETF RFC 5718].

8.7.2 APS insert and extract processes

Figure 8-16 shows a protocol-neutral abstract model of the different processes inside the MEPs and the MIPs that are involved in the automatic protection switching (APS)[4] function.

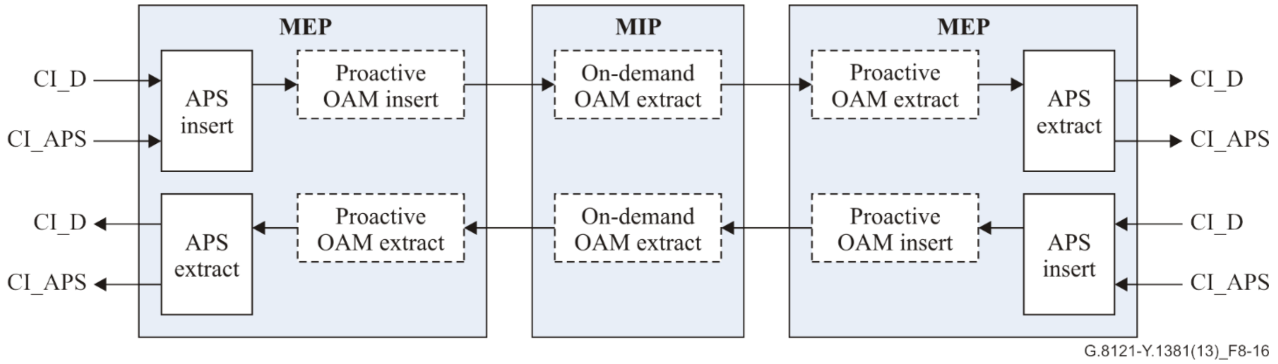


Figure 8-16 – Overview of the processes involved with APS function

APS insert and extract processes are located in MT/MT_Adaptation function. The CI_APS signal contains the APS specific information as defined in clause 8.2 of [ITU-T G.8131]. APS traffic units are inserted into and extracted from the stream of MT_CI_D traffic units.

[4] [ITU-T G.8131] conventionally uses the term APC for this function.

8.7.2.1 APS insert process

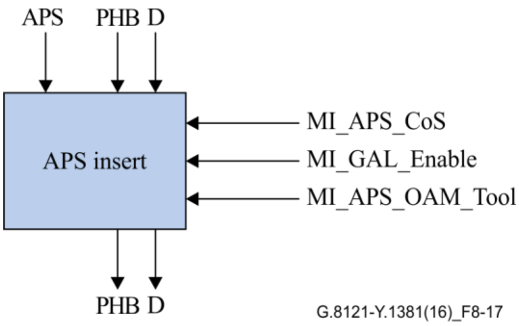


Figure 8-17 – APS insert process

Figure 8-17 shows the APS insert process and Figure 8-18 illustrates the behaviour. The resulting APS traffic unit is inserted into the stream of incoming traffic units, i.e., the outgoing stream consists of the incoming traffic units and the inserted APS traffic units. The MT_CI_APS signal contains the APS specific information as defined in [ITU-T G.8131].

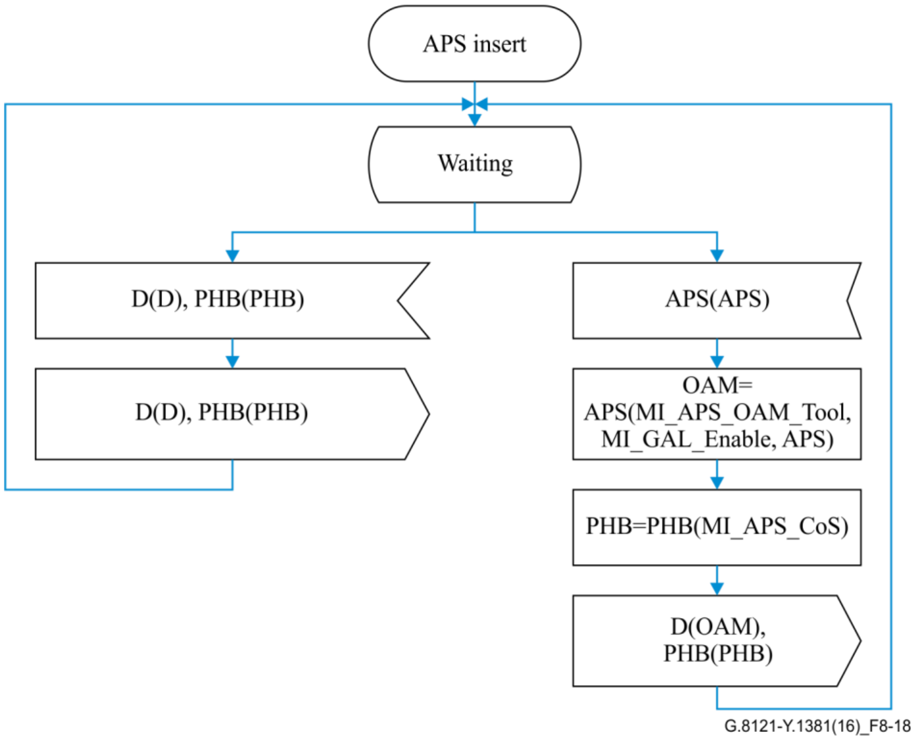


Figure 8-18 – APS insert behaviour

The APS(APS_OAM_Tool, GAL_Enable, APS) function generates an APS traffic unit, whose format is defined by the APS_OAM_Tool parameter, that encodes the APS information defined by the value of the APS parameter. The generated traffic unit is G-ACh encapsulated, as described in clause 8.1, with or without the GAL depending on the GAL_Enable parameter.

The value of the MT_AI_PHB signals associated with the generated APS traffic units are the PHB with the lowest drop precedence within the CoS defined by the MI_APS_CoS input parameter. The PHB(MI_APS_CoS) function generates such PHB information.

8.7.2.2 APS extract process

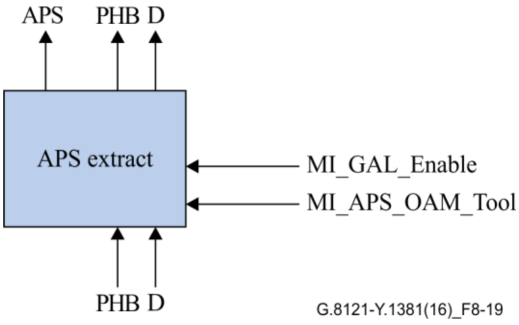


Figure 8-19 – APS extract process

The APS extract process, shown in Figure 8-19, extracts MT_AI_APS signals from the incoming stream of MT_AI traffic units.

The MT_AI_APS is the APS specific information as described in [ITU-T G.8131] contained in the received traffic unit. All other traffic units will be transparently forwarded.

The criteria for filtering are based on the values of the fields within the MT_AI_D signal:

– S bit (1) and GAL included to the MT_AI_D if GAL usage is enabled via MI_GAL_Enable.

– OAM type that is defined in channel type of G-ACh indicates APS

This is illustrated in Figure 8-20. The function APS(D) extracts the APS specific information from the received traffic unit.

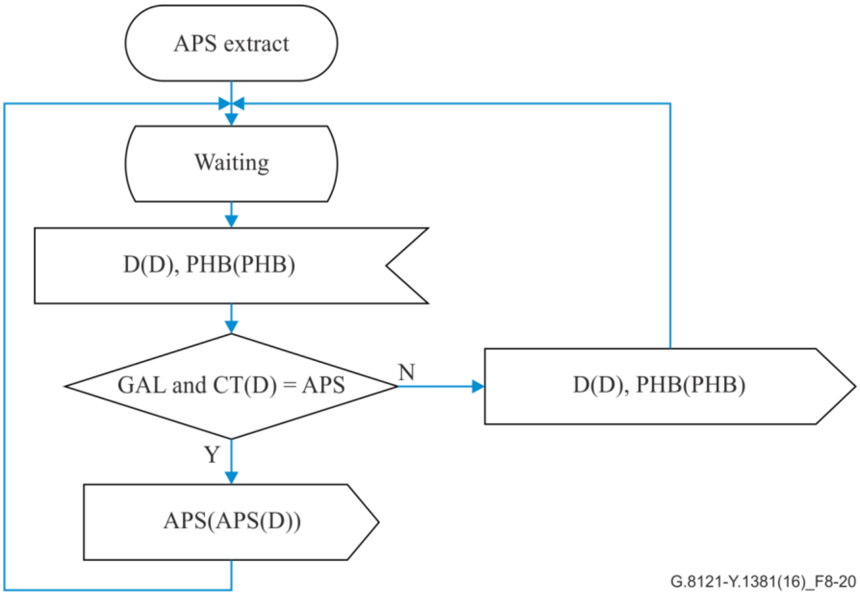


Figure 8-20 – APS extract behaviour

8.7.3 CSF insert and extract processes

Figure 8-21 shows the different processes inside MEPs and MIPs that are involved in the CSF [5] protocol.

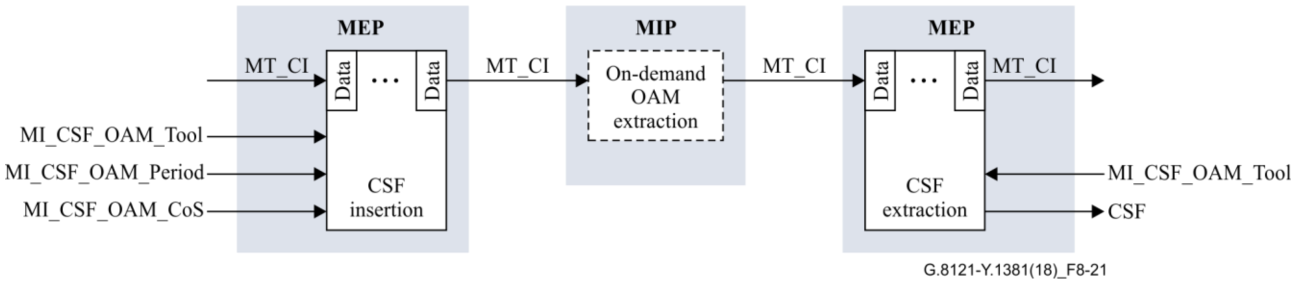


Figure 8-21 – Overview of processes involved with CSF protocol

The MPLS-TP client signal fail function (MT-CSF) is used by a MEP to propagate to a peer MEP the detection of a failure or defect event in a MPLS-TP client signal when the client itself does not support appropriate fault or defect detection or propagation mechanisms, such as MT-CC or MT-AIS. The MT-CSF messages propagate in the direction from MPLS-TP MEP function detecting the failure or defect event to the MPLS-TP sink-adaptation function associated with the peer MEP.

MT-CSF generation is located at the MT/Client_A_So to insert CSF traffic unit and ProActive OAM insertion is located at the MT_TT.

8.7.3.1 CSF Insert process

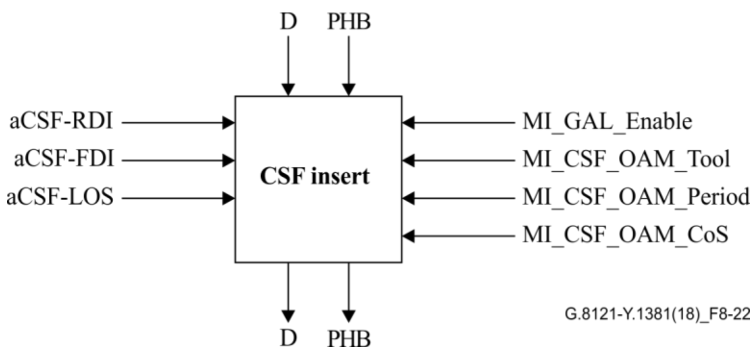


Figure 8-22 – CSF insert process

The CSF insert process is located at the MT/Client_A_So as a part of CSF generation. Figure 8-22 shows the CSF insert process symbol.

The behaviour is specified in [ITU-T G.8121.1] and [ITU-T G.8121.2] depending on the MI_CSF_OAM_Tool parameter. The format of the CSF traffic units is specified in [ITU-T G.8113.1] and [ITU-T G.8113.2] depending on the MI_CSF_OAM_Tool parameter.

5 IETF uses the term "Client Failure Indication" for this function in [IETF RFC 6371].

8.7.3.2 CSF extract process

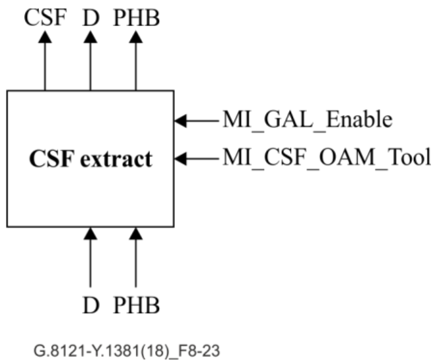


Figure 8-23 – CSF extract process

The CSF extract process is located at MT/Client_A_sk and extracts MT-CSF from MI_AI_D. Figure 8-23 shows the CSF extract process symbol.

AI_CSF signal is extracted from the incoming stream of MT_AI traffic units. Figure 8-23 shows the CSF extract process symbol. The behaviour and criteria for filtering are specified in [ITU-T G.8121.1] and [ITU-T G.8121.2] depending on the MI_CSF_OAM_Tool parameter.

8.8 Proactive and on-demand OAM related processes

As described in [IETF RFC 6371], OAM functions are categorized as proactive and on-demand and these OAM functionalities provide the different interfaces.

OAM functions can be also categorised as single-ended and dual-ended. Single-ended functions are those in which an initiating MEP sends OAM PDUs to a target MEP, which processes it and sends a response OAM PDU back to the initiating MEP. The results of the function are available only on the initiating MEP. Dual-ended functions are those in which an initiating MEP sends OAM PDUs to a target MEP, which processes it and does not send a response. The results of the function are available only on the target MEP. Dual-ended functions are typically deployed in pairs, one in each direction.

Figure 8-24 shows an OAM protocol-neutral abstract model of the different processes inside MEPs and MIPs that are involved in performing single-ended proactive or on-demand OAM functions. In the case of dual-ended functions, the model is equivalent to the top half of the diagram only, and the results are reported by the OAM sink control process on the target MEP.

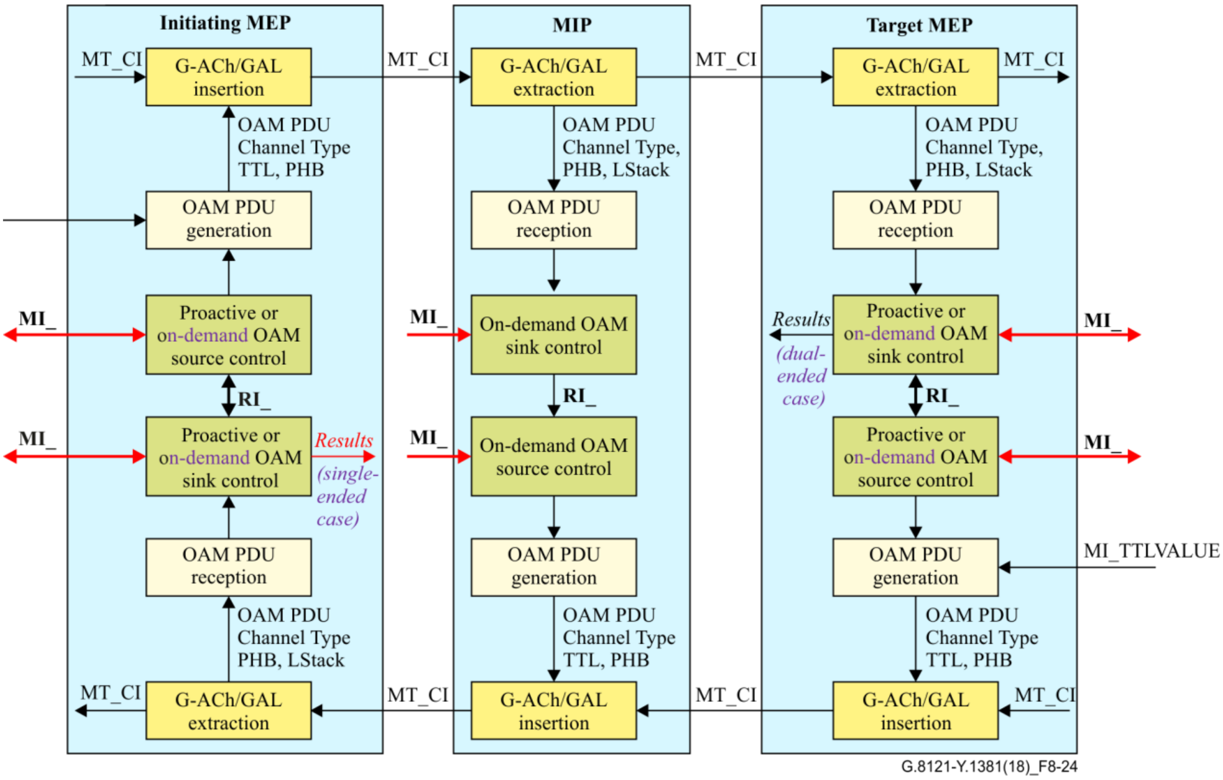


Figure 8-24 – Overview of the processes involved with proactive or on-demand OAM functions

NOTE – The MT_CI signals at the input of the G-ACh insertion process and at the output of the G-ACh extraction process are not input/output signals of the Initiation/Target MEPs but signals which are internal to these MEPs.

The processes shown in Figure 8-24 are described further below, with the exception of the G-ACh insertion and extraction processes which are described in clause 8.1.

The relevant management information (MI_) and remote information (RI_) used by these processes depend on the OAM function to be performed and is defined in the following clauses.

The detailed specification of all the OAM processes, including further process decomposition and the interface between them, is OAM protocol-specific and therefore outside the scope of this Recommendation.

OAM control processes

The four OAM control processes (i.e., the Proactive OAM source control process, Proactive OAM sink control process, On-demand OAM source control process and On-demand OAM sink control process) perform all the OAM control procedures (e.g., they maintain the necessary state machine) that are required for a specific OAM protocol. The proactive OAM source and sink control processes operate within the MT_TT_So and MT_TT_Sk atomic functions respectively.

Similarly, the On-demand OAM source and sink control processes operate within the MTDe_TT_So or MTDi_TT_So and the MTDe_TT_Sk or MTDi_TT_Sk atomic functions respectively.

All four processes consist of a number of OAM protocol-specific control sub-processes, each relating to a different OAM function. The details of these sub-processes are outside the scope of this Recommendation

The OAM source control process within the initiating MEP (Proactive or On-demand as appropriate) requests the OAM PDU generation process to generate OAM Request PDUs towards the target MEP on the basis of the local state machine and the relevant management information (MI_). This supports both single-ended and dual-ended OAM transactions.

In the case of a dual-ended OAM transaction, the appropriate OAM sink control process within the target MEP (Proactive or On-demand) or within the target MIP (On-demand only) reports the dual-ended OAM results on the basis of the OAM Request PDUs received by the OAM PDU Reception process.

In the case of single-ended OAM transactions, the following actions are taken:

– The OAM sink control process within the target MEP or MIP provides the local OAM source control process the relevant remote information (RI_) to generate a reply to the OAM Request PDU received by the local PDU reception process.
– The OAM source control process within the target MEP or MIP requests the OAM PDU generation process to generate OAM Reply PDUs towards the initiating MEP based on the information it receives from the local OAM sink control process via the relevant remote information (RI_).
– The OAM sink control process within the initiating MEP reports the unidirectional or bidirectional OAM results based on the OAM Reply PDUs received by the local OAM PDU reception process

OAM PDU generation process

The OAM PDU generation process builds, when instructed by its control process, the required OAM PDU and passes it to the G-ACh insertion process, defined in clause 8.1, for insertion within the MPLS-TP characteristic information (CI) traffic flow. It also passes the following information elements that are required by the G-ACh insertion process: the PHB associated to the OAM packet (on the basis of the CoS instruction received by the control process); the ACH channel type that identifies the OAM PDU and the TTL value which it is either the TTL distance to a MIP (for OAM PDUs targeted to a MIP and properly requested by the control process) or the default value as configured via MI_TTLValue.

The OAM PDU generation process consists of a number of OAM protocol-specific PDU generation sub-processes (one for each PDU type) and a sub-process that multiplexes all the PDUs generated by these OAM protocol-specific PDU generation sub-processes into a single stream of OAM PDUs, which is sent to the G-ACh insertion process along with the appropriate ACH channel type. The details of these sub-processes are outside the scope of this Recommendation.

OAM PDU reception process

The OAM PDU reception process receives an OAM PDU, together with the ACH channel type value identifying the PDU type, the associated PHB, and the label stack (LStack) data, from the G-ACh process and passes the relevant information to its control process.

The OAM PDU reception process consists of a number of OAM protocol-specific PDU reception sub-processes (one for each PDU type) and a sub-process that demultiplexes OAM PDUs received from the G-ACh extraction process towards these OAM protocol-specific PDU reception sub-processes based on the ACH channel type. The details of these sub-processes are outside the scope of this Recommendation.

8.8.1 Proactive continuity check or connectivity verification (CC/CV)

As described in [IETF RFC 6371], both continuity check (CC) and connectivity verification (CV) OAM functions are based on the proactive generation of OAM packets by the source MEP that are processed by the peer sink MEP(s).

The source MEP generates CC/CV OAM packets if it is enabled via management information (MI). As described in [IETF RFC 6371], the CC/CV OAM packets are generated at a regular rate which is configured by the operator via the MI_CC_Period. These packets are also transmitted using PHB which is configured via MI_CC_CoS (and that is typically the "minimum loss probability PHB").

In order to perform connectivity verification (CV), the generated CC/CV packets also include a globally unique source MEP identifier: the transmitted value is configured via protocol-specific management information on the source MEP while the expected value is configured via different protocol-specific management information on the sink MEP.

The sink MEP always processes received CC/CV OAM packets and detects the following CC/CV defects, as defined in clause 6.1:

– dLOC

– dUNC

– dMMG

– dUNM

– dUNP

CC/CV OAM packets pass transparently through MIPs as described in [IETF RFC 6371].

The EMF can retrieve from the sink MEP the latest CC/CV OAM packet which caused a defect condition via the MI_Get_SvdCC command: the CC/CV OAM packet is returned to the EMF via the MI_SvdCC.

8.8.2 Remote defect indication (RDI)

As described in [IETF RFC 6371], in case of co-routed and associated bidirectional transport paths, RDI is associated with proactive CC/CV, and the RDI indicator can be piggy-backed onto the CC/CV packet.

RDI information is carried in the CC/CV packets based upon the RI_CC/CV_RDI input. It is extracted in the CC/CV reception process.

In case of unidirectional transport paths, the RDI related OAM process is for further study.

8.8.3 On-demand connectivity verification (CV)

As described in [IETF RFC 6371], on-demand CV OAM functions are based on the on-demand generation of OAM packets by the source MEP, that are processed and responded to by the peer sink MIP(s) or MEP(s).

The source MEP generates on-demand CV OAM packets when requested via protocol-specific MI signals. The results of the on-demand CV operation are returned by the source MEP using additional protocol-specific management information.

8.8.4 Proactive packet loss measurement (LMp)

As described in [IETF RFC 6371], proactive loss measurement (LM) is performed by periodically sending LM OAM packets from the initiating MEP to the target MEP and by receiving LM OAM packets from the target MEP on a co-routed bidirectional connection. Each MEP performs measurements of its transmitted and received user data packets (TxFCl and RxFCl). These measurements are then correlated in real time with the target MEP in the ME to derive the impact of packet loss on a number of performance metrics for the ME in the MEG.

As described in [IETF RFC 6374], there are two types of LM mechanism. One is to directly measure dataplane packet loss by counting packets transmitted and received. This is called LM or direct LM. The other is to measure the approximate dataplane loss level by counting synthetic packets transmitted and received. This is called synthetic loss measurement (SLM) or inferred loss measurement.

For single-ended measurement:

- The initiating MEP generates proactive LM OAM Request packets if MI_LMp_Enable is true. These packets are generated at the rate configured via the MI_LMp_Period and, as described in [IETF RFC 6371], with the PHB configured via MI_LMp_CoS that yields the lowest drop precedence within the measured PHB scheduling class, in order to maximize reliability of measurement within the traffic class. The local value of transmitted user data packets (TxFCl) is inserted within the LM OAM packet by the OAM PDU generation process.
- The target MEP replies to the LM OAM packets if it is enabled. The local value of the received user data packets (RxFCl) at the time the proactive LM OAM Request packet has been received is passed by OAM PDU reception process to the proactive OAM sink control process, then to the proactive OAM source control process (via RI_OAM_Info) and inserted by the OAM PDU generation process within the transmitted proactive LM OAM Reply: the actual behaviour on how this information is passed is OAM protocol specific and therefore outside the scope of this Recommendation. The OAM PDU generation process also inserts the local value of the transmitted user data packets (TxFCl) in the reverse direction within the transmitted proactive LM OAM Reply.
- The initiating MEP processes the received proactive LM OAM Reply packet, together with the local value of the received used data packets (RxFCl) in the reverse direction at the time this OAM packet is received, as passed by the OAM PDU reception process, and generates LM results.

Depending on the LMp OAM tool that it is used, the LM results can be either calculated by the proactive OAM sink control process or by the proactive OAM source control process. In the latter case, the proactive OAM sink control process passes the required information in the received LM OAM Reply to the proactive OAM source control process via the RI_LMRp and receives the LM results back via the RI_LMp_Result. In both cases, the proactive OAM sink control process passes the LM results to the relevant performance monitoring processes within the MT_TT_Sk atomic function for reporting to the EMF that is described in [b-ITU-T G.8151].

NOTE – The behaviour is the same when synthetic proactive (SLp) LM is used. In this Recommendation, the term SLp is used to address the management and remote information (MI and RI) for synthetic loss measurement.

For dual-ended measurement:

- The initiating MEP generates proactive LM OAM Request packets if MI_1LMp_Enable is true. These packets are generated at the rate configured via the MI_LMp_Period and, as described in [IETF RFC 6371], with the PHB configured via MI_LMp_CoS that yields the lowest drop precedence within the measured PHB scheduling class, in order to maximize reliability of measurement within the traffic class. The local value of transmitted user data packets (TxFCl) is inserted within the LM OAM packet by the OAM PDU generation process.
- The target MEP receives LM OAM packets if it is enabled to do so via management information. The local value of the received user data packets (RxFCl) at the time the proactive LM OAM Request packet has been received is passed by OAM PDU reception process to the proactive OAM sink control process and generates LM results.

NOTE – The behaviour is the same when dual-ended synthetic proactive (1SLp) LM is used. In this Recommendation, the term 1SLp is used to address the management and remote information (MI and RI) for synthetic loss measurement.

Proactive LM OAM packets pass transparently through MIPs as described in [IETF RFC 6371].

8.8.5 On-demand packet loss measurement (LMo)

As described in [IETF RFC 6371], on-demand LM is performed by the command that sends LM OAM packets from the initiating MEP to the target MEP and by receiving LM OAM packets from the target MEP on a co-routed bidirectional connection. The initiating MEP performs measurements of its transmitted and received user data packets (TxFCl and RxFCl). These measurements are then correlated in real time with values received from the target MEP in the ME to derive the impact of packet loss for the ME in the MEG.

As described in [IETF RFC 6374], there are two types of LM mechanism. One is to directly measure data-plane packet loss by counting packets transmitted and received. This is called LM or direct LM. The other is to measure the approximate dataplane loss level by counting synthetic packets transmitted and received. This is called synthetic loss measurement (SLM) or inferred loss measurement.

For single-ended measurement:

– The initiating MEP generates on-demand LM OAM Request packets when enabled via management information. These packets are generated with the PHB configured via management information that yields the lowest drop precedence within the measured PHB scheduling class, in order to maximize reliability of measurement within the traffic class. The local value of transmitted user data packets (TxFCl) is inserted within the LM OAM packet by the OAM PDU generation process.
– The target MEP replies to the LM OAM packets if enabled via management information. The local value of the received user data packets (RxFCl) at the time the on-demand LM OAM Request packet has been received is passed by OAM PDU reception process to the on-demand OAM sink control process, then to the on-demand OAM source control process (via RI_OAM_Info) and inserted by the OAM PDU generation process within the transmitted on-demand LM OAM Reply: the actual behaviour on how this information is passed is OAM protocol specific and therefore outside the scope of this Recommendation. The OAM PDU generation process also inserts the local value of the transmitted user data packets (TxFCl) in the reverse direction within the transmitted on-demand LM OAM Reply.
– The initiating MEP processes the received on-demand LM OAM Reply packet, together with the local value of the received used data packets (RxFCl) in the reverse direction at the time this OAM packet is received, as passed by the OAM PDU reception process, and generates LM results.
– Depending on the LMo OAM tool that it is used, the LM results can be either calculated by the on-demand OAM sink control process or by the on-demand OAM source control process. In both cases, the LM results are reported to EMF by the MTDe_TT_So using the MI_LMo_Result(N_TF,N_LF,F_TF,F_LF) signal, containing the near-end transmitted packets, near-end lost packets, far-end transmitted packets and far-end lost packets.

NOTE – The behaviour is same when synthetic on-demand (SLo) LM is used. In this Recommendation, the term SLo is used to address the management and remote information (MI and RI) for synthetic loss measurement.

For dual-ended measurement:

– The initiating MEP generates on-demand LM OAM Request packets when enabled via management information. These packets are generated at the rate configured via the MI_LMo_Period and, as described in [IETF RFC 6371], with the PHB configured via MI_LMo_CoS that yields the lowest drop precedence within the measured PHB scheduling class, in order to maximize reliability of measurement within the traffic class. The local value of transmitted user data packets (TxFCl) is inserted within the LM OAM packet by the OAM PDU generation process.

– The target MEP receives LM OAM packets when enabled via management information. The local value of the received user data packets (RxFCl) at the time the on-demand LM OAM Request packet has been received is passed by OAM PDU reception process to the on-demand OAM sink control process and generates LM results.

On demand LM OAM packets pass transparently through MIPs as described in [IETF RFC 6371].

8.8.6 Proactive packet delay measurement (DMp)

As described in [IETF RFC 6371], proactive delay measurement (DM) is performed by periodically sending DM OAM packets from the initiating MEP to the target MEP and by receiving DM OAM packets from the target MEP on a co-routed bidirectional connection. Each MEP records its transmitted and received timestamps. The timestamps from the initiating and target MEPs are then correlated to derive a number of performance metrics relating to delay for the ME in the MEG.

For single-ended measurement:

– The initiating MEP generates proactive DM OAM Request packets if MI_DMp_Enable is true. These packets are generated at the rate configured via the MI_DMp_Period and, as described in [IETF RFC 6371], with the PHB configured via MI_DMp_CoS that yields the lowest drop precedence within the measured PHB scheduling class, in order to maximize reliability of measurement within the traffic class. The local value of transmitted timestamp (TxTimeStampf) is inserted within the DM OAM packet by the OAM PDU generation process.

– The target MEP replies to the DM OAM packets if enabled by management information. The local value of the received timestamp (RxTimeStampl) at the time the proactive DM OAM Request packet has been received is passed by OAM PDU reception process to the proactive OAM sink control process, then to the proactive OAM source control process (via RI_OAM_Info) and inserted by the OAM PDU generation process within the transmitted proactive DM OAM Reply: the actual behaviour on how this information is passed is OAM protocol specific and therefore outside the scope of this Recommendation. The OAM PDU generation process also inserts the local value of the transmitted timestamp (TxTimeStampf) in the reverse direction within the transmitted proactive DM OAM Reply.

– The initiating MEP processes the received proactive DM OAM Reply packet, together with the local value of the received timestamp (RxTimeStampl) in the reverse direction at the time this OAM packet is received, as passed by the OAM PDU reception process, and generates DM results.

– Depending on the DMp OAM tool that it is used, the DM results can be either calculated by the proactive OAM sink control process or by the proactive OAM source control process. In the latter case, the proactive OAM sink control process passes the required information in the received DM OAM Reply to the proactive OAM source control process via the RI_DMRp and receives the DM results back via the RI_DMp_Result. In both cases, the proactive OAM sink control process passes the DM results to the relevant performance monitoring processes within the MT_TT_Sk atomic function for reporting to the EMF.

For dual-ended measurement:

– The initiating MEP generates proactive DM OAM Request packets if MI_1DMp_Enable is true. These packets are generated at the rate configured via the MI_DMp_Period and, as described in [IETF RFC 6371], with the PHB configured via MI_DMp_CoS that yields the lowest drop precedence within the measured PHB scheduling class, in order to maximize reliability of measurement within the traffic class. The local value of transmitted timestamp (TxTimeStampl) is inserted within the DM OAM packet by the OAM PDU generation process.

− The target MEP receives DM OAM packets if enabled by management information. The local value of the received timestamp (RxTimeStampl) at the time the proactive DM OAM Request packet has been received is passed by OAM PDU reception process to the proactive OAM sink control process which generates DM results.

Proactive DM OAM packets pass transparently through MIPs as described in [IETF RFC 6371].

8.8.7 On-demand packet delay measurement (DMo)

As described in [IETF RFC 6371], on-demand DM is performed by the command that sends DM OAM packets from the initiating MEP to the target MEP and by receiving DM OAM packets from the target MEP on a co-routed bidirectional connection. Each MEP records its transmitted and received timestamps. The timestamps from the initiating and target MEPs are then correlated to derive a number of performance metrics relating to delay for the ME in the MEG.

For single-ended measurement:

− The initiating MEP generates on-demand DM OAM Request packets if MI_DMo_Enable is true. These packets are generated with the PHB configured via MI_DMo_CoS that yields the lowest drop precedence within the measured PHB scheduling class, in order to maximize reliability of measurement within the traffic class. The local value of transmitted timestamp (TxTimeStampf) is inserted within the DM OAM packet by the OAM PDU generation process.

− The target MEP replies to the PM OAM packets if the MI_PMo_Enable is true. The local value of the received timestamp (RxTimeStampl) at the time the on-demand PM OAM Request packet has been received is passed by OAM PDU reception process to the on-demand OAM sink control process, then to the on-demand OAM source control process (via RI_OAM_Info) and inserted by the OAM PDU generation process within the transmitted on-demand DM OAM Reply: the actual behaviour on how this information is passed is OAM protocol specific and therefore outside the scope of this Recommendation.

− The initiating MEP processes the received on-demand DM OAM Reply packet, together with the local value of the received timestamp (RxTimestampl) in the reverse direction at the time this OAM packet is received, as passed by the OAM PDU reception process, and generates DM results.

− Depending on the DMo OAM tool that it is used, the DM results can be either calculated by the on-demand OAM sink control process or by the on-demand OAM source control process. In both cases, the DM results are reported to EMF by the MTDe_TT_So using the MI_DMo_Result(B_FD[],F_FD[],N_FD[]) signal, containing the count of measurements, and arrays of round-trip frame delay, far-end frame delay and near-end frame delay measurements.

For dual-ended measurement:

− The initiating MEP generates on-demand DM OAM Request packets if MI_DMo_Enable is true. These packets are generated at the rate configured via the MI_DMo_Period and, as described in [IETF RFC 6371], with the PHB configured via MI_DMo_CoS that yields the lowest drop precedence within the measured PHB scheduling class, in order to maximize reliability of measurement within the traffic class. The local value of transmitted Timestamp (TxTimestampf) is inserted within the DM OAM packet by the OAM PDU Generation process.

− The target MEP receives DM OAM packets if the MI_DMo_Enable is true. The local value of the received timestamp (RxTimestampl) at the time the on-demand DM OAM Request packet has been received is passed by OAM PDU reception process to the on-demand OAM sink control process and generates DM results.

On-demand DM OAM packets pass transparently through MIPs as described in [IETF RFC 6371].

8.8.8 Throughput test

For single-ended throughput test:

− For further study.

For dual-ended throughput test:

− As described in [IETF RFC 6371], out of service on-demand throughput estimation can be performed by sending OAM test packets at an increasing rate (up to the theoretical maximum), computing the percentage of OAM test packets received and reporting the rate at which OAM test packets begin to drop. In general, this rate is dependent on the OAM test packet size.

− The source MEP starts generating test packets when requested via protocol-specific management information and continues generating these packets at the configured period until requested to stop; at this time the number of sent packets is reported via protocol-specific management information.

− The sink MEP, when enabled via protocol-specific management information, starts processing the received OAM test packets until the test is terminated; at this time, the calculated test results are reported.

8.8.9 Route tracing (RT)

For further study.

8.8.10 LCK/AIS reception

As described in [IETF RFC 6371], when a MEP detects a signal fail condition or is locked, it may transmit AIS or lock messages at the client layer, respectively. These are transmitted by the AIS insert process (see clause 8.6.2) or the LCK generation process (see clause 8.6.3), and are received by the LCK/AIS reception sub-process in the Proactive OAM sink control process. The receipt of AIS or lock message triggers the dAIS or dLCK defects respectively, as described in clause 6.1.

8.8.11 Lock instruct processes

As described in [IETF RFC 6371], when a MEP is administratively locked, it puts the local MPLS-TP trail into a locked state and, if enabled by the MI_Lock_Instruct_Enable, it also starts transmitting a lock instruct message to its peer MEP. The locking of the trail at the local MEP is performed by the Selector process defined in clause 8.6.1; the transmission of lock instruct (LKI) messages is performed by the On-demand OAM process.

On receiving a LKI message, the peer MEP must also lock the path.

The lock instruct message is received by the On-demand OAM process and the request is signalled to the EMF via MI_Admin_State_Request. The EMF should then combine this remote request with any local request from the user, and set MI_Admin_State accordingly in the corresponding MT/MT_A_So and MT/MT_A_Sk processes.

Figure 8-25 illustrates how setting MI_Admin_State on the local MEP both locks the local Selector processes (the local selector in the MT/MT_A_Sk is not shown), and triggers the generation of LKI messages to the remote MEP that cause the two remote Selector processes to also be locked.

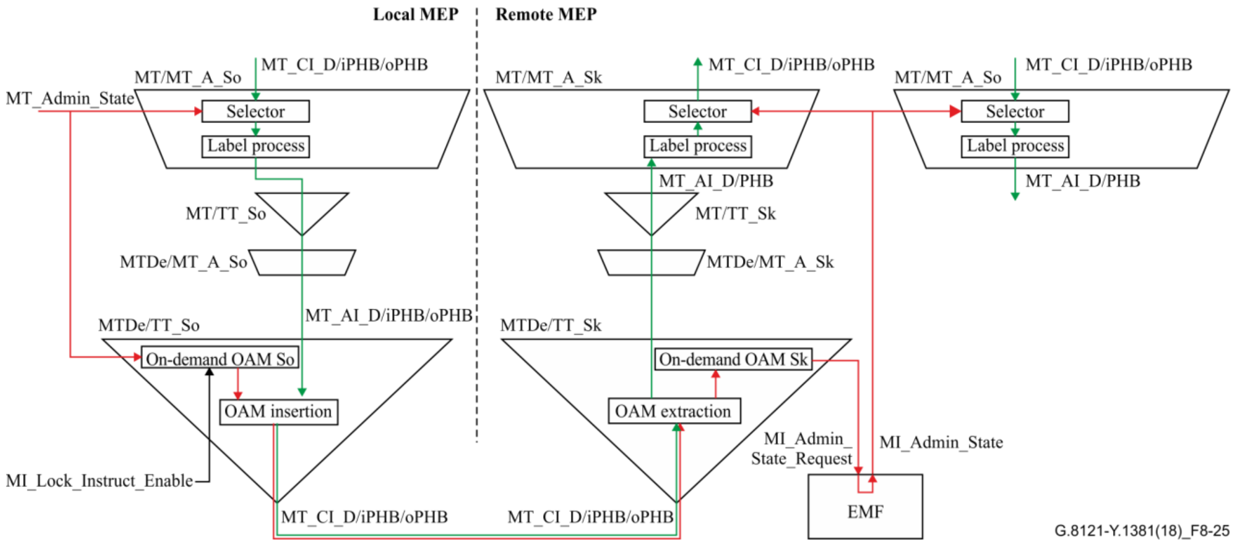


Figure 8-25 – Overview of the processes involved with lock instruct

NOTE 1 – It is important that the remote MEP is locked as soon as possible after receiving the first lock instruct message; therefore the EMF must handle the MI_Admin_State_Request with high priority to ensure it is reflected in MI_Admin_State as quickly as possible.

NOTE 2 – The EMF must combine any local lock request from the user with the received MI_Admin_State_Request, when setting MI_Admin_State in the MT/MT_A_So and MT/MT_A_Sk functions. However, when setting MI_Admin_State in the MTDe_TT_So function, the EMF must only do so based on a local request from the user. This is to prevent a deadlock situation where receipt of a lock instruct message causes transmission of a lock instruct message in the opposite direction.

8.9 Dataplane loopback processes

As described in [IETF RFC 6435], the dataplane loopback process controls looping back of all traffic (i.e., both data and OAM packets), under management control. An overview is shown in Figure 8-26.

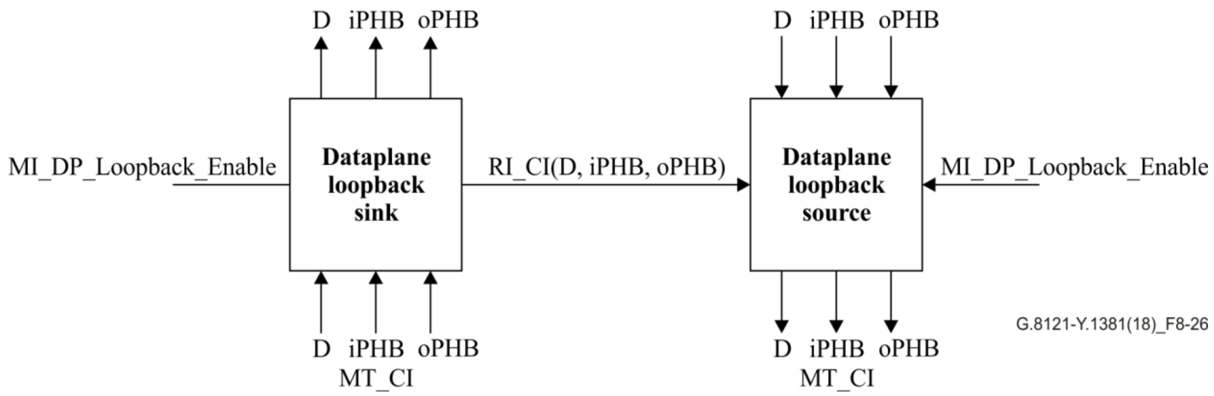


Figure 8-26 – Overview of the processes involved with dataplane loopback

NOTE – The dataplane loopback processes are included in the MEP diagnostic trail termination function (MTDe_TT, see clause 9.4.1.1) and the MIP diagnostic trail termination function (MTDi_TT, see clause 9.4.2.1). Dataplane loopback may also be enabled on a transport path at a point where there is no MEP or MIP; the location of the dataplane loopback processes in this case is for further study.

8.9.1 Dataplane loopback sink processes

The dataplane loopback sink process is illustrated in Figure 8-27.

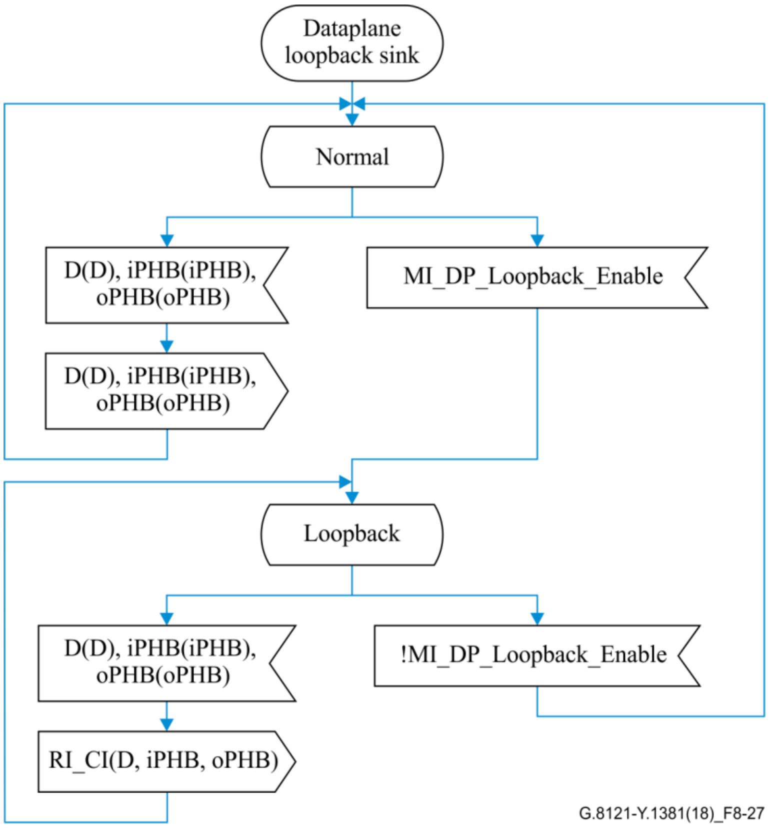


Figure 8-27 – Dataplane loopback sink process

In Normal state, data traffic passes through the process unmodified; in Loopback state, it is intercepted and sent to the dataplane loopback source process via RI_CI.

8.9.2 Dataplane loopback source processes

The dataplane loopback source process is illustrated in Figure 8-28.

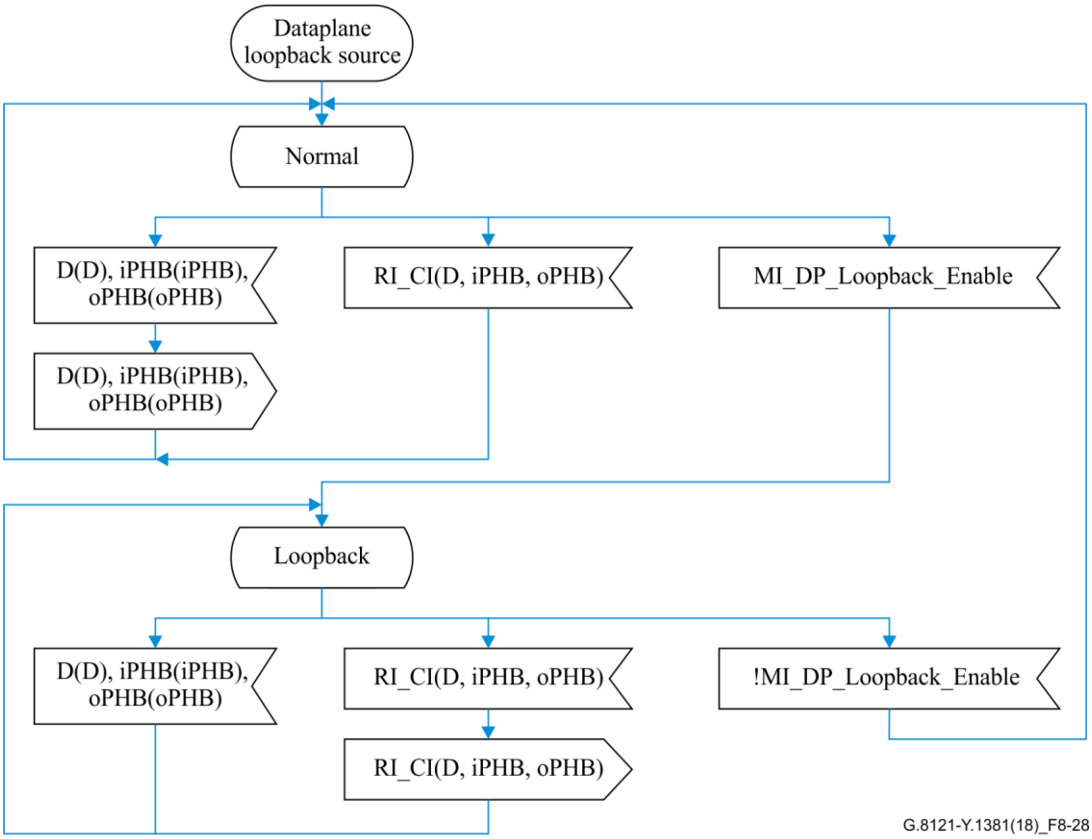


Figure 8-28 – Dataplane loopback source process

In Normal state, data traffic passes through the process unmodified; in Loopback state, data traffic is dropped, and data received via RI_CI from the dataplane loopback sink process is transmitted instead.

9 MPLS-TP layer functions

Figure 9-1 illustrates the MPLS-TP layer network and server and client layer adaptation functions. The information crossing the MPLS-TP connection point (MT_CP) is referred to as the MPLS-TP characteristic information (MT_CI) and consists of:

– MT_CI_D (containing the 'S' bit, TTL and MPLS payload)
– MT_CI_iPHB
– MT_CI_oPHB
– MT_CI_SSF
– MT_CI_APS
– MT_CI_LStack

These are described in clause 6.1.2 of [ITU-T G.8110.1] except for MT_CI_LStack, which contains a copy of the complete label stack received from the network.

The information crossing the MPLS-TP access point (MT_AP) is referred to as the MPLS-TP adapted information (MT_AI) and consists of:

– MT_AI_D (containing the 'S' bit and MPLS payload)
– MT_AI_PHB
– MT_AI_TSF
– MT_AI_TSD

− MT_AI_AIS
− MT_AI_LStack

These are described in clause 6.1.1 of [ITU-T G.8110.1] except for MT_AI_LStack, which is as above, and MT_AI_AIS, which indicates an alarm indication signal (AIS) condition.

The information crossing the MPLS-TP diagnostic access points (MTDe_AP and MTDi_AP) is referred to as MPLS-TP diagnostic adapted information (MTDe_AI and MTDi_AI respectively). These both consist of the same signals, with the same definitions, as MT_CI.

NOTE – MTDe_AI and MTDi_AI are not the same as MT_AI. In particular, MTDe_AI_D and MTDi_AI_D contain a TTL field (whereas MT_AI_D does not); and MTDe_AI and MTDi_AI contain separate iPHB and oPHB signals (whereas MT_AI contains a single PHB signal).

The MPLS-TP layer network provides embedded hierarchy via the label stacking mechanism. This is represented in the model by MPLS-TP tunnel sublayers, which contain MT_TT and MT/MT_A functions. Figure 9-1 shows a generic example for the connection of the MPLS-TP tunnel functions. It is not required to connect them via a MT_C function; they can be directly inserted without a connection function. It is noted that this Recommendation only defines Ethernet for the client of MPLS-TP as MT/ETH adaptation function.

This mechanism (MPLS-TP tunnel sublayers) is also used when sublayer (tandem connection) monitoring is required.

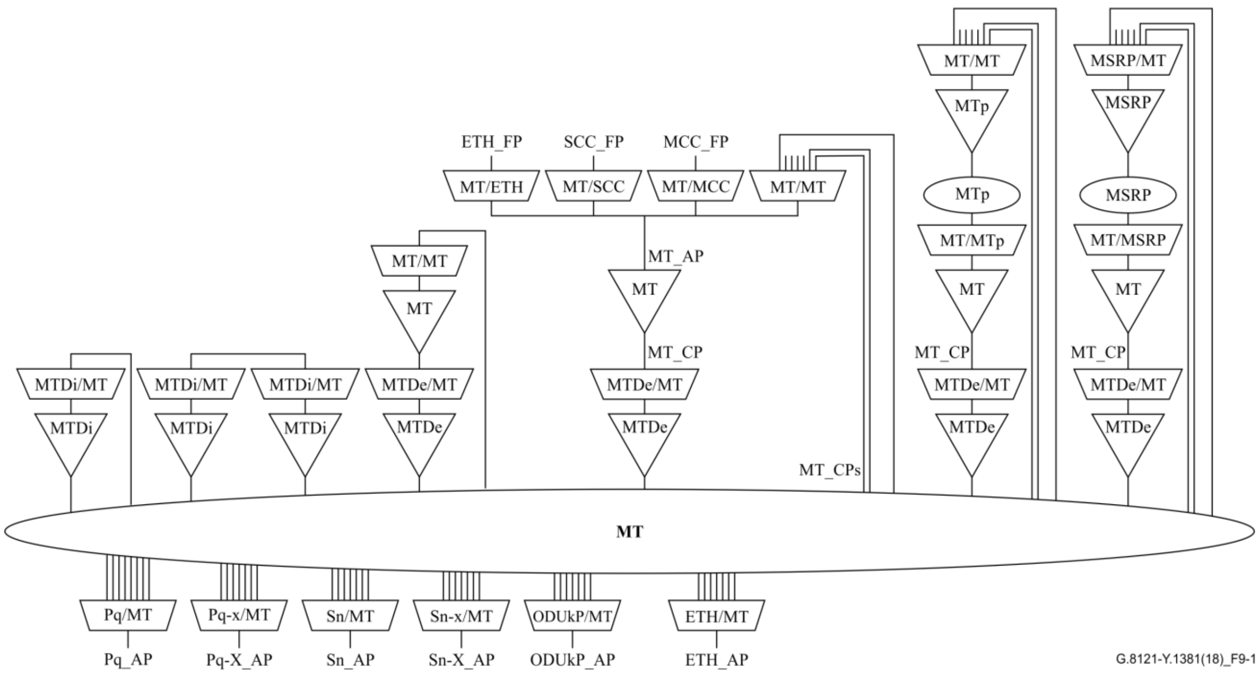


Figure 9-1 – MPLS-TP atomic functions

9.1 Connection functions

9.1.1 MPLS-TP connection function (MT_C)

MT_C is the function that assigns MPLS packets at its input ports to MPLS-TP packets at its output ports.

The MT_C connection process is a unidirectional function as illustrated in Figure 9-2. The signal formats at the input and the output ports of the function are similar, differing only in the logical sequence of the MPLS-TP packets. As the process does not affect the nature of the characteristic information of the signal, the reference point on either side of the MT_C function is the same, as illustrated in Figure 9-2.

Incoming MPLS-TP packets at the MT_CP are assigned to available outgoing MPLS-TP capacity at the MT_CP. In addition, the MT_C function supports the SNC/S protection schemes as defined in [ITU-T G.8131]. The Protection functionality is described in clause 9.1.1.1.

Symbol

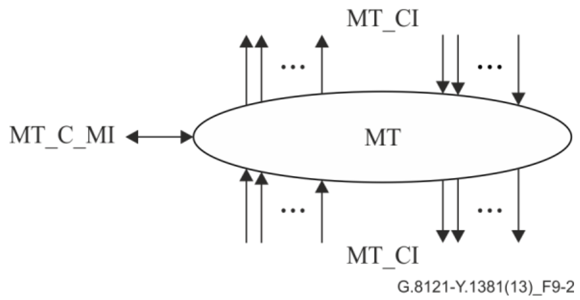


Figure 9-2 – MT_C symbol

Interfaces

The interfaces are described in Table 9-1.

Table 9-1– MT_C input and output signals

Input(s)	Output(s)
Per MT_CP, n × for the function: MT_CI_D MT_CI_iPHB MT_CI_oPHB MT_CI_APS MT_CI_SSF MT_CI_SSD (for SNC/S protection) MT_CI_LStack **Per input and output connection point:** *for further study* **MT_C_MP:** MT_C_MI_MatrixControl **Per matrix connection:** MT_C_MI_ConnectionType MT_C_MI_Return_CP_ID MT_C_MI_ConnectionPortIds **Per protection group:** MT_C_MI_PS_WorkingPortId MT_C_MI_PS_ProtectionPortId MT_C_MI_PS_ProtType MT_C_MI_PS_OperType MT_C_MI_PS_HoTime MT_C_MI_PS_WTR MT_C_MI_PS_ExtCMD MT_C_MI_PS_SD_Protection	**Per MT_CP, m × per function:** MT_CI_D MT_CI_iPHB MT_CI_oPHB MT_CI_APS MT_CI_SSF MT_CI_LStack **Per protection group:** MT_C_MI_cFOP-PMb MT_C_MI_cFOP-PMc MT_C_MI_cFOP-CM MT_C_MI_cFOP-NR MT_C_MI_cFOP-TO

Processes

In the MT_C function MPLS-TP characteristic information is routed between input (termination) connection points ((T)CPs) and output (T)CPs by means of matrix connections. (T)CPs may be allocated within a protection group.

NOTE – Neither the number of input/output signals to the connection function, nor the connectivity is specified in this Recommendation. That is a property of individual network elements.

MT-C process diagram is illustrated in Figure 9-3.

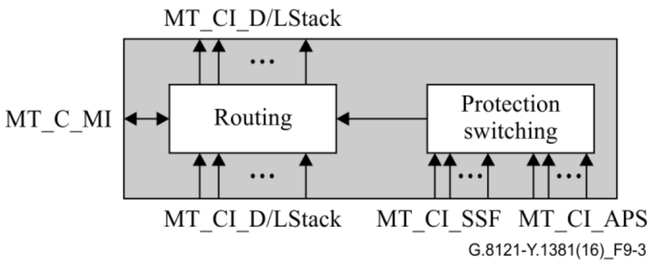


Figure 9-3 – MT_C process diagram

– *Routing process:*

This process passes all the traffic units received from a specific input to the corresponding output according to the matrix connection between the specified input and output, and it shall be able to remove an established matrix connection as defined by MI_MatrixControl.

Each (matrix) connection in the MT_C function shall be characterized by the:

Type of connection (MI_ConnectionType):	Unprotected, protected
Traffic direction (MI_Return_CP_ID):	Unidirectional if NULL, otherwise it identifies the CP of the return connection (Note)
Input and output connection points (MI_ConnectionPortIDs):	Set of connection point identifiers
NOTE – Bidirectional LSPs are supported by associating two unidirectional LSPs in the dataplane, as per [ITU-T G.8110.1].	

– *Protection switching process:*

Protection switching process for SNC/S is defined in clause 9.1.1.1.

Performance monitoring: None.

Defects: None.

Consequent actions: If an output of this function is not connected to one of its inputs, the connection function shall send no traffic units and SSF = false to the output.

Defect correlations: None.

9.1.1.1 Protection process for SNC/S

SNC protection with sublayer monitoring based on TCM is supported. For the term TCM, [IETF RFC 6371] uses the term called sub-path maintenance elements (SPMEs).

Figure 9-4 shows the involved atomic functions in SNC/S. The MT_TT_Sk provides the trail signal fail/trail signal degrade (TSF/TSD) protection switching criterion via the MT/MT_A_Sk function (SSF/SSD) to the MT_C function.

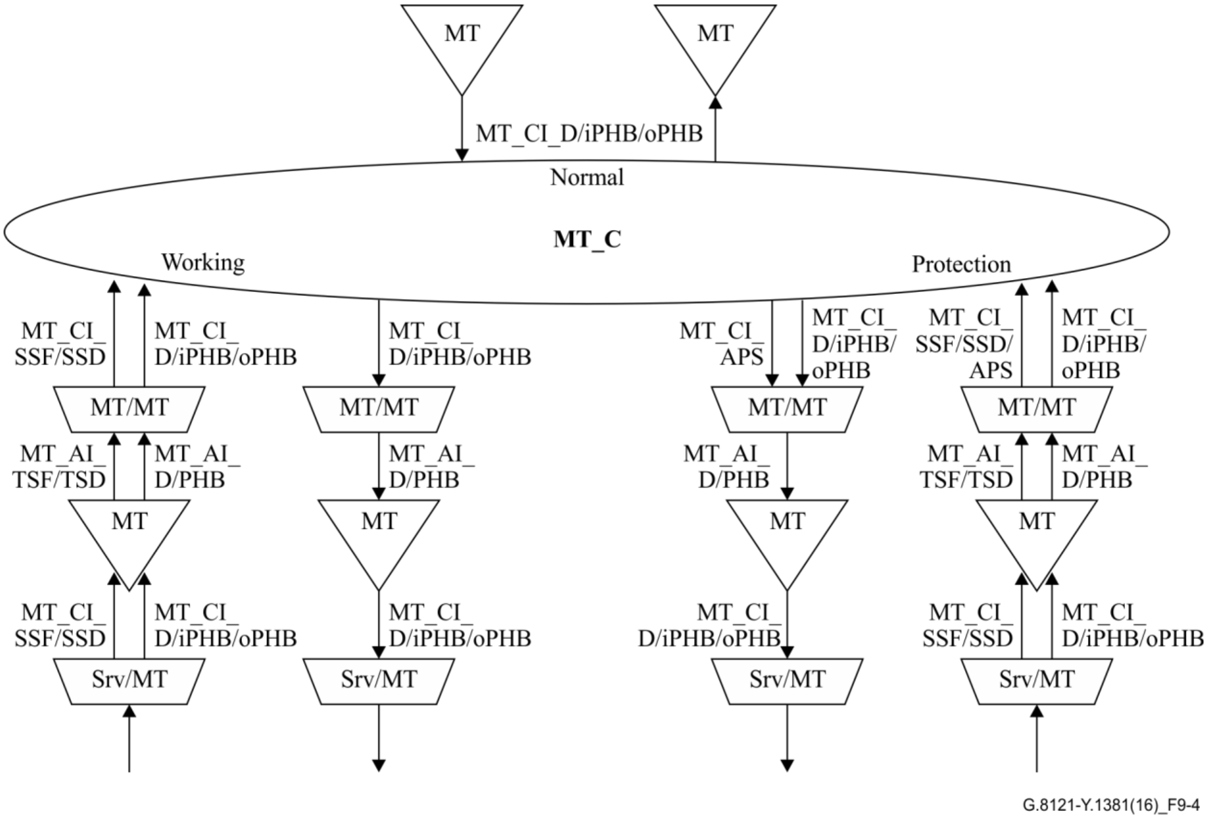


Figure 9-4 – Involved atomic functions in SNC/S

The protection functions at both ends operate the same way, by monitoring the working and protection subnetwork connections for defects, evaluating the system status taking into consideration the priorities of defect conditions and of external switch requests, and switching the appropriate subnetwork connection point (i.e., working or protection) to the protected (sub)network connection point.

The signal flows associated with the MT_ protection process are described with reference to Figure 9-5. The protection process receives control parameters and external switch requests at the management point (MP) reference point. The report of status information at the MP reference point is for further study.

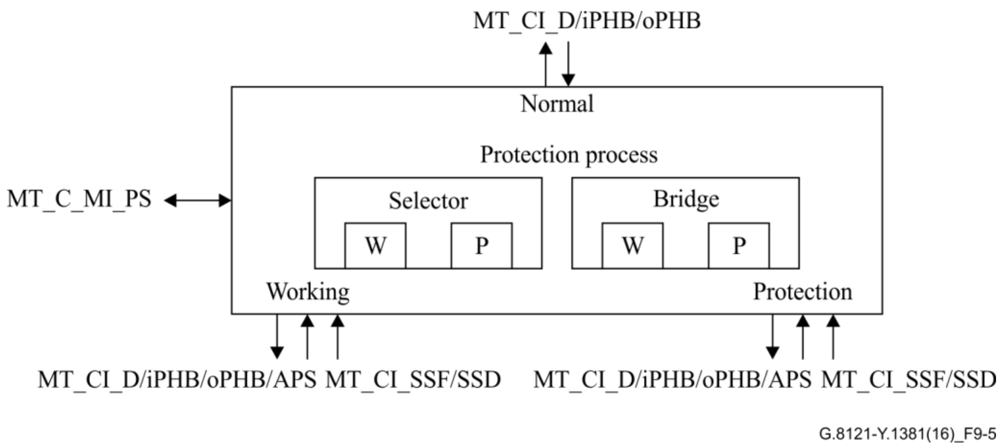


Figure 9-5 – Protection process

Source direction:

For a 1+1 architecture, the CI coming from the normal (protected) MT_CP is bridged permanently to both the working and protection MT_CP.

For a 1:1 architecture, the CI coming from the normal (protected) MT_CP is switched to either the working or the protection or both MT_CPs. A switch-over from working to protection MT_CP or vice versa is initiated by the switch initiation criteria defined below.

Sink direction:

For a 1+1 or 1:1 architecture, the CI coming from either the working or protection MT_CP is switched to the normal (protected) MT_CP. A switch-over from working to protection MT_CP or vice versa is initiated by the switch initiation criteria defined below.

Switch initiation criteria:

Automatic protection switching is based on the defect conditions of the working and the protection (sub)network connections, for SNC/S protection server signal fail (SSF) and server signal degrade (SSD).

A hold-off timer is provided in order to allow interworking between nested protection schemes. The hold-off timer delays switch initiation, in case of signal fail, in order to allow a nested protection to react and clear the fault condition. The hold-off timer is started by the activation of signal fail and runs for the hold-off time. Protection switching is only initiated if signal fail is still present at the end of the hold-off time. The hold-off time shall be provisionable between 0 and 10 s in steps of 100 ms; this is defined in clause 8.11 of [ITU-T G.8131].

Protection switching can also be initiated by external switch commands received via the MP or a request from the far end via the received MT_CI_APS. Depending on the mode of operation, internal states (e.g., wait-to-restore) may also affect a switch-over.

See the switching algorithm described in [ITU-T G.8131].

Switching time:

Refer to [ITU-T G.8131].

Switch restoration:

In the revertive mode of operation, the protected signal shall be switched back from the protection (sub)network connection to the working (sub)network connection when the working (sub)network connection has recovered from the fault.

To prevent frequent operation of the protection switch due to an intermittent fault, a failed working (sub)network connection must become fault-free for a certain period of time before it is used again. This period, called the wait-to-restore (WTR) period, should be of the order of 5-12 minutes and should be capable of being set. The WTR timer is defined in clause 8.12 of [ITU-T G.8131].

In the non-revertive mode of operation, no switch back to the working (sub)network connection is performed when it has recovered from the fault.

Configuration:

The following configuration parameters are defined in [ITU-T G.8131]:

– MT_C_MI_PS_WorkingPortId configures the working port.

– MT_C_MI_PS_ProtectionPortId configures the protection port.

– MT_C_MI_PS_ProtType configures the protection type.

– MT_C_MI_PS_OperType configures to be in revertive mode.

– MT_C_MI_PS_HoTime configures the hold-off timer.

– MT_C_MI_PS_WTR configures the wait-to-restore timer.
– MT_C_MI_PS_ExtCMD configures the protection group command.
– MT_C_MI_PS_SD_Protection configures the ability of a SNC protection switching process to trigger protection switching upon signal degrade (SD).

Defects:

The function detects dFOP-PMb, dFOP-PMc, dFOP-CM, dFOP-NR and dFOP-TO defects in case the APS protocol is used.

Consequent actions: None.

Defect correlations:

cFOP-PMb ← dFOP-PMb and (not CI_SSF)
cFOP-PMc ← dFOP-PMc and (not CI_SSF)
cFOP-CM ← dFOP-CM
cFOP-NR ← dFOP-NR and (not CI_SSF)
cFOP-TO ← dFOP-TO and (not dFOP-CM) and (not CI_SSF)

NOTE – In case of cFOP-PMb/PMc/NR/TO, CI_SSF of the protection transport entity is used.

9.1.2 Connection functions for trail protection (MTp_C)

MTp_C is the specific function that is used to support trail protection defined in [ITU-T G.8131].

The MTp_C connection process is a unidirectional function as illustrated in Figure 9-6 and the interface is described in Table 9-2.

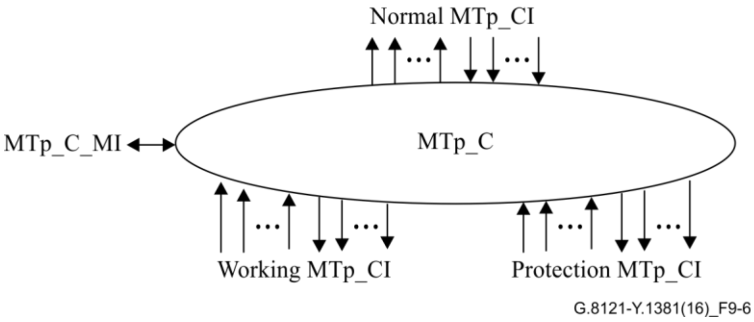


Figure 9-6 – MTp_C symbol

MPLS-TP packets at the normal MTp_CP are bridged/selected to/from the working and/or protection MTp_C by the protection switching process for trail protection. The Protection functionality is described in clause 9.1.2.1.

Table 9-2 – MTp_C input and output signals

Input(s)	Output(s)
Per normal MTp_CP: MTp_CI_D MTp_CI_PHB **Per working MTp_CP**: MTp_CI_D MTp_CI_PHB MTp_CI_SSF MTp_CI_SSD MTp_CI_AIS MTp_CI_LStack **Per protection MTp_CP**: MTp_CI_D MTp_CI_PHB MTp_CI_APS MTp_CI_SSF MTp_CI_SSD MTp_CI_LStack **Per input and output connection point**: *for further study* **Per protection group**: MTp_C_MI_PS_WorkingPortId MTp_C_MI_PS_ProtectionPortId MTp_C_MI_PS_ProtType MTp_C_MI_PS_OperType MTp_C_MI_PS_HoTime MTp_C_MI_PS_WTR MTp_C_MI_PS_ExtCMD MTp_C_MI_PS_SD_Protection	**Per normal MTp_CP**: MTp_CI_D MTp_CI_PHB MTp_CI_SSF MTp_CI_AIS MTp_CI_LStack **Per working MTp_CP**: MTp_CI_D MTp_CI_PHB **Per protection MTp_CP**: MTp_CI_D MTp_CI_PHB MTp_CI_APS **Per protection group**: MTp_C_MI_cFOP-PMb MTp_C_MI_cFOP-PMc MTp_C_MI_cFOP-CM MTp_C_MI_cFOP-NR MTp_C_MI_cFOP-TO

9.1.2.1 Protection process for trail protection

Figure 9-7 shows the involved atomic functions in trail protection. The MT_TT_Sk provides the TSF/TSD protection switching criterion via the MT/MTp_A_Sk function (SSF/SSD) to the MTp_C function.

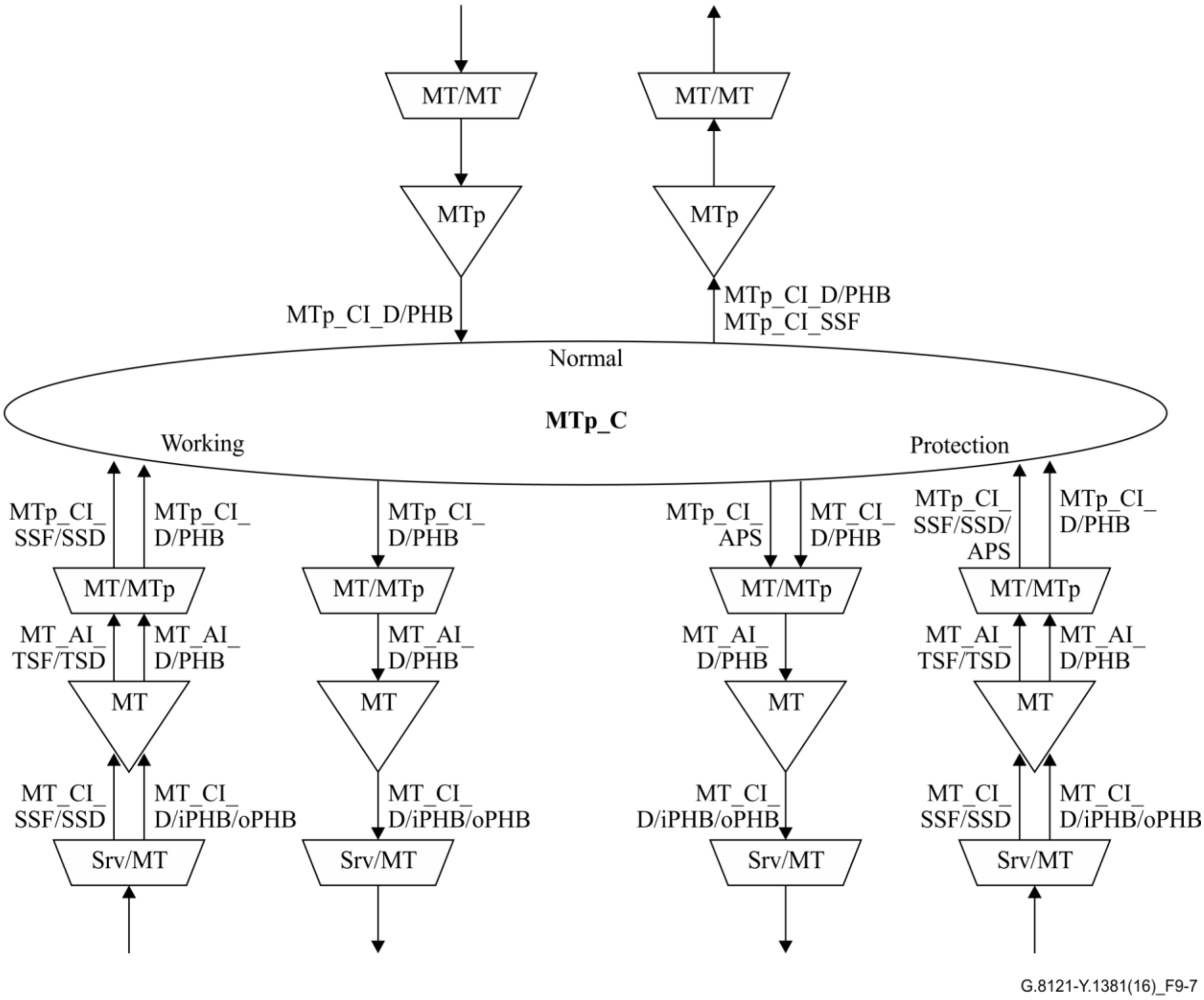


Figure 9-7 – Involved atomic functions in trail protection

The protection functions at both ends operate the same way, by monitoring the working and the protection trail for defects, evaluating the system status, taking into consideration the priorities of defect conditions and of external switch requests, and switching the appropriate connection point (i.e., working or protection) to the protected connection point.

The signal flows associated with the MTp_protection process are described with reference to Figure 9-8. The protection process receives control parameters and external switch requests at the MP reference point. The report of status information at the MP reference point is for further study.

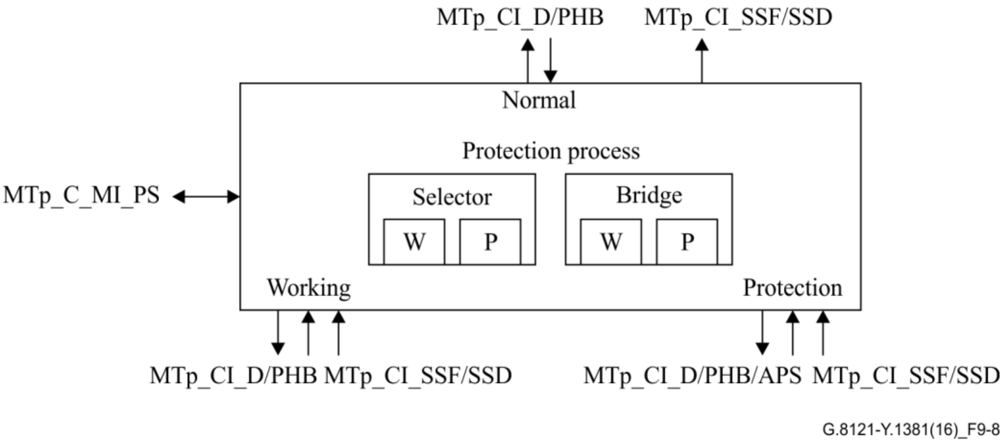


Figure 9-8 – Protection process for trail protection

Source direction:

For a 1+1 architecture, the CI coming from the normal (protected) MTp_CP is bridged permanently to both the working and protection MTp_CP.

For a 1:1 architecture, the CI coming from the normal (protected) MTp_CP is switched to either the working or the protection or both MT_CPs. A switch-over from working to protection MTp_CP or vice versa is initiated by the switch initiation criteria defined below.

Sink direction:

For a 1+1 or 1:1 architecture, the CI coming from either the working or protection MTp_CP is switched to the normal (protected) MTp_CP. A switch-over from working to protection MTp_CP or vice versa is initiated by the switch initiation criteria defined below.

Switch initiation criteria:

Automatic protection switching is based on the defect conditions of the working and protection trails, server signal fail (SSF) and server signal degrade (SSD).

A hold-off timer is provided in order to allow interworking between nested protection schemes. The hold-off timer delays switch initiation, in case of signal fail, in order to allow a nested protection to react and clear the fault condition. The hold-off timer is started by the activation of signal fail and runs for the hold-off time. Protection switching is only initiated if signal fail is still present at the end of the hold-off time. The hold-off time shall be provisionable between 0 and 10 s in steps of 100 ms; this is defined in clause 8.11 of [ITU-T G.8131].

Protection switching can also be initiated by external switch commands received via the MP or a request from the far end via the received MTp_CI_APS. Depending on the mode of operation, internal states (e.g., wait-to-restore) may also affect a switch-over.

See the switching algorithm described in [ITU-T G.8131].

Switching time:

Refer to [ITU-T G.8131].

Switch restoration:

In the revertive mode of operation, the protected signal shall be switched back from the protection trail to the working trail when the working trail has recovered from the fault.

To prevent frequent operation of the protection switch due to an intermittent fault, a failed working trail must become fault-free for a certain period of time before it is used again. This period, called the wait-to-restore (WTR) period, should be of the order of 5-12 minutes and should be capable of being set. The WTR is defined in clause 8.12 of [ITU-T G.8131].

In the non-revertive mode of operation no switch back to the working trail is performed when it has recovered from the fault.

Configuration:

The following configuration parameters are defined in [ITU-T G.8131]:

– MTp_C_MI_PS_WorkingPortId configures the working port.
– MTp_C_MI_PS_ProtectionPortId configures the protection port.
– MTp_C_MI_PS_ProtType configures the protection type.
– MTp_C_MI_PS_OperType configures to be in revertive mode.
– MTp_C_MI_PS_HoTime configures the hold-off timer.
– MTp_C_MI_PS_WTR configures the wait-to-restore timer.
– MTp_C_MI_PS_ExtCMD configures the protection group command.
– MTp_C_MI_PS_SD_Protection configures the ability of a trail protection switching process to trigger protection switching upon SD.

Defects:

The function detects dFOP-PMb, dFOP-PMc, dFOP-CM, dFOP-NR and dFOP-TO defects in case the APS protocol is used.

Consequent actions: None.

Defect correlations:

cFOP-PMb ← dFOP-PMb and (not CI_SSF)
cFOP-PMc ← dFOP-PMc and (not CI_SSF)
cFOP-CM ← dFOP-CM
cFOP-NR ← dFOP-NR and (not CI_SSF)
cFOP-TO ← dFOP-TO and (not dFOP-CM) and (not CI_SSF)

NOTE – In case of cFOP-PMb/PMc/NR/TO, CI_SSF of the protection transport entity is used.

9.1.3 Connection functions for MPLS-TP shared ring protection (MSRP_C)

MSRP_C is the specific function that is used to support ring protection as specified in [ITU-T G.8132].

The MSRP_C connection process is a unidirectional function as illustrated in Figure 9-9 and the interface is described in Table 9-3.

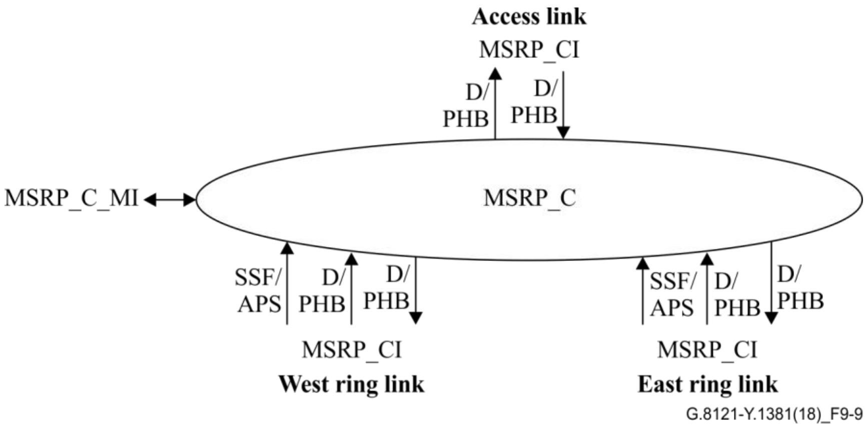


Figure 9-9 – MSRP_C symbol

Table 9-3 – MSRP_C input and output signals

Input(s)	Output(s)
Per access MSRP_CP: MSRP_CI_D MSRP_CI_PHB **Per East/West MSRP_CP**: MSRP_CI_D MSRP_CI_PHB MSRP_CI_SSF MSRP_CI_APS MSRP_CI_LStack **Per input and output connection point**: *for further study* **Per ring protection process**: MSRP_C_MI_MSRP_EastPort MSRP_C_MI_MSRP_WestPort MSRP_C_MI_MSRP_ProtType MSRP_C_MI_MSRP_HoTime MSRP_C_MI_MSRP_WTR MSRP_C_MI_MSRP_ExtCMD	**Per access MSRP_CP**: MSRP_CI_D MSRP_CI_PHB MSRP_CI_LStack **Per East/West MSRP_CP**: MSRP_CI_D MSRP_CI_PHB MSRP_CI_SSF MSRP_CI_APS **Per ring protection process**: *for further study*

9.1.3.1 Protection process for MPLS-TP shared ring protection

Figure 9-10 provides the involved atomic functions that support MSRP. This figure includes the LSPs entering to/exiting from MSRP.

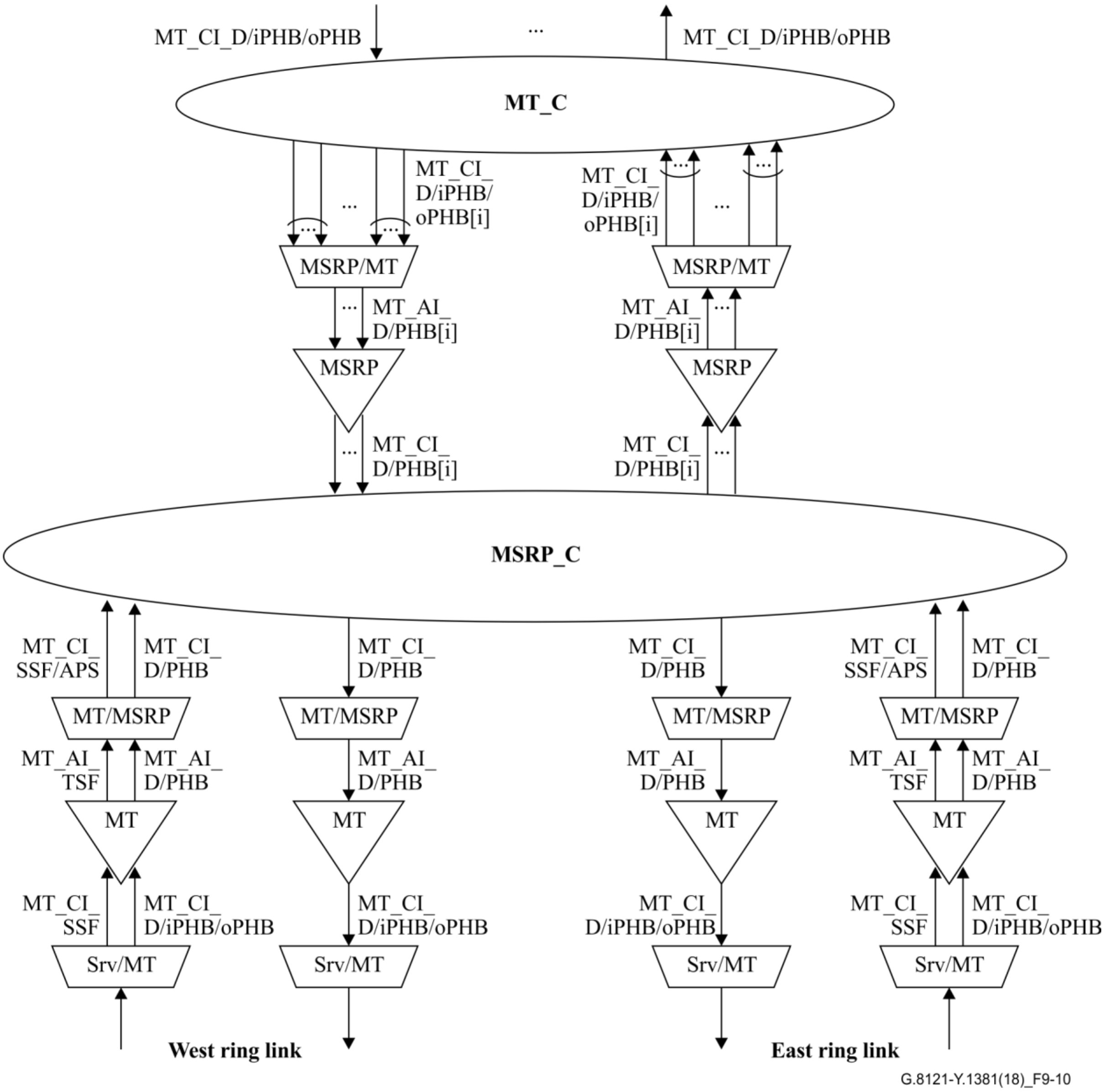


Figure 9-10 – Involved atomic functions MSRP

Figure 9-11 shows the involved atomic functions for MSRP_C. This figure is the same as Figure 8-1 in [ITU-T G.8132]. The MT_TT_Sk provides the trail signal fail (TSF) protection switching criterion via the MT/MT_A_Sk function (SSF) to the MSRP_C function. The ring protection switch (RPS) in Figure 9-11 corresponds to the APS in Figure 9-10.

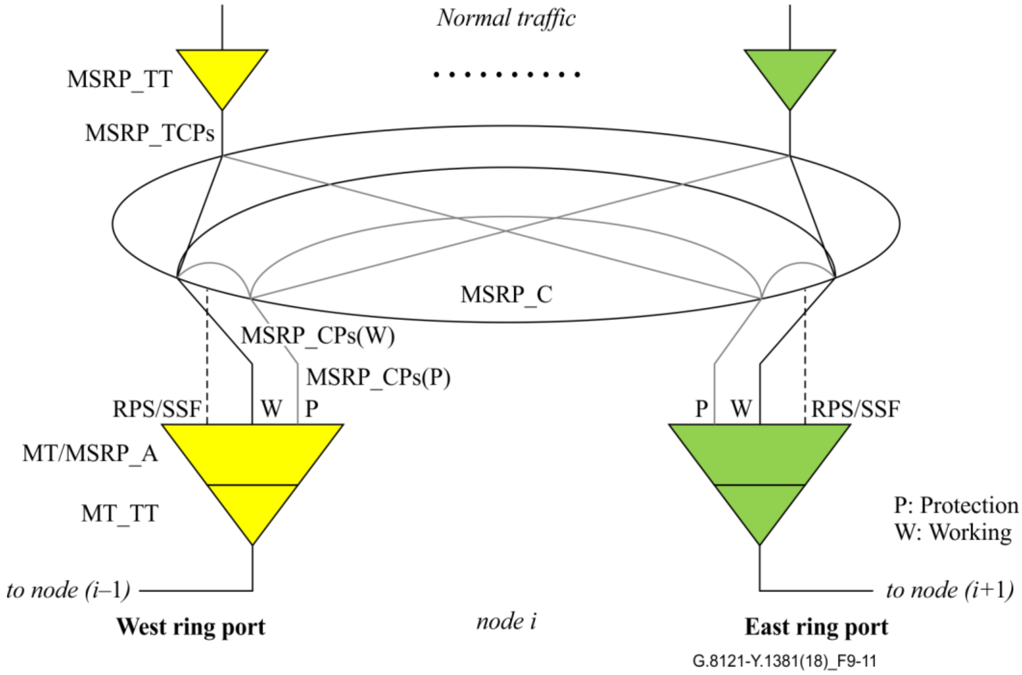


Figure 9-11– Involved atomic functions for MSRP_C

The signal flows associated with the MSRP protection process are described with reference to Figure 9-12. The protection process receives control parameters and external switch requests at the management point (MP) reference point. The report of status information at the MP reference point is for further study.

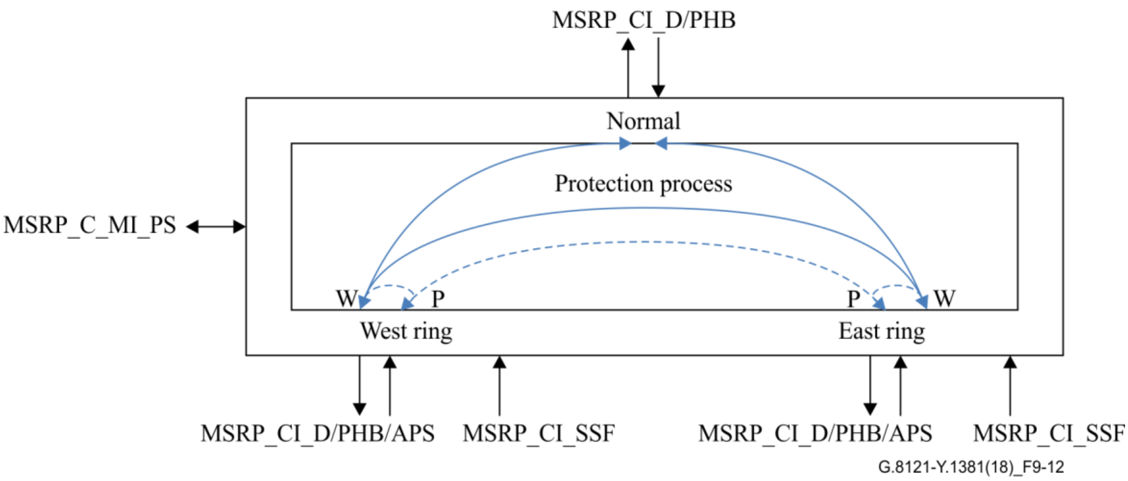


Figure 9-12 – Protection process for MSRP

Switching direction:

The CI coming from the normal MSRP_CP is switched to either the east ring or the west ring MSRP_CP.

The CI coming from the east ring or west ring MSRP_CP is switched to either the east ring or the west ring or the normal MSRP_CP. A switch-over is initiated by the switch initiation criteria defined below.

Switch initiation criteria:

Automatic protection switching is based on the defect conditions of section MEP. SF is declared when the MPLS-TP trail termination sink (MT_TT_Sk) function of an MPLS-TP section MEP detects failure.

A hold-off timer is provided. This is defined in clause 13 of [ITU-T G.8132].

Protection switching can also be initiated by external switch commands received via the MP or a request from the far end via the received MT_CI_APS. Depending on the mode of operation, internal states (e.g., wait-to-restore) may also affect a switch-over.

See the switching algorithm described in [ITU-T G.8132].

Switching time:

Refer to clause 7 in [ITU-T G.8132].

Switch restoration:

MSRP supports only the revertive protection operation type. Refer to clause 11 in [ITU-T G.8132].

Configuration:

The following configuration parameters are defined in [ITU-T G.8132]:

– MI_MSRP_EastPort configures the east ring port that contains Tunnel labels. In case of Node D in Figure 8-2 of [ITU-T G.8132], This MI configures RcW_D(D) and RcW_X(D) (X means other nodes but C in the ring.)
– MI_MSRP_WestPort configures the protection port. In case of Node D in Figure 8-2 of [ITU-T G.8132], This MI configures RaW_Y(D) (Y means all the nodes in the ring.)
– MI_MSRP_ProtType configures the protection type.
– MI_MSRP_HoTime configures the hold-off timer.
– MI_MSRP_WTR configures the wait-to-restore timer.
– MI_MSRP_ExtCMD configures the protection group command.

Defects: None.

Consequent actions: None.

Defect correlations: None.

9.2 Termination functions

9.2.1 MPLS-TP trail termination (MT_TT) function

The bidirectional MPLS-TP trail termination (MT_TT) function terminates the MPLS-TP OAM to determine the status of the MPLS-TP (sub)layer trail. The MT_TT function is performed by a co-located pair of the MPLS-TP trail termination source (MT_TT_So) and sink (MT_TT_Sk) functions as shown in Figure 9-13.

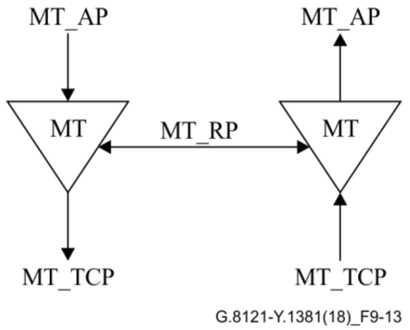


Figure 9-13 – MT_TT function

9.2.1.1 MPLS-TP trail termination source (MT_TT_So) function

The MT_TT_So function determines and inserts the TTL value in the shim header TTL field and adds MPLS-TP OAM for proactive monitoring to the MT_AI signal at its MT_AP.

Symbol

The MT_TT_So function symbol is shown in Figure 9-14.

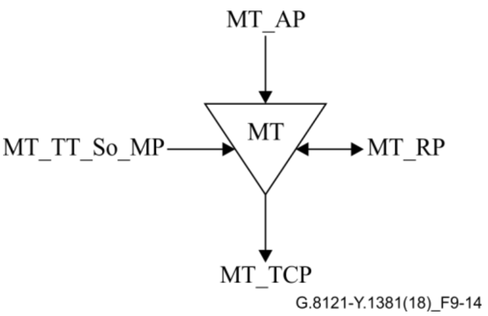


Figure 9-14 – MT_TT_So function

Interfaces

The interfaces are described in Table 9-4.

Table 9-4 – MT_TT_So inputs and outputs

Input(s)	Output(s)
MT_AP: MT_AI_D MT_AI_PHB **MT_RP:** MT_RI_CC_RDI MT_RI_CC_Blk MT_RI_OAM_Info(D,CoS,DP) **MT_TT_So_MP:** MT_TT_So_MI_GAL_Enable MT_TT_So_MI_TTLVALUE MT_TT_So_MI_ MEG_ID MT_TT_So_MI_ MEP_ID	**MT_TCP:** MT_CI_D MT_CI_oPHB MT_CI_iPHB **MT_RP:**

Table 9-4 – MT_TT_So inputs and outputs

Input(s)	Output(s)
MT_TT_So_MI_CC_OAM_Tool MT_TT_So_MI_RDI_OAM_Tool MT_TT_So_MI_CC_Enable (Note) MT_TT_So_MI_CVp_Enable (Note) MT_TT_So_MI_CC_CoS MT_TT_So_MI_CC_Period MT_TT_So_MI_1LMp_Enable MT_TT_So_MI_LMp_OAM_Tool MT_TT_So_MI_LMp_Enable[1...M_{LMp}] MT_TT_So_MI_LMp_Period[1...M_{LMp}] MT_TT_So_MI_LMp_CoS[1...M_{LMp}] MT_TT_So_MI_DMp_OAM_Tool MT_TT_So_MI_DMp_Enable[1...M_{DMp}] MT_TT_So_MI_DMp_Period[1...M_{DMp}] MT_TT_So_MI_DMp_Test_ID[1...M_{DMp}] MT_TT_So_MI_DMp_CoS[1...M_{DMp}] MT_TT_So_MI_DMp_Length[1...M_{DMp}] MT_TT_So_MI_1DMp_OAM_Tool MT_TT_So_MI_1DMp_Enable[1...M_{1DMp}] MT_TT_So_MI_1DMp_Period[1...M_{1DMp}] MT_TT_So_MI_1DMp_Test_ID[1...M_{1DMp}] MT_TT_So_MI_1DMp_Length[1...M_{1DMp}] MT_TT_So_MI_1DMp_CoS[1...M_{1DMp}] MT_TT_So_MI_SLp_OAM_Tool MT_TT_So_MI_SLp_Enable[1...M_{SLp}] MT_TT_So_MI_SLp_Period[1...M_{SLp}] MT_TT_So_MI_SLp_Test_ID[1...M_{SLp}] MT_TT_So_MI_SLp_Length[1...M_{SLp}] MT_TT_So_MI_SLp_CoS[1...M_{SLp}] MT_TT_So_MI_1SLp_OAM_Tool MT_TT_So_MI_1SLp_Enable[1...M_{1SLp}] MT_TT_So_MI_1SLp_Period[1...M_{1SLp}] MT_TT_So_MI_1SLp_Test_ID[1...M_{1SLp}] MT_TT_So_MI_1SLp_Length[1...M_{1SLp}] MT_TT_So_MI_1SLp_CoS[1...$M1_{1SLp}$] **MT_TP:** MT_TT_So_TI_TimeStampl	
NOTE – MI_CC_Enable and MI_CVp_Enable are used to enable CC and CV functions respectively. The possible combinations are: – no CC function and no CV function: MI_CC_Enable = false and MI_CVp_Enable = false – CC-only function: MI_CC_Enable = true and MI_CVp_Enable = false	

Table 9-4 – MT_TT_So inputs and outputs

Input(s)	Output(s)
– CC and CV functions: MI_CC_Enable = true and MI_CVp_Enable = true	

Processes

The processes associated with the MT_TT_So function are as depicted in Figure 9-15.

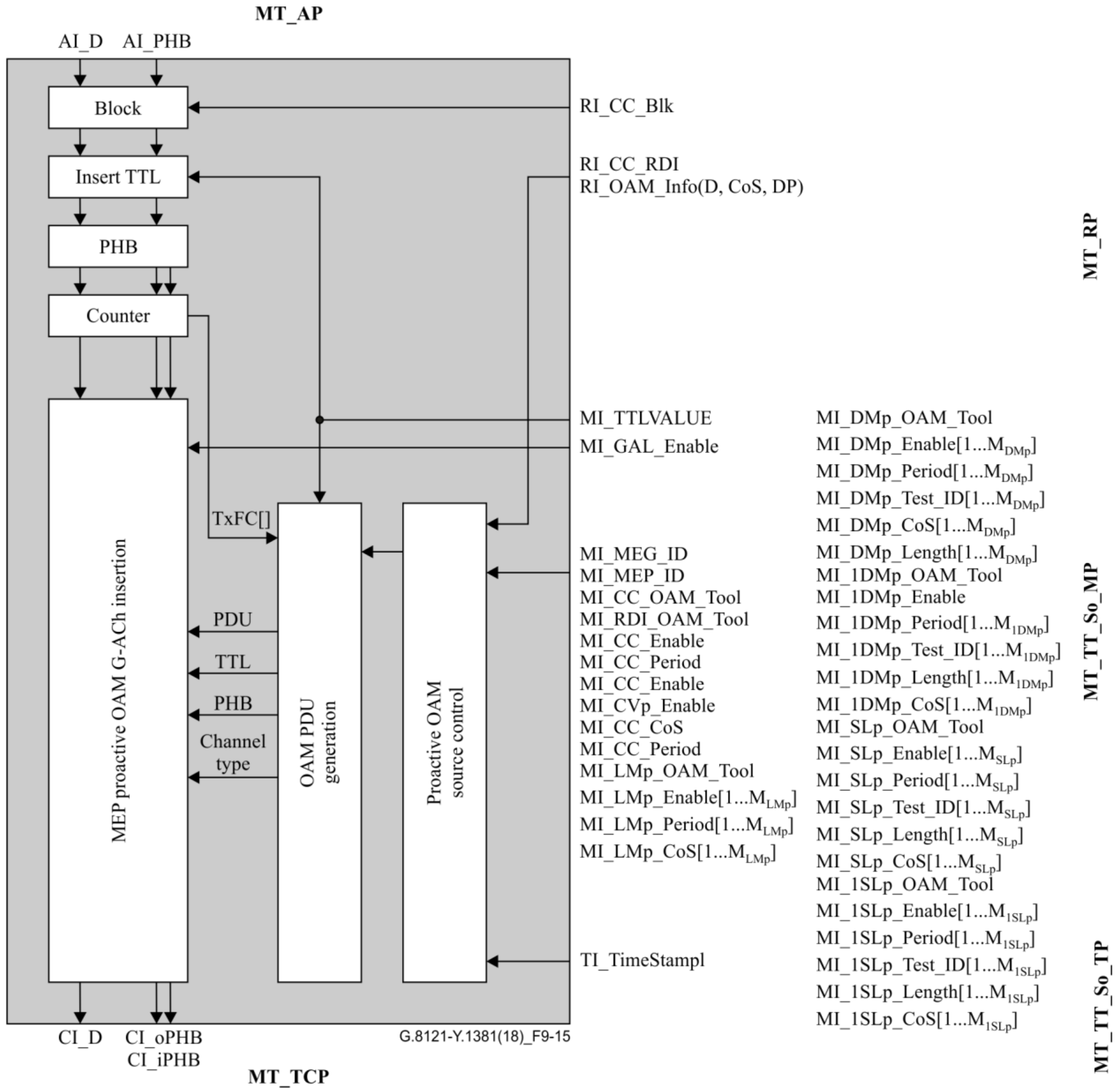


Figure 9-15 – MT_TT_So process diagram

NOTE 1 – The interface between proactive OAM control and OAM PDU generation is protocol specific.

NOTE 2 –The parameters and values in the MT_TT_So_MI_XX_OAM_Tool are outside the scope of this Recommendation.

Block: When RI_CC_Blk is raised, the Block process will discard all AI_D traffic units it receives. If RI_CC_Blk is cleared, the received AI_D traffic units will be passed to the output port.

Insert TTL: The time-to-live value is inserted in the outer shim header's TTL field within the MT_AI traffic unit

PHB: The AI_PHB signal is assigned to both the CI_iPHB and CI_oPHB signals at the MT_TCP reference point.

Counter: This process is used to count packets for proactive loss measurements. The location of the Counter process is shown for illustration only. The exact set of packets to be counted is outside the scope of this Recommendation.

MEP proactive OAM G-ACh insertion: See clause 8.1.

OAM PDU generation: See clause 8.8.

Proactive OAM source control: See clause 8.8.

Defects: None.

Consequent actions: None.

Defect correlations: None.

Performance monitoring: None.

9.2.1.2 MPLS-TP trail termination sink (MT_TT_Sk) function

The MT_TT_Sk function reports the state of the MPLS-TP trail (network connection). It extracts MPLS-TP trail OAM - for proactive monitoring - from the MPLS-TP signal at its MT_TCP, detects defects, counts during 1-second periods errors and defects to feed performance monitoring when connected and forwards the defect information as backward indications to the companion MT_TT_So function.

NOTE – The MT_TT_Sk function extracts and processes one level of MPLS-TP OAM irrespective of the presence of more levels.

Symbol

The MT_TT_Sk function symbol is shown in Figure 9-16.

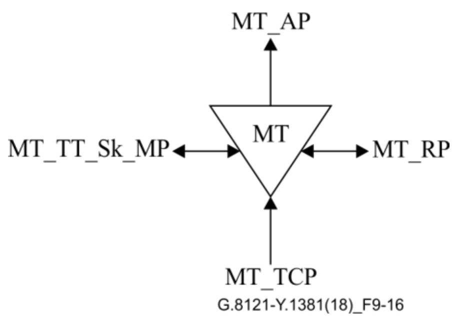


Figure 9-16 – MT_TT_Sk function

Interfaces

The interfaces are described in Table 9-5.

Table 9-5 – MT_TT_Sk inputs and outputs

Input(s)	Output(s)
MT_TCP: MT_CI_D MT_CI_iPHB MT_CI_oPHB MT_CI_SSF MT_CI_Lstack	**MT_AP**: MT_AI_D MT_AI_PHB MT_AI_TSF MT_AI_TSD MT_AI_AIS
MT_RP:	MT_AI_LStack
MT_TT_Sk_MP:	**MT_RP**: MT_RI_CC_RDI

Table 9-5 – MT_TT_Sk inputs and outputs

Input(s)	Output(s)
MT_TT_Sk_MI_GAL_Enable MT_TT_Sk_MI_ MEG_ID MT_TT_Sk_MI_ PeerMEP_ID MT_TT_Sk_MI_ CC_OAM_Tool MT_TT_Sk_MI_ RDI_OAM_Tool MT_TT_Sk_MI_CC_Enable (Note) MT_TT_Sk_MI_CVp_Enable (Note) MT_TT_Sk_MI_CC_Period MT_TT_Sk_MI_CC_CoSMT_TT_Sk_MI_Get_SvdCC MT_TT_Sk_MI_1LMp_Enable MT_TT_Sk_MI_LMp_OAM_Tool MT_TT_Sk_MI_LMp_Enable[1... M_{LMp}] MT_TT_Sk_MI_LMp_CoS[1... M_{LMp}] MT_TT_Sk_MI_LM_DEGM MT_TT_Sk_MI_LM_M MT_TT_Sk_MI_LM_DEGTHR MT_TT_Sk_MI_LM_TFMIN MT_TT_Sk_MI_ DMp_OAM_Tool MT_TT_Sk_MI_DMp_Enable[1… M_{DMp}] MT_TT_Sk_MI_DMp_CoS[1… M_{DMp}] MT_TT_Sk_MI_1DMp_OAM_Tool MT_TT_Sk_MI_1DMp_Enable[1...M_{1DMp}] MT_TT_Sk_MI_1DMp_Test_ID[1...M_{1DMp}] MT_TT_Sk_MI_ SLp_OAM_Tool MT_TT_Sk_MI_SLp_Enable[1… M_{SLp}] MT_TT_Sk_MI_SLp_CoS[1… M_{SLp}] MT_TT_Sk_MI_ 1SLp_OAM_Tool MT_TT_Sk_MI_1SLp_Enable[1...M_{1SLp}] MT_TT_Sk_MI_1SLp_Test_ID[1...M_{1SLp}] MT_TT_Sk_MI_AIS_OAM_Tool MT_TT_Sk_MI_LCK_OAM_Tool MT_TT_Sk_MI_1second **MT_TP**: MT_TT_Sk_TI_TimeStampl	MT_RI_CC_Blk MT_RI_OAM_Info(D,CoS,DP) **MT_TT_Sk_MP**: MT_TT_Sk_MI_SvdCC MT_TT_Sk_MI_cSSF MT_TT_Sk_MI_cLCK MT_TT_Sk_MI_cLOC MT_TT_Sk_MI_cMMG MT_TT_Sk_MI_cUNM MT_TT_Sk_MI_cUNP MT_TT_Sk_MI_cUNC MT_TT_Sk_MI_cDEG MT_TT_Sk_MI_cRDI MT_TT_Sk_MI_pN_LF[1...P] MT_TT_Sk_MI_pN_TF[1...P] MT_TT_Sk_MI_pF_LF[1...P] MT_TT_Sk_MI_pF_TF[1...P] MT_TT_Sk_MI_pF_DS MT_TT_Sk_MI_pN_DS MT_TT_Sk_MI_pB_FD[1...P] MT_TT_Sk_MI_pB_FDV[1...P] MT_TT_Sk_MI_pN_FD[1...P] MT_TT_Sk_MI_pN_FDV[1...P] MT_TT_Sk_MI_pF_FD[1...P] MT_TT_Sk_MI_pF_FDV[1...P]
NOTE – See NOTE in Table 9-4	

Processes

The processes associated with the MT_TT_Sk function are as depicted in Figure 9-17.

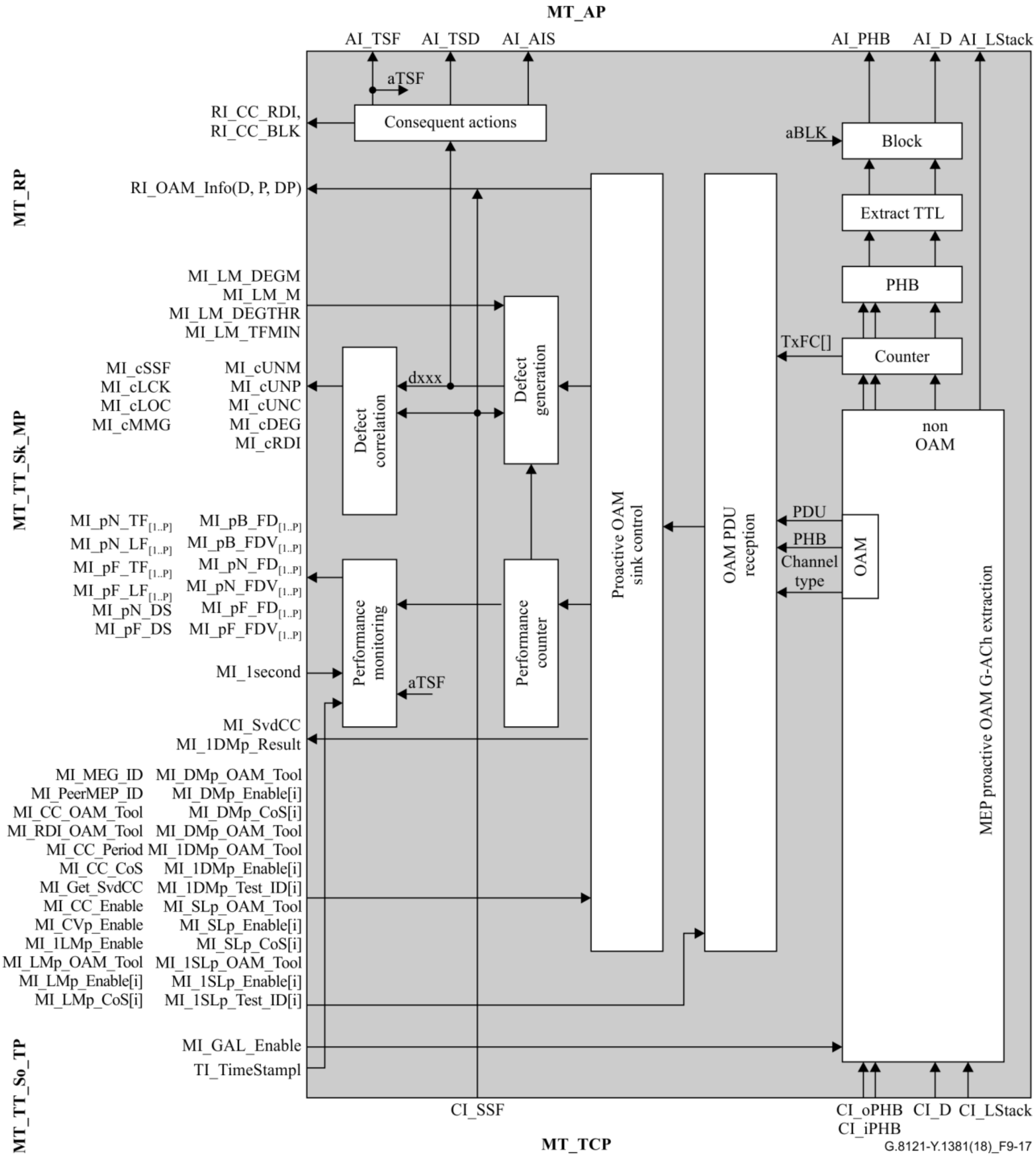


Figure 9-17 – MT_TT_Sk process diagram

NOTE – The parameters and values in the MT_TT_Sk_MI_XX_OAM_Tool are outside the scope of this Recommendation.

Extract TTL: The time-to-live value is extracted from the outer shim header's TTL field within the MT_CI traffic unit

Block: When the aBlock (aBLK) consequent action is asserted, this process drops all traffic units arriving at its input.

PHB: The CI_oPHB signal is assigned to the AI_PHB signal at the reference point MT_AP.

Note that the CI_iPHB signal is not used by any of the processes in the function.

Counter: This process is used to count packets for proactive loss measurements. The location of the counter process is shown for illustration only. The exact set of packets to be counted is outside the scope of this Recommendation.

MEP proactive OAM G-ACh extraction: See clause 8.1.3. The process of CI_D is made according to the following rule:

```
if ( ((MI_GAL_Enable && MT-label(D) == GAL) ||
       (!MI_GAL_Enable && 1stNibble(D) == 0b0001)) &&
         (Packet_Type(D) == Proactive_OAM && !APS_OAM) )
    forward to G-ACh port
} else {
    forward to data port
}
```

NOTE – MT-label(D) and 1stNibble(D) are functions which return, respectively, the Label field at the top of the stack within the MPLS payload, the first nibble of the MPLS payload as defined in Figure 6-3 of [ITU-T G.8110.1].

These fields are used to identify G-ACh packets as described in clause 4.2.1 of [IETF RFC 5586]. Packet_Type (D) is a protocol-specific function that determines whether the traffic unit contains a proactive OAM packet.

OAM PDU reception: See clause 8.8

Proactive OAM sink control: See clause 8.8

Defects generation:

This function detects and clears the defects (dLOC, dMMG, dUNM, dDEG, dUNP, dUNC, dRDI, dAIS and dLCK) as defined in clause 6.1

Consequent actions:

aBLK ← (dMMG or dUNM)

aTSF ← (dLOC and MI_CC_Enable) or (dAIS and not(MI_CC_Enable)) or (dLCK and not (MI_CC_Enable)) or dMMG or dUNM or CI_SSF

aTSD ← dDEG and (not aTSF)

aAIS ← aTSF

aRDI ← aTSF

Defect correlations:

cLOC ← dLOC and (not dAIS) and (not dLCK) and (not CI_SSF) and (MI_CC_Enable)

cMMG ← dMMG

cUNM ← dUNM

cDEG ← dDEG and (not dAIS) and (not dLCK) and (not CI_SSF) and (not (dLOC or dMMG or dUNM)) and (MI_CC_Enable))

cUNP ← dUNP

cUNC ← dUNC

cRDI ← dRDI and (MI_CC_Enable)

cSSF ← CI_SSF or dAIS

cLCK ← dLCK and (not dAIS)

Performance monitoring:

pN_TF ← N_TF

pN_LF ← N_LF

pF_TF ← F_TF

pF_LF ← F_LF

pN_DS ← aTSF

pF_DS ← aRDI[1]

pB_FD ← B_FD

pB_FDV ← B_FDV

pF_FD ← F_FD

pF_FDV ← F_FDV

pN_FD ← N_FD

pN_FDV ← N_FDV

9.2.2 MPLS-TP trail termination function for trail protection (MTp_TT)

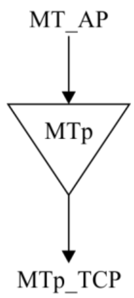
The MTp_TT function is used for trail protection and it is included to satisfy the modelling rules.

The bidirectional MTp_TT is performed by a co-located pair of MTp_TT source (MTp_TT_So) and sink (MTp_TT_Sk) functions.

9.2.2.1 MPLS-TP trail termination source function for trail protection (MTp_TT_So)

Symbol

The MT_TT_So function symbol is shown in Figure 9-18.



G.8121-Y.1381(18)_F9-18

Figure 9-18 – MTp_TT_So function

Interfaces

The interfaces are described in Table 9-6.

Table 9-6 – MTp_TT_So inputs and outputs

Input(s)	Output(s)
MT_AP: MT_AI_D MT_AI_PHB	**MTp_TCP**: MTp_CI_D MTp_CI_PHB

Processes

The processes associated with the MTp_TT_So function are as depicted in Figure 9-19.

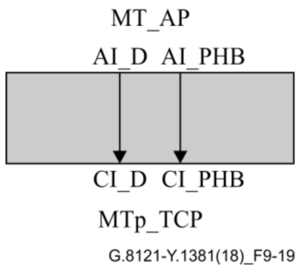


Figure 9-19– MTp_TT_So process diagram

Defects: None.

Consequent actions: None.

Defect correlations: None.

Performance monitoring: None.

9.2.2.2 MPLS-TP trail termination sink function for trail protection (MTp_TT_Sk)

Symbol

The MTp_TT_Sk function symbol is shown in Figure 9-20.

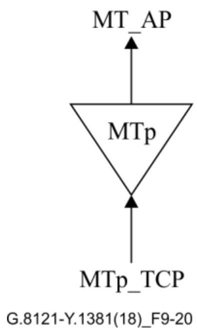


Figure 9-20 – MTp_TT_Sk function

Interfaces

The interfaces are described in Table 9-7.

Table 9-7 – MTp_TT_Sk inputs and outputs

Input(s)	Output(s)
MTp_TCP: MTp_CI_D MTp_CI_PHB	**MT_AP**:

Input(s)	Output(s)
MTp_CI_Lstack MTp_CI_SSF MTp_CI_AIS	MT_AI_D MT_AI_PHB MT_AI_LStack MT_AI_TSF MTp_CI_AIS

Processes

The processes associated with the MTp_TT_Sk function are as depicted in Figure 9-21.

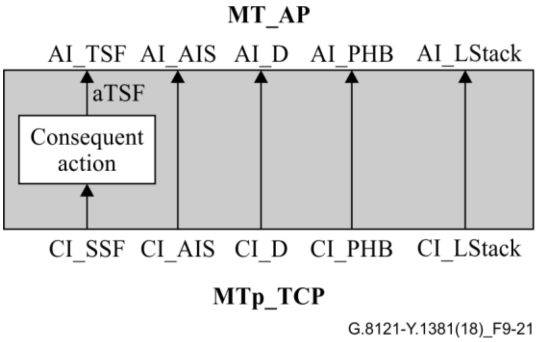


Figure 9-21 – MTp_TT_Sk process diagram

Defects: None.

Consequent actions:

The function shall perform the following consequent actions:

aTSF ← CI_SSF

Defect correlations: None.

Performance monitoring: None.

9.2.3 MPLS-TP shared ring protection trail termination function (MSRP_TT)

9.2.3.1 MPLS-TP shared ring protection trail termination source function (MSRP_TT_So)

Symbol

The MSRP_TT_So function symbol is shown in Figure 9-22.

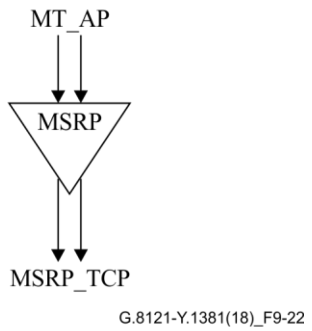


Figure 9-22 – MSRP_TT_So function

Interfaces

The interfaces are described in Table 9-8.

Table 9-8 – MSRP_TT_So inputs and outputs

Input(s)	Output(s)
per MT_AP: MT_AI_D MT_AI_PHB	**per MSRP_TCP**: MSRP_CI_D MSRP_CI_PHB

Processes

The processes associated with the MSRP_TT_So function are as depicted in Figure 9-23.

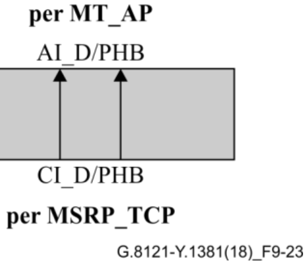


Figure 9-23 – MSRP_TT_So process diagram

Defects: None.

Consequent actions: None.

Defect correlations: None.

Performance monitoring: None.

9.2.3.2 MPLS-TP shared ring protection trail termination sink function (MSRP_TT_Sk)

Symbol

The MSRP_TT_Sk function symbol is shown in Figure 9-24.

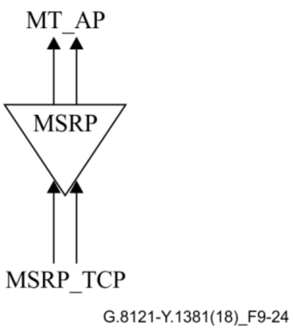


Figure 9-24 – MSRP_TT_Sk function

Interfaces

The interfaces are described in Table 9-9.

Table 9-9 – MSRP_TT_Sk inputs and outputs

Input(s)	Output(s)
per MSRP_TCP: MSRP_CI_D MSRP_CI_PHB	**per MT_AP**: MT_AI_D MT_AI_PHB

Processes

The processes associated with the MSRP_TT_Sk function are as depicted in Figure 9-25.

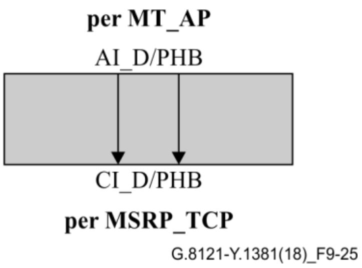


Figure 9-25 – MSRP_TT_Sk process

Defects: None.

Consequent actions: None.

Defect correlations: None.

Performance monitoring: None.

9.3 Adaptation functions

9.3.1 MPLS-TP to MPLS-TP adaptation function (MT/MT_A)

9.3.1.1 MPLS-TP to MPLS-TP adaptation source function (MT/MT_A_So)

Symbol

The MT/MT_A_So function symbol is shown in Figure 9-26. This function maps client MT_CI traffic units into server MT_AI traffic units.

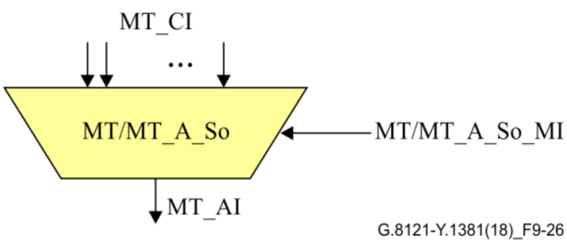


Figure 9-26 – MT/MT_A_So function

Interfaces

The interfaces are described in Table 9-10.

Table 9-10 – MT/MT_A_So interfaces

Input(s)	Output(s)
Each MT_CP: MT_CI_Data MT_CI_iPHB MT_CI_oPHB **MT_CP**: MT_CI_APS **MT/MT_A_So_MI**: MT/MT_A_So_MI_Admin_State MT/MT_A_So_MI_Label[1…M] MT/MT_A_So_MI_LSPType[1…M] MT/MT_A_So_MI_CoS[1…M] MT/MT_A_So_MI_PHB2TCMapping[1…M] MT/MT_A_So_MI_QoSEncodingMode[1…M] MT/MT_A_So_MI_Mode MT/MT_A_So_MI_LCK_Period[1…M] MT/MT_A_So_MI_LCK_CoS[1…M] MT/MT_A_So_MI_LCK_OAM_Tool[1...M] MT/MT_A_So_MI_GAL_Enable[1...M] MT/MT_A_So_MI_APS_CoS MT/MT_A_So_MI_APS_OAM_Tool	**MT_AP**: MT_AI_Data MT_AI_PHB

Processes

The processes associated with the MT/MT_A_So function are as depicted in Figure 9-27.

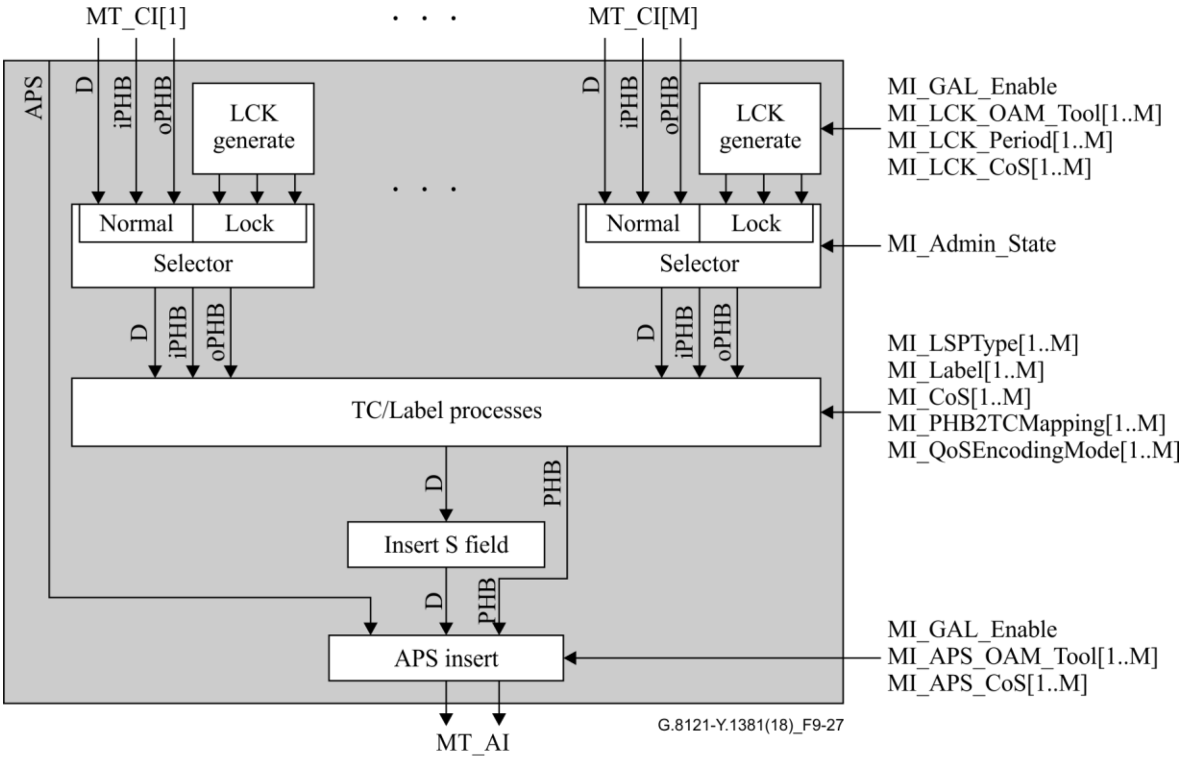


Figure 9-27 – MT/MT_A_So process diagram

– *LCK generation process:*

See clause 8.6.3. Each CP has its LCK generation process.

– *Selector process:*

See clause 8.6.1. The normal CI is blocked if Admin_State = LOCKED.

– *TC/Label processes:*

See 8.2.1.

– *S Field Insertion:*

A 1-bit S Field set to 0 (not bottom of label stack) is inserted to indicate the client is MPLS.

– *APS insert processes:*

See clause 8.7.2.1.

Defects: None.

Consequent actions: None.

Defect correlations: None.

Performance monitoring: None.

9.3.1.2 MPLS-TP to MPLS-TP adaptation sink function (MT/MT_A_Sk)

The MT/MT_A_Sk function symbol is shown in Figure 9-28. This function retrieves client MT_CI traffic units from server MT_AI traffic units.

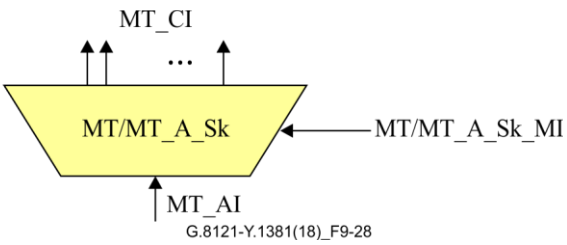


Figure 9-28 – MT/MT_A_Sk function

Interfaces

The interfaces are described in Table 9-11.

Table 9-11 – MT/MT_A_Sk interfaces

Input(s)	Output(s)
MT_AP: MT_AI_Data MT_AI_PHB MT_AI_TSF MT_AI_TSD MT_AI_AIS MT_AI_LStack **MT/MT_A_Sk_MP**: MT/MT_A_Sk_MI_AdminState MT/MT_A_Sk_MI_Label[1…M] MT/MT_A_Sk_MI_LSPType[1…M] MT/MT_A_Sk_MI_CoS[1…M] MT/MT_A_Sk_MI_ TC2PHBMapping[1…M] MT/MT_A_Sk_MI_QoSDecodingMode[1…M] MT/MT_A_Sk_MI_Mode MT/MT_A_Sk_MI_AIS_Period[1...M] MT/MT_A_Sk_MI_AIS_CoS[1...M] MT/MT_A_Sk_MI_AIS_OAM_Tool[1...M] MT/MT_A_Sk_MI_LCK_Period[1...M] MT/MT_A_Sk_MI_LCK_CoS[1...M] MT/MT_A_Sk_MI_LCK_OAM_Tool[1...M] MT/MT_A_Sk_MI_APS_OAM_Tool MT/MT_A_Sk_MI_GAL_Enable [1...M]	**Each MT_CP**: MT_CI_Data MT_CI_iPHB MT_CI_oPHB MT_CI_LStack MT_CI_SSF **MT_CP[1]**: MT_CI_SSD[1] MT_CI_APS[1]

Processes

The processes associated with the MT/MT_A_Sk function are as depicted in Figure 9-29.

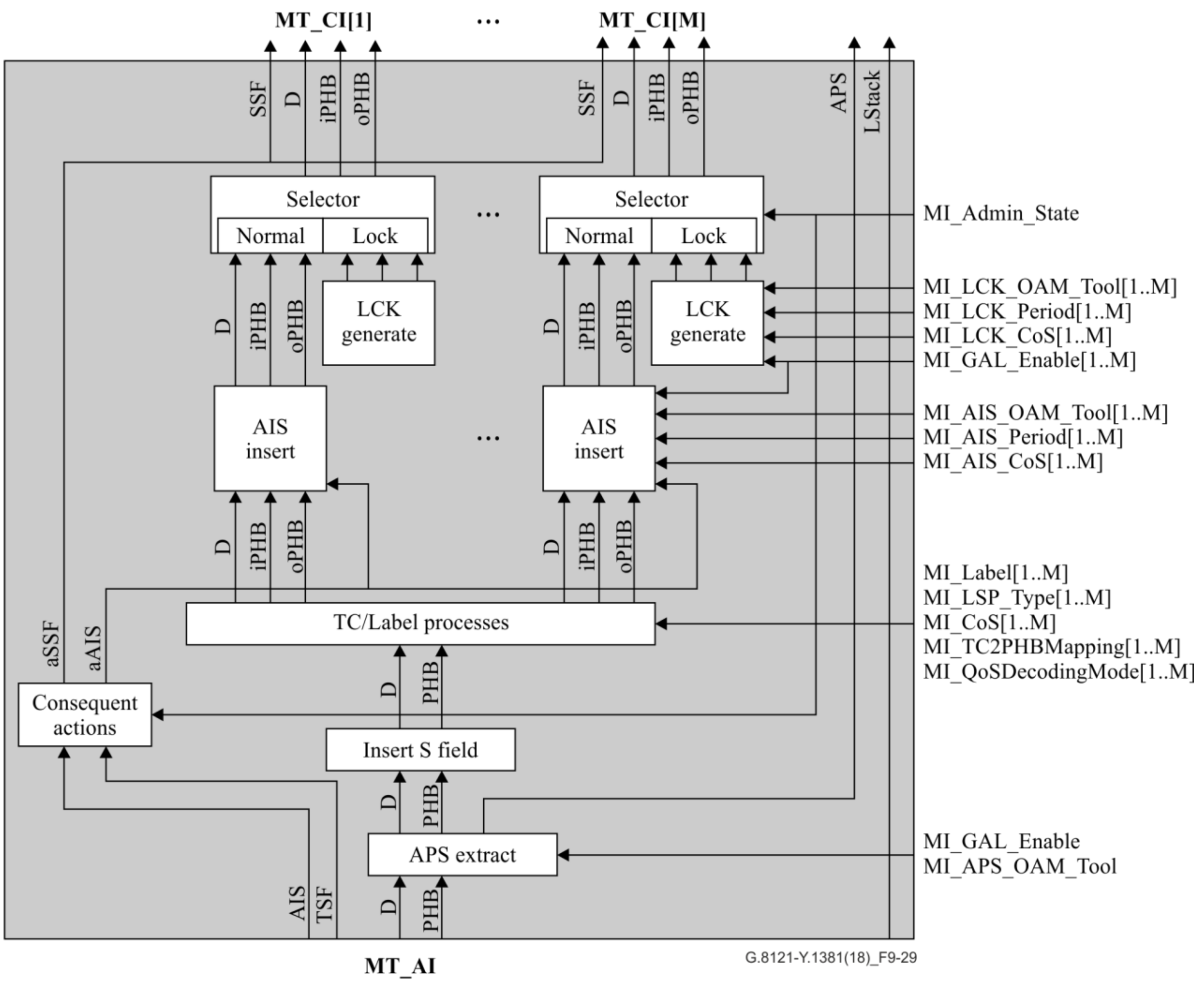


Figure 9-29 – MT/MT_A_Sk process diagram

– *Selector process:*

See clause 8.6.1. The normal CI is blocked if Admin_State = LOCKED.

– *LCK generation process:*

See clause 8.6.3.

– *AIS insert process:*

See clause 8.6.2.

– *TC/Label sink processes:*

See clause 8.2.2.

– *Label stack copy processes:*

See clause 8.2.3.

– *S field extraction:*

Extract and process the 1-bit S Field: the retrieved S Field should have the value 0 (not bottom of label stack) to indicate the client is MPLS; for such case the traffic unit is accepted and forwarded (together with the PHB information) after extraction of the S-bit field to the next process. For the case the S-bit has the value 1, the traffic unit is silently discarded.

– *APS extract processes:*

See clause 8.7.2.2.

Defects: None.

Consequent actions:

The function shall perform the following consequent actions:

aSSF ← AI_TSF

aSSD ← AI_TSD

aAIS ← AI_AIS

Defect correlations: None.

Performance monitoring: None.

9.3.2 MPLS-TP to MPLS-TP adaptation function for trail protection (MT/MTp_A)

The MT/MTp_A function is used for trail protection.

The bidirectional MT/MTp_A is performed by a co-located pair of MT/MTp_A source (MT/MTp_A_So) and sink (MT/MTp_A_Sk) functions.

9.3.2.1 MPLS-TP to MPLS-TP adaptation source function for Trail Protection (MT/MTp_A_So)

Symbol

The MT/MTp_A_So function symbol is shown in Figure 9-30. This function maps client MTp_CI traffic units into server MT_AI traffic units.

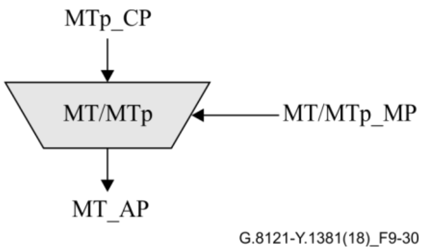


Figure 9-30 – MT/MTp_A_So function

Interfaces

The interfaces are described in Table 9-12.

Table 9-12 – MT/MTp_A_So interfaces

Input(s)	Output(s)
MT_CP: MTp_CI_Data MTp_CI_PHB MTp_CI_APS **MT/MTp_A_So_MI**: MT/MTp_A_So_MI_GAL_Enable MT/MTp_A_So_MI_APS_CoS MT/MTp_A_So_MI_APS_OAM_Tool	**MT_AP**: MT_AI_Data MT_AI_PHB

Processes

The processes associated with the MT/MTp_TT_So function are as depicted in Figure 9-31.

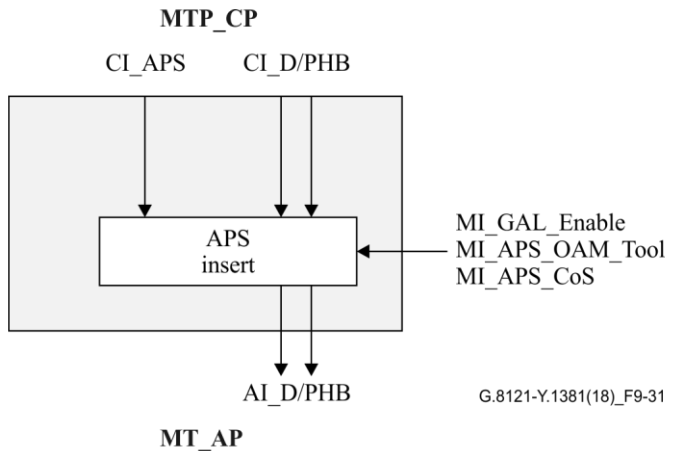


Figure 9-31 – MT/MTp_A_So process diagram

– *APS insert process:*

See clause 8.7.2.1.

Defects: None.

Consequent actions: None.

Defect correlations: None.

Performance monitoring: None.

9.3.2.2 MPLS-TP to MPLS-TP adaptation sink function for trail protection (MT/MTp_A_Sk)

The MT/MTp_A_Sk function symbol is shown in Figure 9-32. This function retrieves client MTp_CI traffic units from the server MT_AI traffic units.

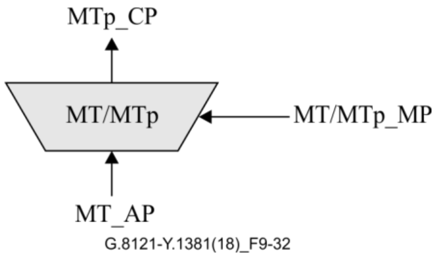


Figure 9-32 – MT/MTp_A_Sk function

Interfaces

The interfaces are described in Table 9-13.

Table 9-13 – MT/MTp_A_Sk interfaces

Input(s)	Output(s)
MT_AP: MT_AI_Data MT_AI_PHB MT_AI_TSF MT_AI_TSD MT_AI_AIS MT_AI_LStack **MT/MTp_A_Sk_MP**: MT/MTp_A_Sk_MI_APS_OAM_Tool MT/MTp_A_Sk_MI_GAL_Enable	**MTp_CP**: MTp_CI_Data MTp_CI_PHB MTp_CI_SSF MTp_CI_SSD MTp_CI_AIS MTp_CI_LStack MTp_CI_APS

Processes

The processes associated with the MT/MT_A_Sk function are as depicted in Figure 9-33.

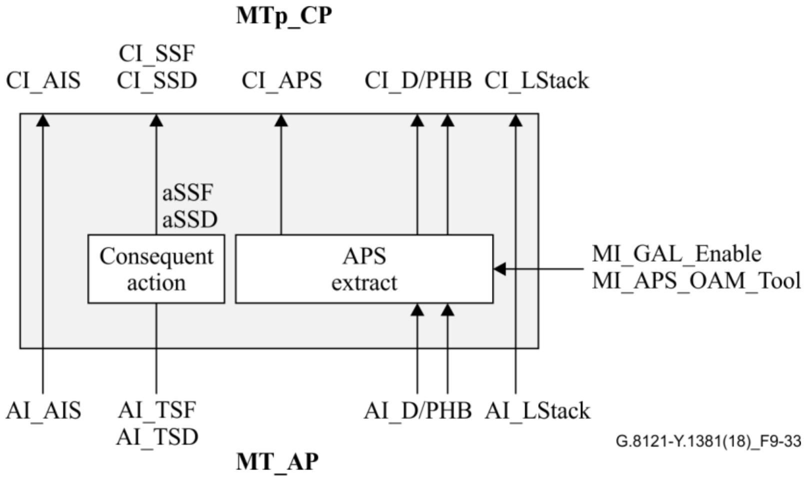


Figure 9-33 – MT/MTp_A_Sk process diagram

– *APS extract process:*

See clause 8.7.2.2.

Defects: None.

Consequent actions:

The function shall perform the following consequent actions:

aSSF ← AI_TSF

aSSD ← AI_TSD

Defect correlations: None.

Performance monitoring: None.

9.3.3 MPLS-TP to MSRP adaptation function (MT/MSRP_A)

9.3.3.1 MPLS-TP to MSRP adaptation source function (MT/MSRP_A_So)

Symbol

The MT/MSRP_A_So function symbol is shown in Figure 9-34. This function maps client MSRP _CI traffic units into server MT_AI traffic units.

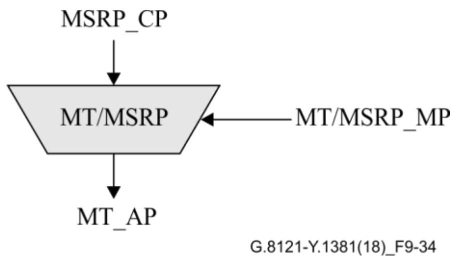


Figure 9-34 – MT/MSRP_A_So function

Interfaces

The interfaces are described in Table 9-14.

Table 9-14 – MT/MSRP_A_So interfaces

Input(s)	Output(s)
MSRP_CP: MSRP_CI_Data MSRP_CI_PHB MSRP_CI_APS **MT/ MSRP_A_So_MI**: MT/MSRP_A_So_MI_GAL_Enable MT/MSRP_A_So_MI_APS_CoS MT/MSRP_A_So_MI_APS_OAM_Tool	**MT_AP**: MT_AI_Data MT_AI_PHB

Processes

The processes associated with the MT/MSRP_TT_So function are as depicted in Figure 9-35.

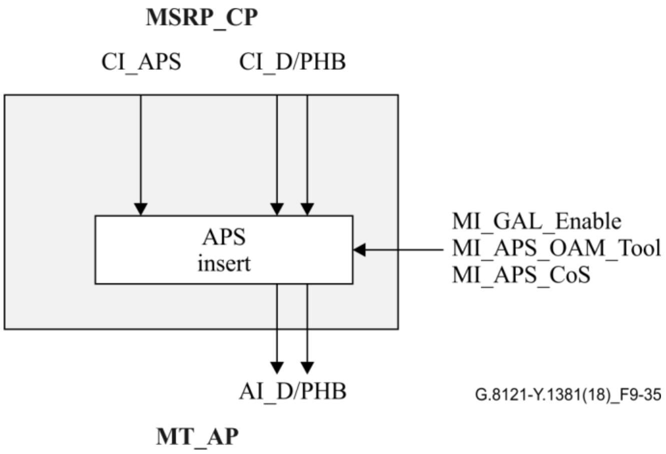


Figure 9-35 – MT/MSRP_A_So process diagram

– *APS insert process:*

See clause 8.7.2.1.

Defects: None.

Consequent actions: None.

Defect correlations: None.

Performance monitoring: None.

9.3.3.2 MPLS-TP to MSRP adaptation sink function (MT/MSRP_A_Sk)

The MT/MSRP_A_Sk function symbol is shown in Figure 9-36. This function extracts client MSRP_CI traffic units from the server MT_AI traffic units.

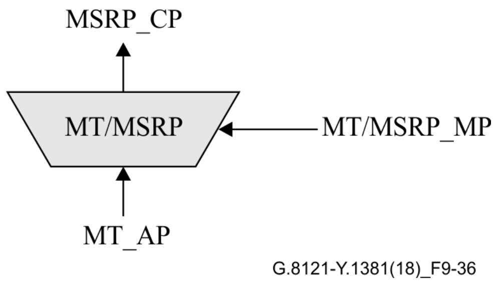


Figure 9-36 – MT/MSRP_A_Sk function

Interfaces

The interfaces are described in Table 9-15.

Table 9-15 – MT/MSRP_A_Sk interfaces

Input(s)	Output(s)
MT_AP: MT_AI_Data MT_AI_PHB MT_AI_TSF MT_AI_LStack **MT/MSRP_A_Sk_MP**: MT/MSRP_A_Sk_MI_APS_OAM_Tool MT/MSRP_A_Sk_MI_GAL_Enable	**MSRP_CP**: MSRP_CI_Data MSRP_CI_PHB MSRP_CI_SSF MSRP_CI_LStack MSRP_CI_APS

Processes

The processes associated with the MT/MSRP_A_Sk function are as depicted in Figure 9-37.

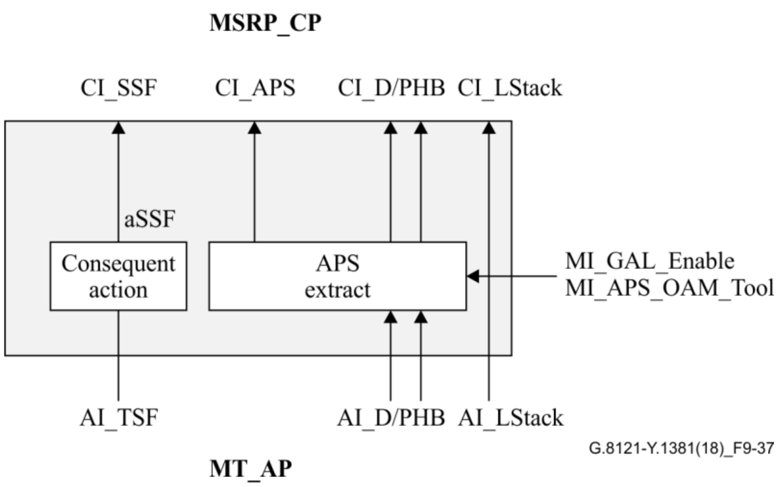


Figure 9-37– MT/MSRP_A_Sk process

9.3.4 MSRP to MPLS-TP adaptation function (MSRP/MT_A)

9.3.4.1 MSRP to MPLS-TP adaptation source function (MSRP/MT_A_So)

Symbol

The MSRP/MT_A_So function symbol is shown in Figure 9-38.

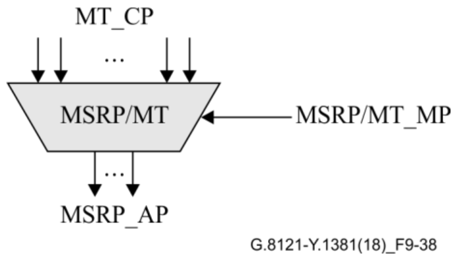


Figure 9-38 – MSRP/MT_A_So function

Interfaces

The interfaces are described in Table 9-16.

Table 9-16 – MSRP/MT_A_So interfaces

Input(s)	Output(s)
per MT_CP[1..n]: MSRP_CI_Data MSRP_CI_iPHB MSRP_CI_oPHB MSRP_CI_LSTACK **MT/MSRP_A_So_MI**: MT/MSRP_MI_DestNode_ID[1…m]	**per MT_AP[1…m]**: MT_AI_Data MT_AI_PHB
NOTE – 'n' means number of LSPs to MSRP sublayer. 'm' means number of nodes in a ring.	

Processes

The processes associated with the MSRP/MT_A_So function are as depicted in Figure 9-39.

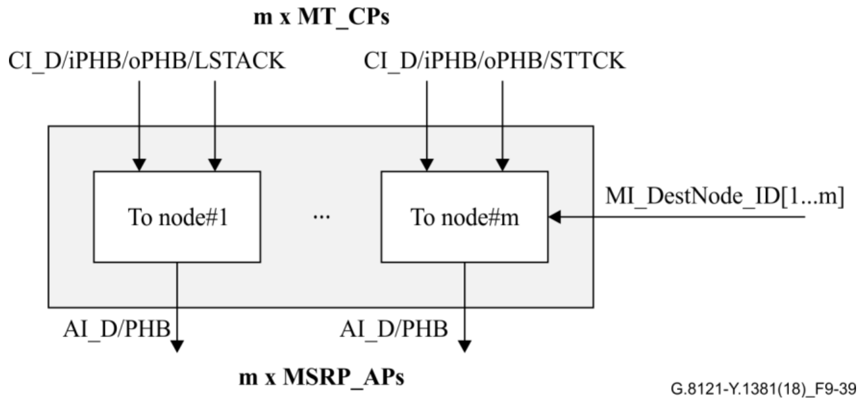


Figure 9-39 – MSRP/MT_A_So process

– *To node#i process:*

This process enables each MPLS-TP LSP traffic to go to each ring node.

9.3.4.2 MSRP to MPLS-TP adaptation sink function (MSRP/MT_A_Sk)

Symbol

The MSRP/MT_A_Sk function symbol is shown in Figure 9-40.

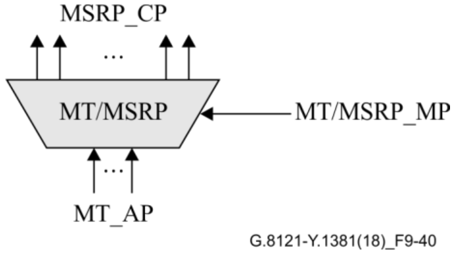


Figure 9-40– MSRP/MT_A_Sk function

Interfaces

The interfaces are described in Table 9-17.

Table 9-17 – MSRP/MT_A_Sk interfaces

Input(s)	Output(s)
per MT_AP[1…m]: MT_AI_Data MT_AI_PHB	**per MT_CP[1..n]**: MSRP_CI_Data MSRP_CI_iPHB MSRP_CI_oPHB MSRP_CI_LSTACK
NOTE – 'n' means number of LSPs from MSRP sublayer. 'm' means number of nodes in a ring.	

Processes

The processes associated with the MSRP/MT_A_Sk function are as depicted in Figure 9-41.

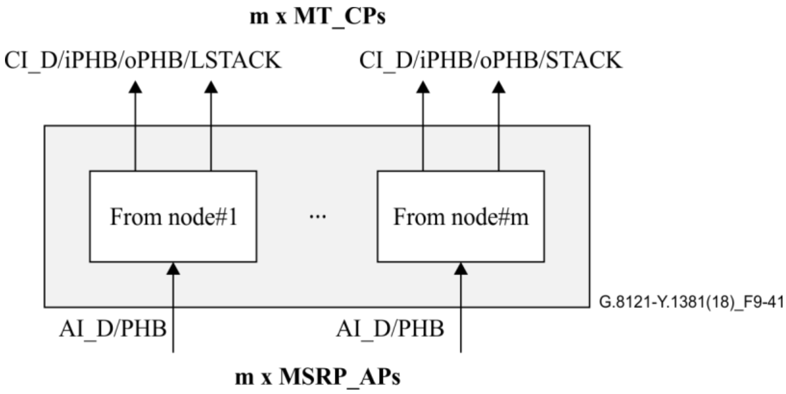


Figure 9-41 – MSRP/MT_A_Sk process

– *From node#i process:*

This process enables MPLS-TP LSPs from each ring traffic to go out of the ring.

9.4 MT diagnostic function

9.4.1 MT diagnostic functions for MEPs (MTDe)

9.4.1.1 MT diagnostic trail termination functions for MEPs (MTDe_TT)

The bidirectional MTDe trail termination (MTDe_TT) function is performed by a co-located pair of MTDe trail termination source (MTDe_TT_So) and sink (MTDe_TT_Sk) functions as shown in Figure 9-42.

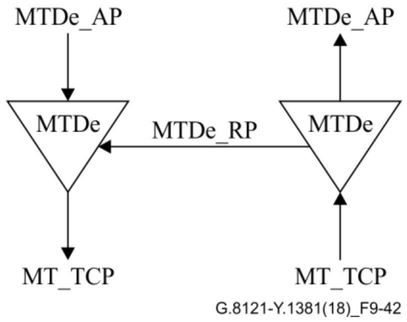


Figure 9-42 – MTDe_TT

9.4.1.1.1 MT diagnostic trail termination source function for MEPs (MTDe_TT_So)

Symbol

The MTDe_TT_So function symbol is shown in Figure 9-43.

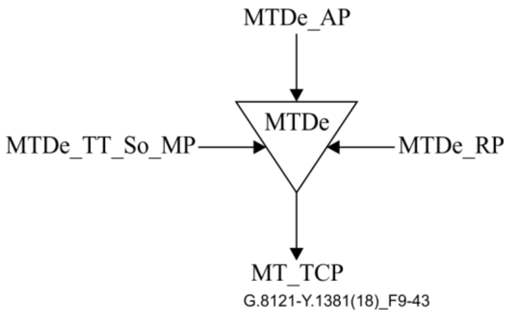


Figure 9-43 – MTDe_TT_So symbol

Interfaces

The interfaces are described in Table 9-18.

Table 9-18 – MTDe_TT_So interfaces

Input(s)	Output(s)
MTDe_AP: MTDe_AI_D MTDe_AI_oPHB MTDe_AI_iPHB **MT De_RP**: MTDe_RI_OAM_Info(D,CoS,DP) MTDe_RI_CI **MTDe_TT_So_MP**: MTDe_TT_So_MI_ GAL_Enable MTDe_TT_So_MI_TTLVALUE MTDe_TT_So_MI_CV_OAM_Tool MTDe_TT_So_MI_CV_Series () [Note]	**MT_TCP**: MT_CI_D MT_CI_oPHB MT_CI_iPHB **MTDe_TT_So_MP**: MTDe_TT_So_MI_CV_Series_Result[Note] MTDe_TT_So_MI_1TH_Result(Sent) MTDe_TT_So_MI_LMo_Result(N_TF,N_LF,F_TF,F_LF)[1...M_{LMo}] MTDe_TT_So_MI_DMo_Result(count,B_FD[],F_FD[],N_FD[])[1...M_{DMo}] MTDe_TT_So_MI_SLo_Result(N_TF,N_LF,F_TF,F_LF)[1...M_{SLo}]

Table 9-18 – MTDe_TT_So interfaces

Input(s)	Output(s)
MTDe_TT_So_MI_1TH_OAM_Tool MTDe_TT_So_MI_1TH_Start (CoS,Length,Period) MTDe_TT_So_MI_1TH_Terminate MTDe_TT_So_MI_ LMo_OAM_Tool MTDe_TT_So_MI_LMo_Start(CoS,Period) [1...M_{LMo}] MTDe_TT_So_MI_LMo_Intermediate_Request[1...M_{LMo}] MTDe_FT_So_MI_LMo_Terminate[1...M_{LMo}] MTDe_TT_So_MI_ DMo_OAM_Tool MTDe_TT_So_MI_DMo_Start (CoS,Test_ID,Length,Period)[1...M_{DMo}] MTDe_TT_So_MI_DMo_Intermediate_Request[1...M_{LMo}] MTDe_TT_So_MI_DMo_Terminate[1...M_{DMo}] MTDe_TT_So_MI_ 1DMo_OAM_Tool MTDe_TT_So_MI_1DMo_Start (CoS,Test_ID,Length,Period)[1...M_{1DMo}] MTDe_TT_So_MI_1DMo_Terminate[1...M_{1DMo}] MTDe_TT_So_MI_ SLo_OAM_Tool MTDe_TT_So_MI_SLo_Start (CoS,Test_ID,Length,Period)[1...M_{SLo}] MTDe_TT_So_MI_SLo_Intermediate_Request[1...M_{LMo}] MTDe_TT_So_MI_SLo_Terminate[1...M_{SLo}] MTDe_TT_So_MI_Admin_State MTDe_TT_So_MI_Lock_Instruct_Enable MTDe_TT_So_MI_DP_Loopback_Enable **MTDe_TT_So_TP**: MTDe_TT_So_TI_ TimeStampl	
NOTE: The parameters for MI_CV_Series and MI_CV_ Series_ Result are not within the scope of this Recommendation.	

Processes

The processes associated with the MTDe_TT_So function are as depicted in Figure 9-44.

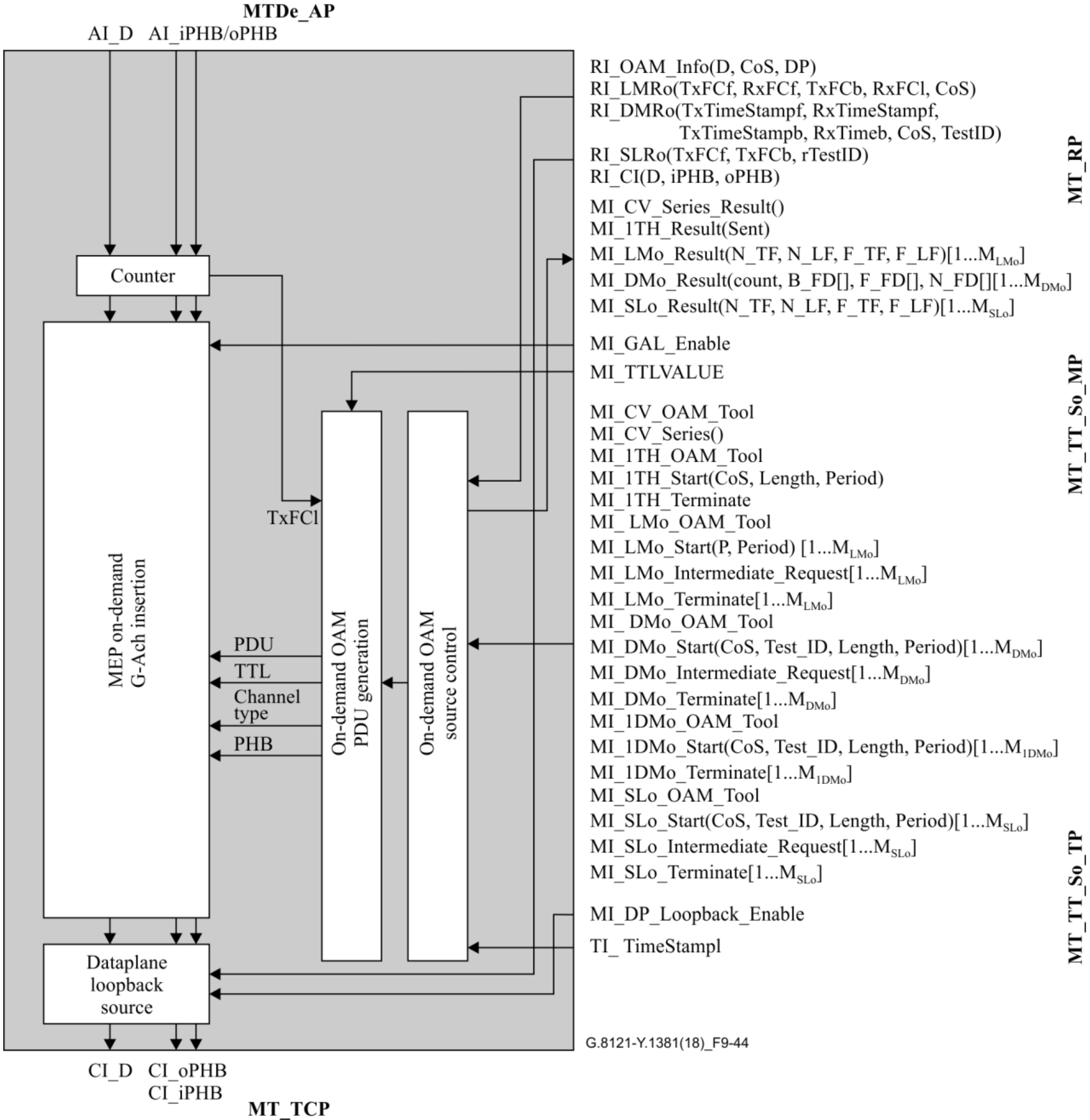


Figure 9-44 – MTDe_TT_So process

MEP on-demand G-ACh insertion: See clause 8.1.2.

OAM PDU generation: See clause 8.8.

On-demand OAM source control: See clause 8.8.

Dataplane loopback source process: See clause 8.9.2.

Counter: This process is used to count packets for on-demand loss measurements. The location of the counter process is shown for illustration only. The exact set of packets to be counted is outside the scope of this Recommendation.

Defects: None.

Consequent actions: None.

Defect correlations: None.

Performance monitoring: None.

9.4.1.1.2 MT diagnostic trail termination sink function for MEPs (MTDe_TT_Sk)

Symbol

The MTDe_TT_Sk function symbol is shown in Figure 9-45.

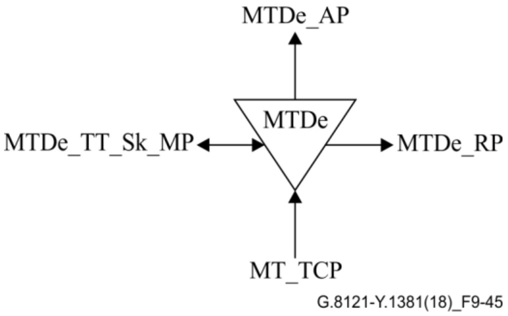


Figure 9-45 – MTDe_TT_Sk symbol

Interfaces

The interfaces are described in Table 9-19.

Table 9-19 – MTDe_TT_Sk interfaces

Input(s)	Output(s)
MT_TCP: MT_CI_D MT_CI_iPHB MT_CI_oPHB MT_CI_LStack **MTDe_TT_Sk_MP**: MTDe_TT_Sk_MI_ GAL_Enable MTDe_TT_Sk_MI_CV_OAM_Tool MTDe_TT_Sk_MI_1TH_OAM_Tool MTDe_TT_Sk_MI_1TH_Start MTDe_TT_Sk_MI_1TH_Terminate MTDe_TT_Sk_MI_ LMo_OAM_Tool MTDe_TT_Sk_MI_ DMo_OAM_Tool MTDe_TT_Sk_MI_ 1DMo_OAM_Tool MTDe_TT_Sk_MI_1DMo_Start(Test_ID, CoS)[1...M_{1DMo}] MTDe_TT_Sk_MI_1DMo_Intermediate_Request[1...M_{LMo}] MTDe_TT_Sk_MI_1DMo_Terminate[1...M_{1DMo}] MTDe_TT_Sk_MI_ SLo_OAM_Tool MTDe_TT_Sk_MI_DP_Loopback_Enable **MTDe_TP**: MTDe_TT_Sk_TI_ TimeStampl	**MTDe_AP**: MTDe_AI_D MTDe_AI_oPHB MTDe_AI_iPHB MTDe_AI_LStack **MTDe_RP**: MTDe_RI_OAM_Info(D,CoS,DP) MTDe_RI_CI **MTDe_TT_Sk_MP**: MTDe_TT_Sk_MI_1TH_Result(REC,CRC,BER,OO) MTDe_TT_Sk_MI_1DMo_Result(count,N_FD[])[1...M_{DMo}]] MTDe_TT_Sk_MI_Admin_State_Request

Processes

The processes associated with the MTDe_TT_Sk function are as depicted in Figure 9-46.

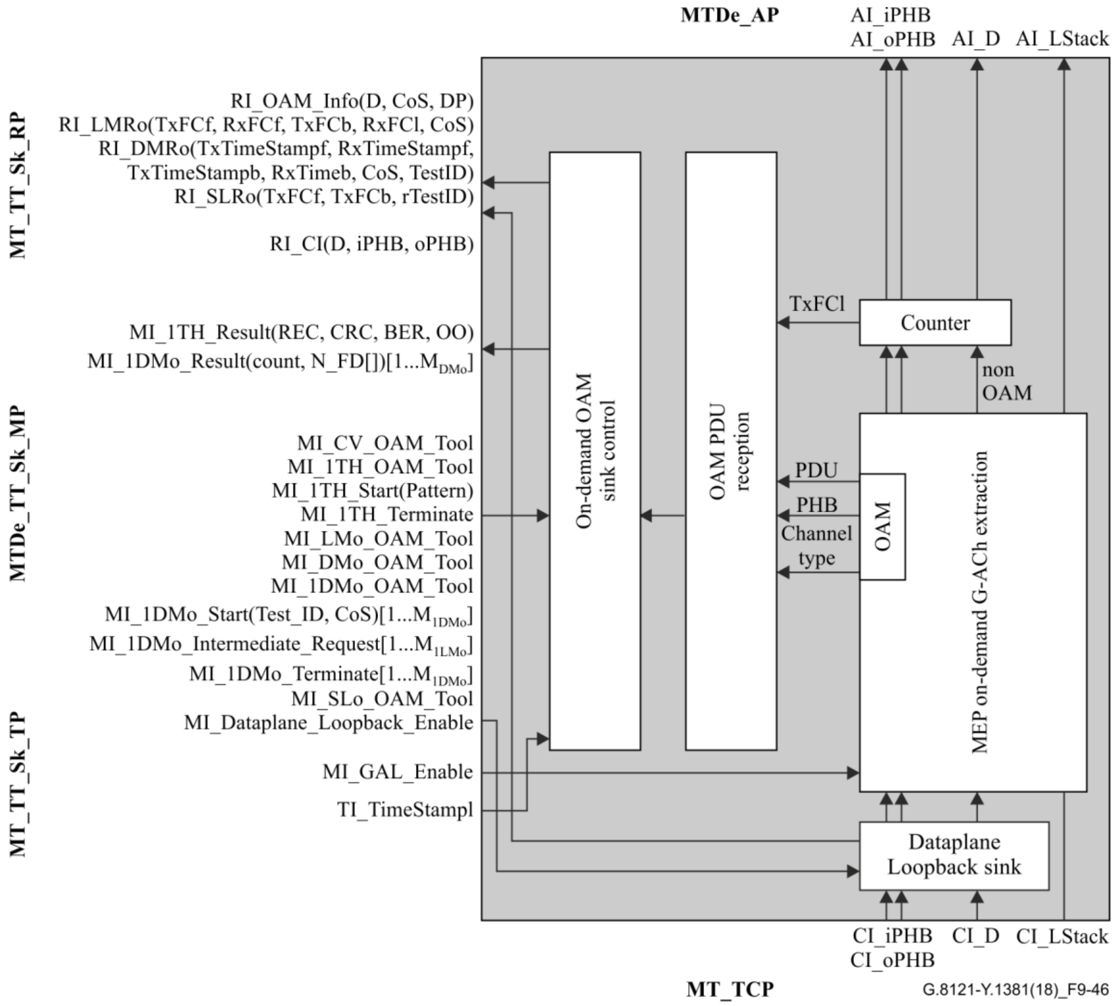


Figure 9-46 – MTDe_TT_Sk process

MEP On-demand G-ACh extraction: See clause 8.1. The process of CI_D is made according to the following rule:

```
if ( ((MI_GAL_Enable && MT-label(D) == GAL) ||
      (!MI_GAL_Enable && 1stNibble(D) == 0b0001)) &&
        (Packet_Type(D) == On-demand_OAM) )
    forward to G-ACh port
} else {
    forward to data port
}
```

NOTE – MT-label(D) and 1stNibble(D) are functions which return, respectively, the Label field at the top of the stack within the MPLS payload, the first nibble of the MPLS payload as defined in Figure 6-3 of [ITU-T G.8110.1].

These fields are used to identify G-ACh packets as described in clause 4.2.1 of [IETF RFC 5586].

Packet_Type(D) is a protocol-specific function that determines whether the traffic unit contains an on-demand OAM packet.

OAM PDU reception: See clause 8.8

On-demand OAM sink control: See clause 8.8

Dataplane loopback sink process: See clause 8.9.1

Counter: This process is used to count packets for on-demand loss measurements. The location of the counter process is shown for illustration only. The exact set of packets to be counted is outside the scope of this Recommendation.

Defects: None.

Consequent actions: None.

Defect correlations: None.

Performance monitoring: None.

9.4.1.2 MTDe to MT adaptation functions (MTDe/MT_A)

9.4.1.2.1 MTDe to MT adaptation source function (MTDe/MT_A_So)

This function consists of input MT_CI and output MTDe_AI. The function inside is empty, i.e., the input signals are simply passed to the output.

9.4.1.2.2 MTDe to MT adaptation sink function (MTDe/MT_A_Sk)

This function consists of input MTDe_AI and output MT_CI. The function inside is empty, i.e., the input signals are simply passed to the output.

9.4.2 MT diagnostic functions for MIPs (MTDi)

9.4.2.1 MT diagnostic trail termination functions for MIPs (MTDi_TT)

9.4.2.1.1 MT diagnostic trail termination source function for MIPs (MTDi_TT_So)

Symbol

The MTDi_TT_So function symbol is shown in Figure 9-47

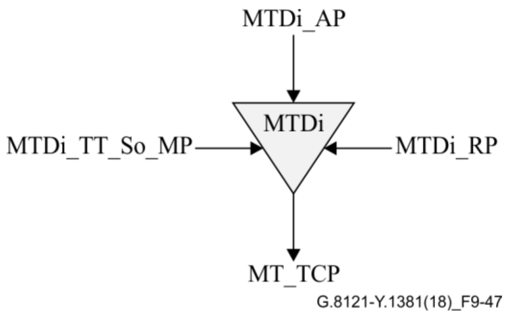


Figure 9-47 – MTDi_TT_So symbol

Interfaces

The interfaces are described in Table 9-20.

Table 9-20 – MTDi_TT_So interfaces

Input(s)	Output(s)
MTDi_AP MTDi_AI_D MTDi_AI_iPHB MTDi_AI_oPHB MTDi_AI_Lstack **MTDi_RP** MTDi_RI_OAM_Info (D, CoS, DP) MTDi_RI_CI **MTDi_TT_So_MP** MTDi_TT_So_MI_GAL_Enable MTDi_TT_So_MI_TTLVALUE MTDi_TT_So_MI_MIP_ID MTDi_TT_So_MI_CV_OAM_Tool MTDi_TT_So_MI_DP_Loopback_Enable	**MT_TCP** MT_CI_D, MT_CI_iPHB, MT_CI_oPHB, MT_CI_LStack

Processes

The processes associated with the MTDi_TT_So function are as depicted in Figure 9-48.

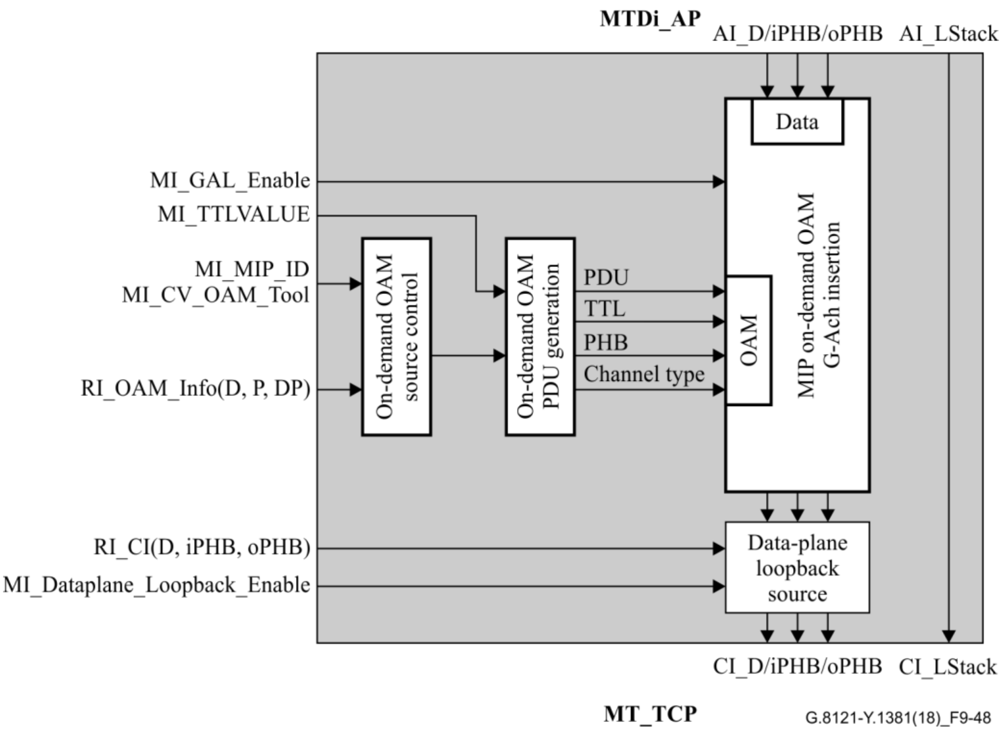


Figure 9-48 – MTDi_TT_So Process

MIP On-demand OAM G-ACh insertion: The MIP OAM insertion process inserts OAM traffic units that are generated in the MTDi_TT_So process into the stream of traffic units.

The GAL is used or not according to the MI_GAL_Enable parameter.

OAM PDU generation: See clause 8.8.

On-demand OAM source control: See clause 8.8.

Dataplane loopback source process: See clause 8.9.2

Defects: None.

Consequent actions: None.

Defect correlations: None.

Performance monitoring: None.

9.4.2.1.2 MT diagnostic trail termination sink function for MIPs (MTDi_TT_Sk)

Symbol

The MTDi_TT_Sk function symbol is shown in Figure 9-49.

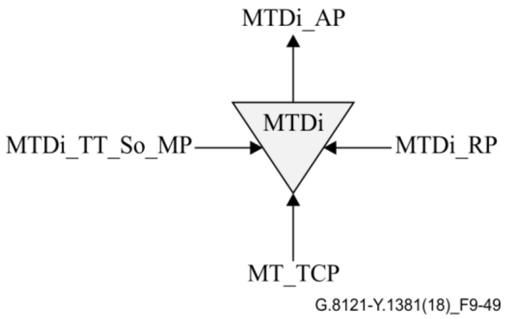


Figure 9-49 – MTDi_TT_Sk symbol

Interfaces

The interfaces are described in Table 9-21.

Table 9-21 – MTDi_TT_Sk interfaces

Input(s)	Output(s)
MT_TCP MT_CI_D MT_CI_iPHB MT_CI_oPHB MT_CI_LStack **MTDi_TT_Sk_MP** MTDi_TT_Sk_MI_GAL_Enable MTDi_TT_Sk_MI_MIP_ID MTDi_TT_Sk_MI_CV_OAM_Tool MTDi_TT_Sk_MI_DP_Loopback_Enable	**MTDi_AP** MTDi_AI_D MTDi_AI_iPHB MTDi_AI_oPHB MTDi_AI_LStack **MTDi_RP** MTDi_RI_OAM_Info (D, CoS, DP) MTDi_RI_CI

Processes

The processes associated with the MTDi_TT_Sk function are as depicted in Figure 9-50.

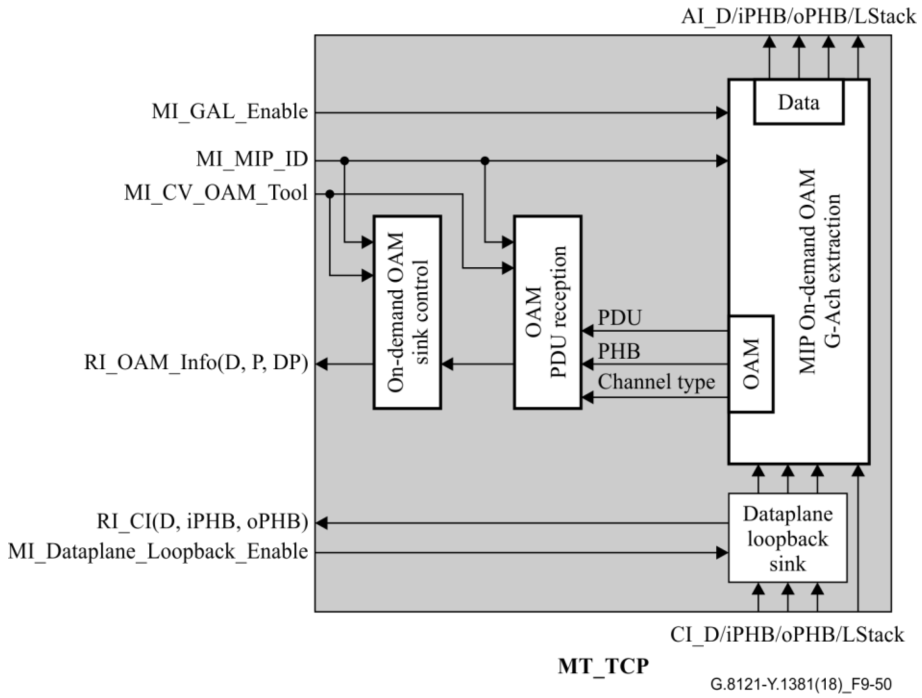


Figure 9-50 – MTDi_TT_Sk Process

MIP On-demand OAM G-ACh extraction: The MIP OAM extraction process classifies the OAM traffic units targeted to the MIP to which this MTDi_TT belongs, as configured by MI_MIP_ID, and delivers them to the On-demand OAM PDU reception process. All the other traffic units are delivered to MTDi_AP. The process is made according to the following rule:

```
if ( (TTL(D) == 0) &&
     ((MI_GAL_Enable && MT-label(D) == GAL) ||
        (!MI_GAL_Enable && 1stNibble(D) == 0b0001)) &&
         (Packet_Type(D) == OnDemandForThisMIP) )
     {
        forward to G-Ach port
     } else {
        forward to data port
     }
```

NOTE – For LSP and pseudowire MIPs, MT-label(D), 1stNibble(D) and TTL(D) are functions which return, respectively, the Label field at the top of the stack within the MPLS payload, the first nibble of the MPLS payload and the TTL field as defined in Figure 6-3 of [ITU-T G.8110.1].

These fields are used to identify G-ACh packets as described in clause 4.2.1 of [IETF RFC 5586].

Packet_Type(D) is a protocol-specific function that determines whether the traffic unit contains an on-demand OAM packet to be processed by this MIP.

OAM PDU reception: See clause 8.8.

On-demand OAM sink control: See clause 8.8.

Dataplane loopback sink process: See clause 8.9.1

Defects: None.

Consequent actions: None.

Defect correlations: None.

Performance monitoring: None.

9.4.2.2 MTDi to MT adaptation functions (MTDi/MT_A)

The MTDi/MT adaptation function is an empty function; it is included to satisfy the modelling rules.

The bidirectional MTDi/MT adaptation function is performed by a co-located pair of MTDi/MT adaptation source (MTDi/MT_A_So) and sink (MTDi/MT_A_Sk) functions.

9.4.2.2.1 MTDi to MT adaptation source functions (MTDi/MT_A_So)

Symbol

The MTDi/MT_A_So function symbol is shown in Figure 9-51.

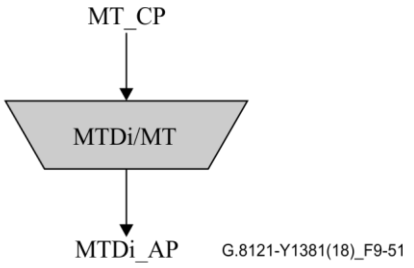


Figure 9-51 – MTDi/MT_A_So symbol

Interfaces

The interfaces are described in Table 9-22.

Table 9-22 – MTDi/MT_A_So interfaces

Input(s)	Output(s)
MT_CP: MT_CI_D MT_CI_iPHB MT_CI_oPHB MT_CI_LStack	**MTDi_AP**: MTDi_AI_D MTDi_AI_iPHB MTDi_AI_oPHB MTDi_AI_LStack

Processes

The processes associated with the MTDi/MT_So function are as depicted in Figure 9-52.

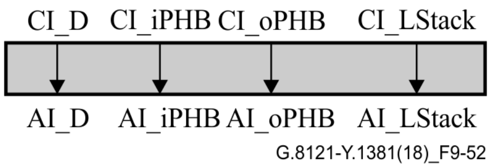


Figure 9-52 – MTDi/MT_A_So process

Defects: None.

Consequent actions: None.

Defect correlations: None.

Performance monitoring: None.

9.4.2.2.2 MTDi to MT adaptation sink function (MTDi/MT_A_Sk)

Symbol

The MTDi/MT_A_Sk function symbol is shown in Figure 9-53.

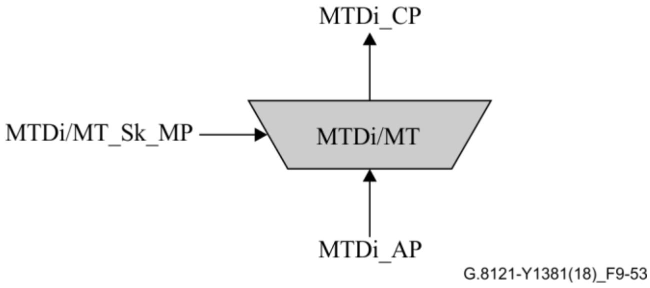


Figure 9-53 – MTDi/MT_A_Sk symbol

Interfaces

The MTDi/MT_A_Sk interfaces are described in Table 9-23.

Table 9-23 – MTDi/MT_A_Sk interfaces

Input(s)	Output(s)
MTDi_AP: MTDi_AI_D MTDi_AI_iPHB MTDi_AI_oPHB MTDi_AI_LStack **MTDi/MT_Sk_MP**: MIDi/MT_A_MI_DS_MP_Type	**MT_CP**: MT_CI_D MT_CI_iPHB MT_CI_oPHB MT_CI_LStack

Processes

The processes associated with the MTDi/MT_A_Sk function are as depicted in Figure 9-54.

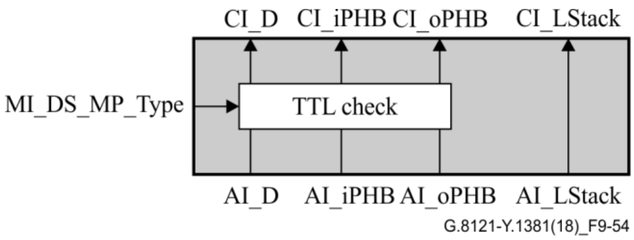


Figure 9-54 – MTDi/MT_A_Sk process

TTL check process:

For LSP and pseudowire MIPs, the TTL check process examines the TTL field in the MTDi_AI_D traffic unit. TTL check process drops all MPLS-TP packets with TTL = 0 by default (MI_DS_MP_Type set to none).

When MI_DS_MP_Type is set to MIP, TTL check process drops only user data MPLS-TP packets with TTL = 0 while OAM packets with TTL = 0 are not dropped in this process and are forwarded.

When the MI_DS_MP_Type is set to MEP, TTL check process does not block any MPLS-TP packet with TTL = 0: all MPLS-TP packets with TTL = 0 are forwarded.

NOTE – The MI_DS_MP_Type parameter should be properly configured by the EMF on the basis of the MPLS-TP connection configuration within the node and not exposed to the operator as a configuration parameter of the equipment management interface. Examples of MI_DS_MP_Type configuration are described in Appendix I.

Defects: None.

Consequent actions: None.

Defect correlations: None.

Performance monitoring: None.

9.5 MPLS-TP MEP and MIP functions

MEP and MIP compound functions are defined in [ITU-T G.806]. This clause specifies the composition of those functions with MT termination, adaptation and diagnostic atomic functions described in clauses 9.2, 9.3 and 9.4 respectively.

9.5.1 MPLS-TP NCM MEP function

An MPLS-TP network connection monitoring (NCM) MEP function is capable of originating, filtering and terminating proactive MPLS-TP OAM signals and originating, responding to and terminating diagnostic MPLS-TP OAM signals at the MPLS-TP layer. The NCM MEP is composed of MT_TT, MTDe/MT_A and MTDe_TT atomic functions. This MEP is located at the MPLS-TP layer boundary and connected with MT/client_A or MT/MT_A.

Figure 9-55 illustrates MT NCM MEP compound functions.

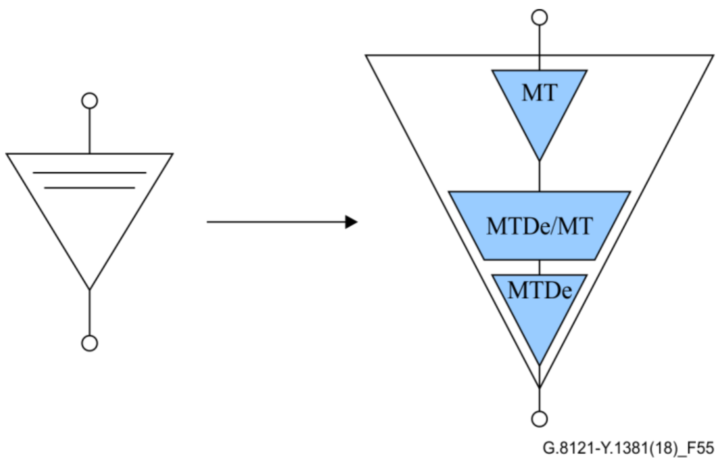


Figure 9-55 – MT NCM MEP compound functions

9.5.2 MPLS-TP TCM MEP function

Tandem connection monitoring (TCM) can be supported by the instantiation of sub-path maintenance entity (SPME), as described in [IETF RFC 6371], that has a 1:1 relationship with the monitored connection.

A MPLS-TP TCM MEP function is capable of originating, filtering and terminating proactive MPLS-TP OAM signals and originating, responding to and terminating diagnostic MPLS-TP OAM signals at the MPLS-TP sub-layer. The TCM MEP is composed of MT_TT, MTDe/MT_A and MTDe_TT atomic functions.

Figure 9-56 illustrates MT TCM MEP compound functions.

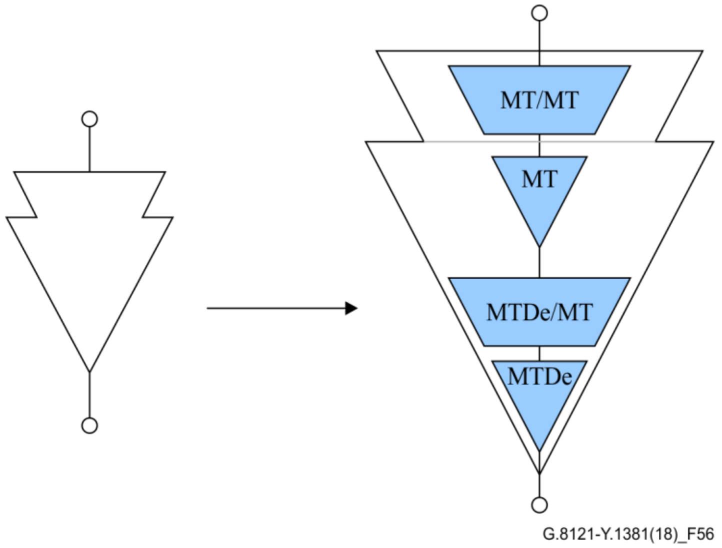


Figure 9-56 – MT TCM MEP compound functions

9.5.3 MT MIP function

An MT MIP function is capable of responding to on-demand MT OAM signals at one of the MPLS-TP (sub-)layers in both directions. The MIP combines two back-to-back half-MIP functions. It consists of two pairs of MTDi/MT_A and MTDi_TT atomic functions, each facing opposite directions.

Figure 9-57 illustrates MT MIP compound functions.

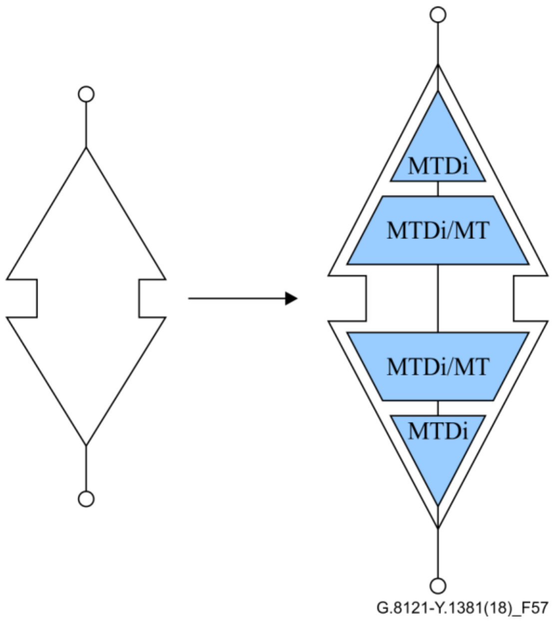


Figure 9-57 – MT MIP compound functions

9.5.4 MT half MIP function

An MPLS-TP half MIP function is capable of responding to on-demand MPLS-TP OAM signals at one of the MPLS-TP sublayers in a single direction. The half MIP is composed of a pair of MTDi/MT_A and MTDi_TT atomic functions.

Figure 9-58 illustrates MT half MIP compound functions.

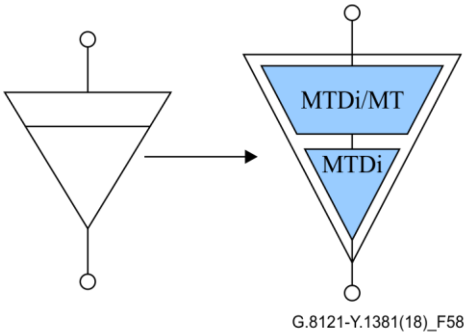


Figure 9-58 – MT half MIP compound functions

10 MPLS-TP to Non-MPLS-TP client adaptation functions

10.1 MPLS-TP to ETH adaptation function (MT/ETH_A)

The MPLS-TP to ETH adaptation functions that include CSF processes are specified in [ITU-T G.8121.1] and [ITU-T G.8121.2].

10.1.1 MPLS-TP to ETH adaptation source function (MT/ETH_A_So)

This function maps the ETH_CI information for transport in a MT_AI signal.

The information flow and processing of the MT/ETH_A_So function is defined with reference to Figure 10-1.

Symbol

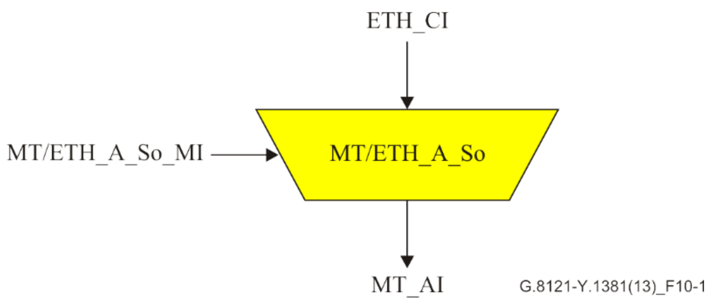


Figure 10-1 – MT/ETH_A_So function

Interfaces

The MT/ETH_A_So interfaces are described in Table 10-1.

Table 10-1 – MT/ETH_A_So inputs and outputs

Input(s)	Output(s)
ETH_FP: ETH_CI_Data ETH_CI_P ETH_CI_DE **MT/ETH_A_So_MP**: MT/ETH_A_So_MI_AdminState MT/ETH_A_So_MI_FCSEnable	**MT_AP**: MT_AI_Data MT_AI_PHB

Table 10-1 – MT/ETH_A_So inputs and outputs

Input(s)	Output(s)
MT/ETH_A_So_MI_CWEnable MT/ETH_A_So_MI_SQUse MT/ETH_A_So_MI_PRI2CoSMapping MT/ETH_A_So_MI_MEP_MAC* MT/ETH_A_So_MI_Client_MEL* MT/ETH_A_So_MI_LCK_Period* MT/ETH_A_So_MI_LCK_Pri* MT/ETH_A_So_MI_MEL* * ETH OAM related	

Processes

The processes associated with the MT/ETH_A_So function are as depicted in Figure 10-2.

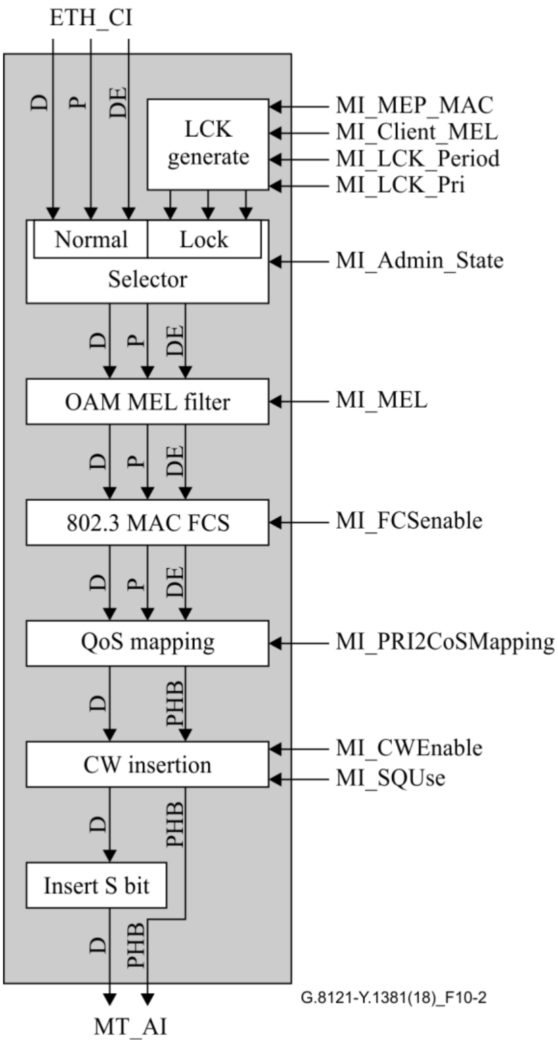


Figure 10-2 – MT/ETH_A_So process diagram

– *LCK generation process:*

See clause 8.1.2 of [ITU-T G.8021].

– *Selector process:*

See clause 8.1.3 of [ITU-T G.8021]. The normal CI is blocked if Admin_State = LOCKED.

– *OAM MEL filter process:*

See clause 8.1.1 of [ITU-T G.8021].

– *802.3 MAC FCS generation:*

See clause 8.8.1 of [ITU-T G.8021]. MAC FCS generation is optional (see [IETF RFC 4720] and [ITU-T Y.1415]): MAC FCS is generated if MI_FCSEnabled is true.

– *QoS mapping process:*

This process maps the Ethernet-based QoS signals into MPLS-based QoS signals.

The CoS part of the AI_PHB is generated by the received CI_P according to the 1:1 mapping configured by the MI_PRI2CoSMapping.

The DP part of the AI_PHB is generated by the received CI_DE according to the following rule:

```
If CI_DE = True
     DP(AI_PHB) = Yellow
Else
     DP(AI_PHB) = Green
```

– *CW insertion process:*

See clause 8.5.1.

– *Insert S bit process:*

A 1-bit S Field set to 1 (bottom of label stack) is inserted to indicate the client is not MPLS.

Defects: None.

Consequent actions: None.

Defect correlations: None.

Performance monitoring: None.

10.1.2 MPLS-TP to ETH adaptation sink function (MT/ETH_A_Sk)

This function extracts the ETH_CI information from a MT_AI signal.

The information flow and processing of the MT/ETH_A_Sk function is defined with reference to Figure 10-3.

Symbol

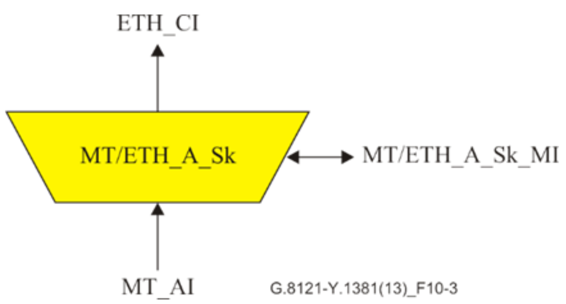


Figure 10-3 – MT/ETH_A_Sk function

Interfaces

The MT/ETH_A_Sk interfaces are described in Table 10-2.

Table 10-2 – MT/ETH_A_Sk Inputs and Outputs

Input(s)	Output(s)
Each MT_AP: MT_AI_Data MT_AI_PHB MT_AI_TSF MT_AI_AIS **MT/ETH_A_Sk_MP**: MT/ETH_A_Sk_MI_FCSEnable MT/ETH_A_Sk_MI_CWEnable MT/ETH_A_Sk_MI_SQUse MT/ETH_A_Sk_MI_GAL_Enable MT/ETH_A_Sk_MI_CoS2PRIMapping MT/ETH_A_Sk_MI_MEL* MT/ETH_A_Sk_MI_Admin_State MT/ETH_A_Sk_MI_LCK_Period * MT/ETH_A_Sk_MI_LCK_Pri * MT/ETH_A_Sk_MI_Client_MEL * MT/ETH_A_Sk_MI_MEP_MAC * MT/ETH_A_Sk_MI_AIS_Pri * MT/ETH_A_Sk_MI_AIS_Period * * ETH OAM related	**ETH_FP**: ETH_CI_Data ETH_CI_P ETH_CI_DE ETH_CI_SSF **MT/ETH_A_Sk_MP**: MT/ETH__A_Sk_MI_pFCSErrors

Processes

The processes associated with the MT/ETH_A_Sk function are as depicted in Figure 10-4.

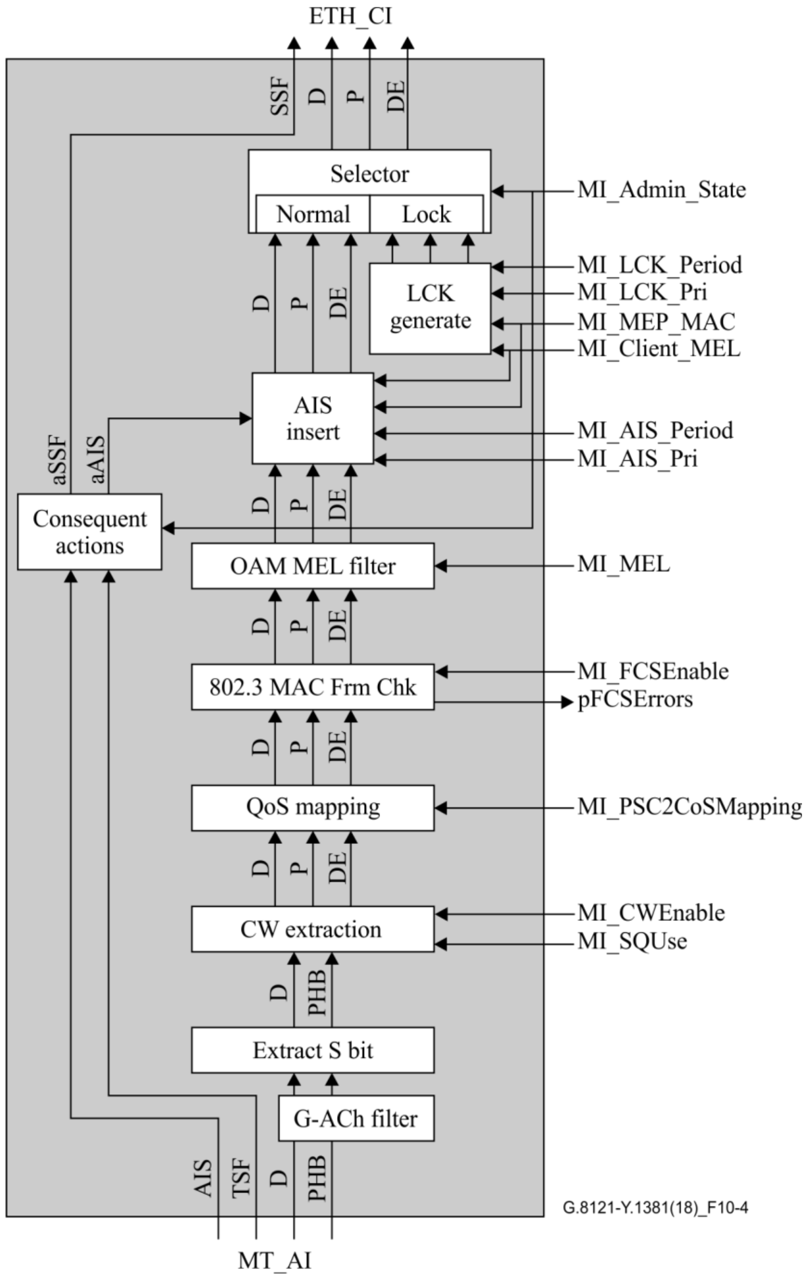


Figure 10-4 – MT/ETH_A_Sk process diagram

– *Selector process:*

See clause 8.1.3 of [ITU-T G.8021]. The normal CI is blocked if Admin_State = LOCKED.

– *LCK generate process:*

See clause 8.1.2 of [ITU-T G.8021].

– *AIS insert process:*

See clause 8.1.4 of [ITU-T G.8021].

– *OAM MEL filter process:*

See clause 8.1.1 of [ITU-T G.8021].

– *"802.3 MAC Frame Check" process:*

See clause 8.8.2 of [ITU-T G.8021]. MAC Frame Check is optional (see [IETF RFC 4720] and [ITU-T Y.1415]): MAC FCS is checked if MI_FCSEnabled is true.

– *QoS mapping process:*

This process maps the MPLS-based QoS signals into Ethernet-based QoS signals.

The CI_P is generated by the received PSC part of the AI_PHB according to the 1:1 mapping configured by the MI_CoS2PRIMapping.

The CI_DE is generated by the received DP part of the AI_PHB according to the following rule:

```
If DP(AI_PHB) = Green
     CI_DE = False
Else
     CI_DE = True
```

– *CW extraction process:*

See clause 8.5.2.

– *G-ACh filter process:*

This process removes all the received traffic units which are G-ACh encapsulated, which include GAL or not depending on the MI_GAL_Enable.

– *Extract S bit process:*

Extract and process the 1-bit S Field: the retrieved S Field should have the value 1 (bottom of label stack) to indicate the client is not MPLS: for such case the traffic unit is accepted and forwarded (together with the PHB information) after extraction of the S-bit field to the next process. For the case the S-bit has the value 0, the traffic unit is silently discarded.

Defects: None.

Consequent actions:

The function shall perform the following consequent actions:

aSSF ← AI_TSF and (not MI_Admin_State == LOCKED)

aAIS ← AI_AIS

Defect correlations: None.

Performance monitoring: For further study.

10.2 MPLS-TP to SCC and MCC adaptation functions

This clause provides the descriptions of the MPLS-TP adaptation functions for the MPLS-TP MCC and SCC.

Figure 10-5 shows the MPLS-TP adaptation functions providing access to the MCC and SCC. These MT/MCC and MT/SCC adaptation functions are defined in more detail below.

In case that the client is MPLS-TP, MT/Client will be MT/MT.

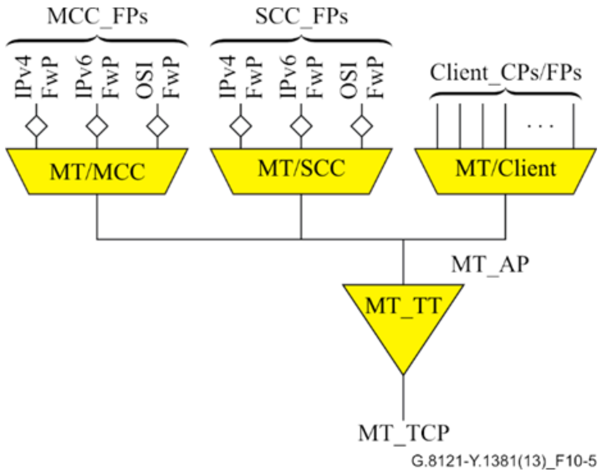


Figure 10-5 – MT/SCC_A function, MT/MCC_A function, and MT/client_A function

10.2.1 MT/SCC_A adaptation function

The MT to SCC adaptation function provides access to the SCC for signalling communication. It is used for the scenarios where the signalling communication network (SCN) utilizes the SCC as defined in [IETF RFC 5718].

10.2.1.1 MT to SCC adaptation source function (MT/SCC_A_So function)

The MT/SCC_A_So function maps the SCN data into the G-ACh SCC packets as defined in [IETF RFC 5718]. The diamonds in Figure 10-6 represent traffic shaping and conditioning functions that may be needed to prevent the SCC forwarding points from exceeding their committed bandwidth in congestion situations. These traffic shaping and conditioning functions as well as the related bandwidth management and bandwidth assignment functions are outside the scope of this Recommendation.

The information flow and processing of the MT/SCC_A_So functions is defined with reference to Figures 10-6 and 10-7.

Symbol

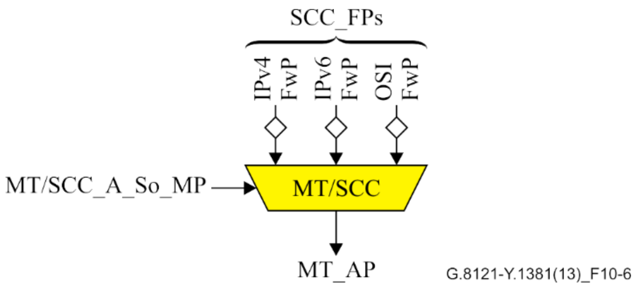


Figure 10-6 – MT/SCC_A_So function

Interfaces

The MT/SCC_A_So interfaces are described in Table 10-3.

Table 10-3 – MT/SCC_A_So inputs and outputs

Input(s)	Output(s)
SCC_FP: SCC_CI_D **MT/SCC_A_So_MP**: MT/SCC_A_So_MI_ECC_CoS MT/SCC_A_So_MI_GAL_Enable	**MT_AP**: MT_AI_D MT_AI_PHB

Processes

Activation

– The MT/SCC_A_So function shall access the access point when it is activated (MI_Active is true). Otherwise, it shall not access the access point.

The process associated with the MT/SCC_A_So function is as depicted in Figure 10-7.

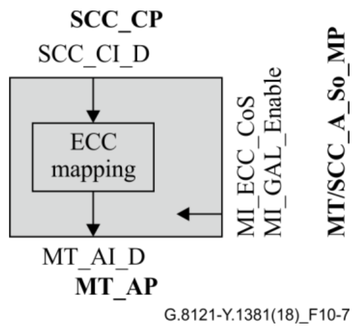


Figure 10-7 – MT/SCC_A_So processes

ECC mapping process: See clause 8.7.1.1

Defects: None.

Consequent actions: None.

Defect correlations: None.

Performance monitoring: None.

10.2.1.2 MT to SCC adaptation sink function (MT/SCC_A_Sk function)

The MT/SCC_A_Sk function extracts the SCN from the G-ACh SCC packets as defined in [IETF RFC 5718].

The information flow and processing of the MT/SCC_A_Sk functions is defined with reference to Figures 10-8 and 10-9.

Symbol

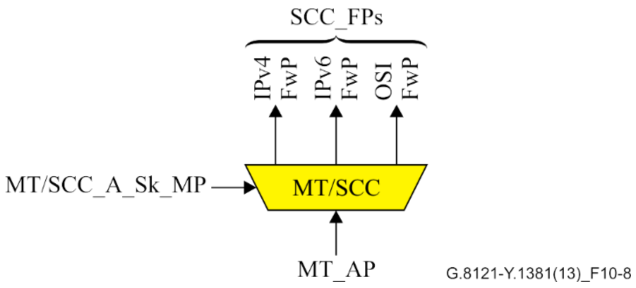


Figure 10-8 – MT/SCC_A_Sk function

Interfaces

The MT/SCC_A_Sk interfaces are described in Table 10-4.

Table 10-4 – MT/SCC_A_Sk inputs and outputs

Input(s)	Output(s)
MT_AP: MT_AI_D MT_AI_PHB MT_AI_TSF **MT/SCC_A_Sk_MP**: MT/SCC_A_Sk_MI_GAL_Enable	**SCC_FP**: SCC_CI_D SCC_CI_SSF

Processes

The processes associated with the MT/SCC_A_Sk function are as depicted in Figure 10-9.

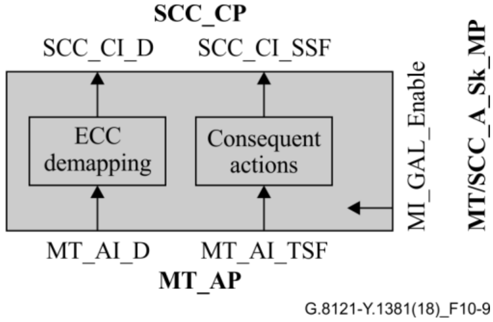


Figure 10-9 – MT/SCC_A_Sk processes

ECC demapping process: See clause 8.7.1.2

Defects: None.

Consequent actions: The function shall perform the following consequent actions:

aSSF ← AI_TSF

Defect correlations: None.

Performance monitoring: None.

10.2.2 MT/MCC_A Adaptation Function

The MT to MCC adaptation function provides access to the MCC for signalling communication. It is used for the scenarios where the MCN utilizes the MCC as defined in [IETF RFC 5718].

10.2.2.1 MT to MCC adaptation source function (MT/MCC_A_So function)

The MT/MCC_A_So function maps the MCN data into the G-ACh MCC packets as defined in [IETF RFC 5718]. The diamonds in Figure 10-10 represent traffic shaping and conditioning functions that may be needed to prevent the MCC forwarding points from exceeding their committed bandwidth in congestion situations. These traffic shaping and conditioning functions as well as the related bandwidth management and bandwidth assignment functions are outside the scope of this Recommendation.

The information flow and processing of the MT/MCC_A_So functions is defined with reference to Figures 10-10 and 10-11.

Symbol

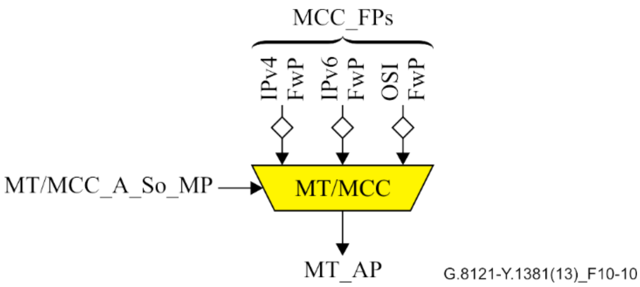


Figure 10-10 – MT/MCC_A_So function

Interfaces

The MT/MCC_A_So interfaces are described in Table 10-5.

Table 10-5 – MT/MCC_A_So inputs and outputs

Input(s)	Output(s)
MCC_FP: MCC_CI_D **MT/MCC_A_So_MP**: MT/MCC_A_So_MI_ECC_CoS MT/MCC_A_So_MI_GAL_enable	**MT_AP**: MT_AI_D MT_AI_PHB

Processes

The process associated with the MT/MCC_A_So function is as depicted in Figure 10-11.

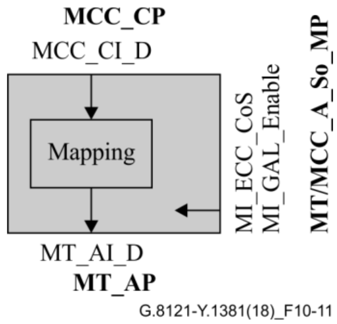


Figure 10-11 – MT/MCC_A_So processes

MCC mapping process: See clause 8.7.1.1

Defects: None.

Consequent actions: None.

Defect correlations: None.

Performance monitoring: None.

10.2.2.2 MT to MCC adaptation sink function (MT/MCC_A_Sk function)

The MT/MCC_A_Sk function extracts the MCN data from the G-ACh MCC packets as defined in [IETF RFC 5718].

The information flow and processing of the MT/MCC_A_Sk functions is defined with reference to Figures 10-12 and 10-13.

Symbol

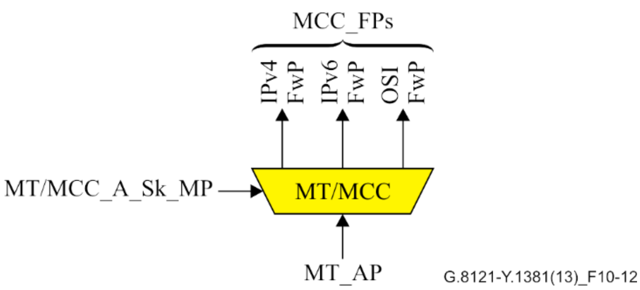


Figure 10-12 – MT/MCC_A_Sk function

Interfaces

The MT/MCC_A_Sk interfaces are described in Table 10-6.

Table 10-6 – MT/MCC_A_Sk inputs and outputs

Input(s)	Output(s)
MT_AP: MT_AI_D MT_AI_PHB MT_AI_TSF **MT/MCC_A_Sk_MP**: MT/SCC_A_Sk_MI_GAL_Enable	**MCC_FP**: MCC_CI_D MCC_CI_SSF

Processes

The processes associated with the MT/MCC_A_Sk function are as depicted in Figure 10-13.

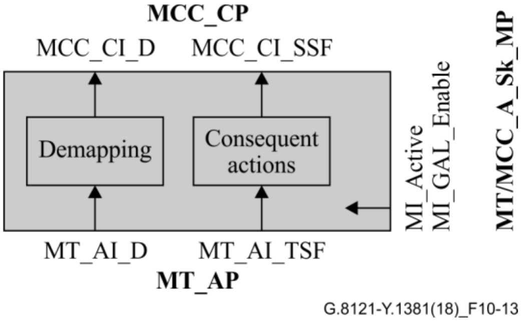


Figure 10-13 – MT/MCC_A_Sk processes

ECC demapping process: See clause 8.7.1.2

Defects: None.

Consequent actions: The function shall perform the following consequent actions:

aSSF ← AI_TSF

Defect correlations: None.

Performance monitoring: None.

11 Non-MPLS-TP server to MPLS-TP adaptation functions

11.1 SDH to MPLS-TP adaptation function (S/MT_A)

11.1.1 VC-n to MPLS-TP adaptation functions (S*n*/MT_A; *n*=3, 3-X, 4, 4-X)

11.1.1.1 VC-n to MPLS-TP adaptation source function (S*n*/MT_A_So)

This function maps MT_CI information onto a Sn_AI signal (n=3, 3-X, 4, 4-X).

Data at the Sn_AP is a VC-n (n = 3, 3-X, 4, 4-X), having a payload as described in [ITU-T G.707], but with indeterminate path overhead (POH) bytes: J1, B3, G1.

Symbol

The Sn/MT_A_So function symbol is shown in Figure 11-1.

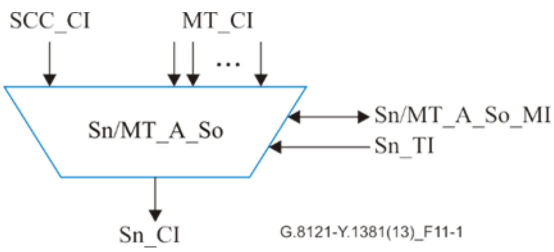


Figure 11-1 – Sn/MT_A_So symbol

Interfaces

The Sn/MT_A_So interfaces are described in Table 11-1.

Table 11-1 – Sn/MT_A_So interfaces

Input(s)	Output(s)
Each MT_CP: MT_CI_Data MT_CI_iPHB MT_CI_oPHB **SCC_CP:** SCC_CI_Data **Sn_TP:** Sn_TI_Clock Sn_TI_FrameStart **Sn/MT_A_So_MP:** Sn/MT_A_So_MI_SCCType Sn/MT_A_So_MI_Label[1…M] Sn/MT_A_So_MI_LSPType[1…M] Sn/MT_A_So_MI_CoS[1…M] Sn/MT_A_So_PHB2TCMapping[1…M] Sn/MT_A_So_MI_QoSEncodingMode[1…M] Sn/MT_A_So_MI_Mode[1…M] Sn/MT_A_So_MI _GAL_Enable[1…M]	**Sn_AP:** Sn_AI_Data Sn_AI_Clock Sn_AI_FrameStart

Processes

A process diagram of this function is shown in Figure 11-2.

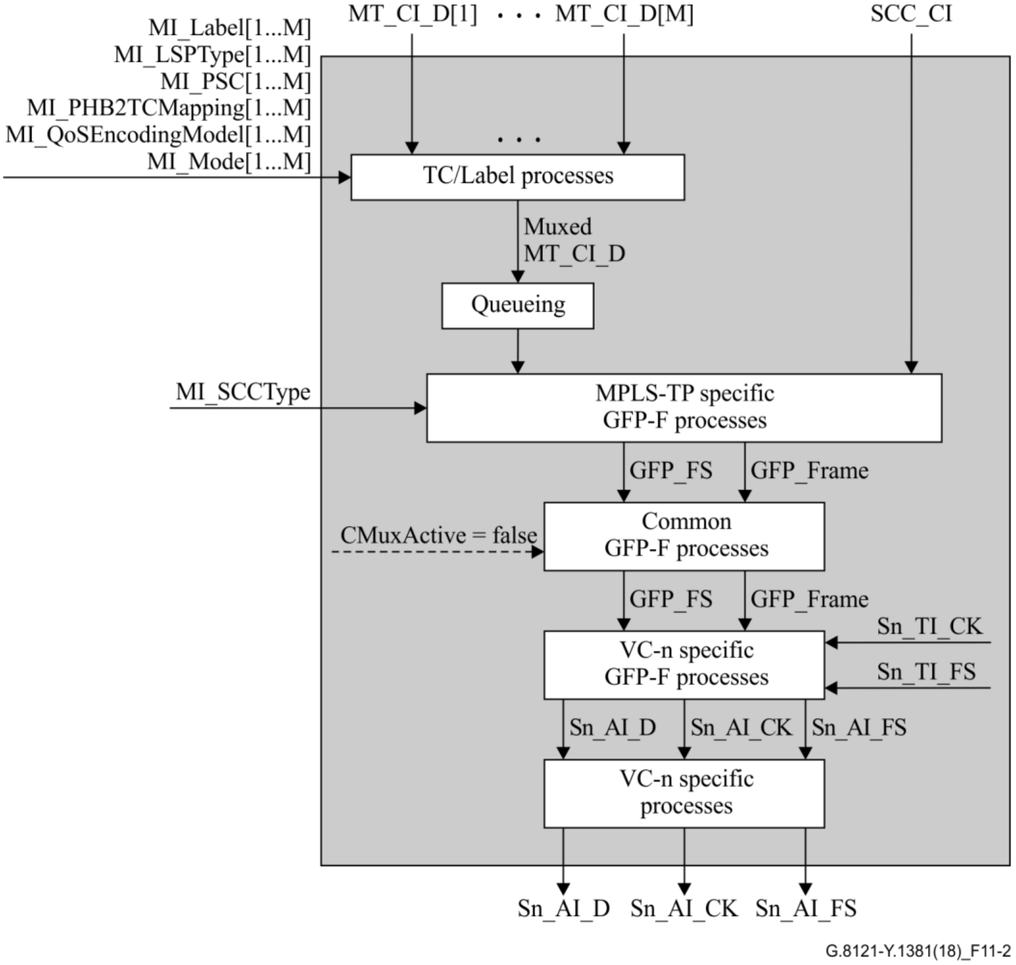


Figure 11-2 – Sn/MT_A_So process diagram

– *TC/Label processes:*

See clause 8.2.1.

– *Queuing process:*

See clause 8.3.

– *MPLS-TP-specific GFP-F source process:*

See clause 8.4.1.

– *Common GFP source process:*

See clause 8.5.3.1 of [ITU-T G.806]. GFP channel multiplexing is not supported (CMuxActive=false).

– *VC-n specific GFP source process:*

See clause 8.5.2.1 of [ITU-T G.806]. The GFP frames are mapped into the VC-n payload area according to clause 10.6 of [ITU-T G.707].

– *VC-n specific source process:*

C2: Signal label information is derived directly from the Adaptation function type. The value for "GFP mapping" in Table 9-11of [ITU-T G.707] is placed in the C2 byte position.

H4: For Sn/MT_A_So with n=3, 4, the H4 byte is sourced as all-zeros.

NOTE 1 – For Sn/MT_A_So with n=3-X, 4-X, the H4 byte is undefined at the Sn-X_AP output of this function (as per clause 12 of [ITU-T G.783]).

NOTE 2 – For Sn/MT_A_So with n=3, 4, 3-X, 4-X, the K3, F2, F3 bytes are undefined at the Sn-X_AP output of this function (as per clause 12 of [ITU-T G.783]).

Defects: None.

Consequent actions: None.

Defect correlations: None.

Performance monitoring: For further study.

11.1.1.2 VC-n to MPLS-TP adaptation sink function (S*n*/MT_A_Sk)

This function extracts MT_CI information from the Sn_AI signal (n=3, 3-X, 4, 4-X), delivering MT_CI.

Data at the Sn _AP is a VC-n (n=3, 3-X, 4, 4-X) but with indeterminate POH bytes J1, B3, G1, as per [ITU-T G.707].

Symbol

The Sn/MT_A_Sk function symbol is shown in Figure 11-3.

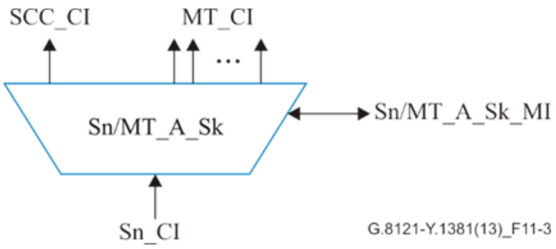


Figure 11-3 – Sn/MT_A_Sk symbol

Interfaces

The Sn/MT_A_Sk interfaces are described in Table 11-2.

Table 11-2 – Sn/MT_A_Sk interfaces

Input(s)	Output(s)
Sn_AP: Sn_AI_Data Sn_AI_ClocK Sn_AI_FrameStart Sn_AI_TSF **Sn/MT_A_Sk_MP**: Sn/MT_A_Sk_MI_SCCType Sn/MT_A_Sk_MI_Label[1…M] Sn/MT_A_Sk_MI_LSPType[1…M] Sn/MT_A_Sk_MI_CoS[1…M] Sn/MT_A_Sk_MI_TC2PHBMapping[1…M] Sn/MT_A_Sk_MI_QoSDecodingMode[1…M] Sn/MT_A_Sk_MI_Mode[1…M] Sn/MT_A_Sk_MI _LCK_Period[1…M] Sn/MT_A_Sk_MI _LCK_CoS[1…M] Sn/MT_A_Sk_MI_Admin_State Sn/MT_A_Sk_MI_AIS_Period[1…M] Sn/MT_A_Sk_MI_AIS_CoS[1…M] Sn/MT_A_Sk_MI_GAL_enable[1…M] Sn/MT_A_Sk_MI_LCK_OAM_Tool [1…M] Sn/MT_A_Sk_MI_AIS_OAM_Tool[1…M]	**Each MT_CP**: MT_CI_Data MT_CI_iPHB MT_CI_oPHB MT_CI_SSF MT_CI_LStack **SCC_CP**: SCC_CI_Data SCC_CI_SSF **Sn/MT_A_Sk_MP**: Sn/MT_A_Sk_MI_AcSL Sn/MT_A_Sk_MI_AcEXI Sn/MT_A_Sk_MI_LastInvalidUPI Sn/MT_A_Sk_MI_cPLM Sn/MT_A_Sk_MI_cLFD Sn/MT_A_Sk_MI_cEXM Sn/MT_A_Sk_MI_cUPM

Processes

A process diagram of this function is shown in Figure 11-4.

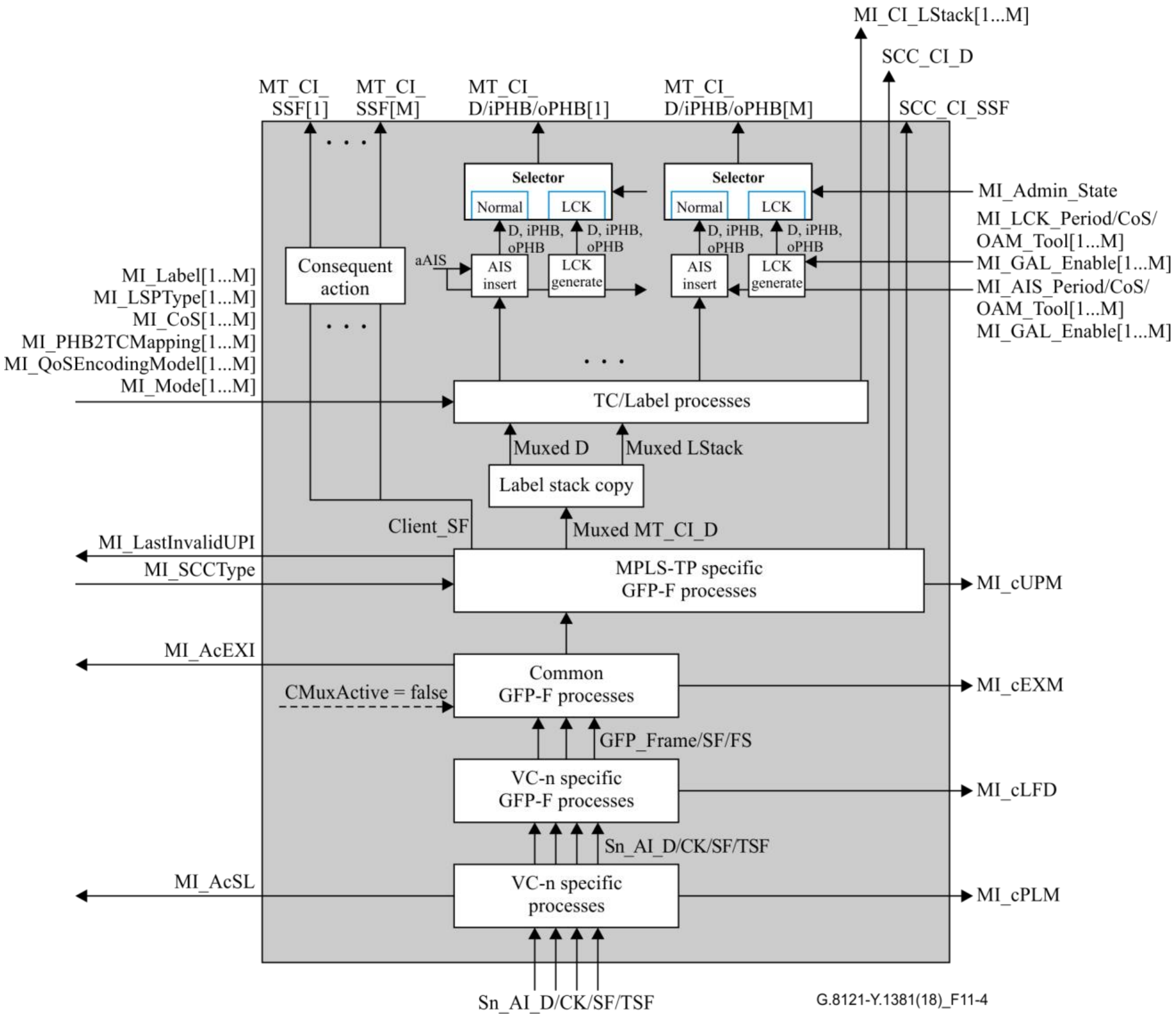


Figure 11-4 – Sn/MT_A_Sk process diagram

– *Selector generation process:*

See clause 8.6.1. The normal CI is blocked if Admin_State = LOCKED.

– *AIS insert process:*

See clause 8.6.2. There is a single AIS insert process for each MT.

– *LCK generation process:*

See clause 8.6.3. There is a single LCK generation process for each MT.

– *TC/Label sink processes:*

See clause 8.2.2.

– *Label stack copy process:*

See clause 8.2.3.

– *MPLS-TP-specific GFP-F sink process:*

See clause 8.4.2.

– *Common GFP sink process:*

See clause 8.5.3.2 of [ITU-T G.806]. GFP channel multiplexing is not supported (CMuxActive=false).

– *VC-n specific GFP sink process*

See clause 8.5.2.2 of [ITU-T G.806]. The GFP frames are demapped from the VC-n payload area according to clause 10.6 of [ITU-T G.707].

– *VC-n-specific sink process:*

C2: The signal label is recovered from the C2 byte as per clause 6.2.4.2 of [ITU-T G.806]. The signal label for "GFP mapping" in Table 9-11 of [ITU-T G.707] shall be expected. The accepted value of the signal label is also available at the Sn/MT_A_Sk_MP.

Defects:

dPLM – See clause 6.2.4.2 of [ITU-T G.806].

dLFD – See clause 6.2.5.2 of [ITU-T G.806].

dEXM – See clause 6.2.4.4 of [ITU-T G.806].

dUPM – See clause 8.4.2.

Consequent actions:

The function shall perform the following consequent actions:

aSSF ← AI_TSF or dPLM or dLFD or dUPM or dEXM

aAIS ← AI_TSF or dPLM or dLFD or dUPM or dEXM

Defect correlations:

The function shall perform the following defect correlations to determine the most probable fault cause (see clause 6.4 of [ITU-T G.806]). This fault cause shall be reported to the EMF.

cPLM ← dPLM and (not AI_TSF)

cLFD ← dLFD and (not dPLM) and (not AI_TSF)

cEXM ← dEXM and (not dPLM) and (not dLFD) and (not AI_TSF)

cUPM ← dUPM and (not dEXM) and (not dPLM) and (not dLFD) and (not AI_TSF)

Performance monitoring:

For further study.

11.1.2 LCAS-capable VC-n to MPLS-TP adaptation functions (S*n*-X-L/MT_A; *n*=3, 4)

11.1.2.1 LCAS-capable VC-n to MPLS-TP adaptation source function (S*n*-X-L/MT_A_So)

This function maps MT_CI information onto a Sn-X-L_AI signal (n=3, 4).

Data at the Sn-X-L_AP is a VC-n-X (n = 3, 4), having a payload as described in [ITU-T G.707], but with indeterminate path overhead (POH) bytes: J1, B3, G1.

Symbol

The Sn-X-L/MT_A_So function symbol is shown in Figure 11-5.

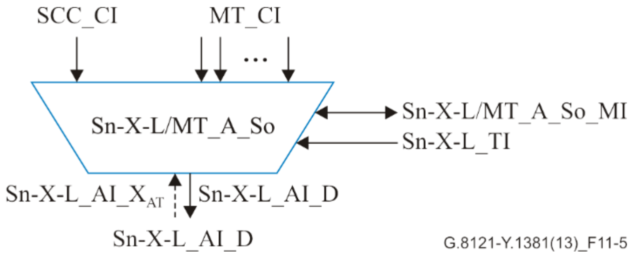


Figure 11-5 – Sn-X-L/MT_A_So symbol

Interfaces

The Sn-X-L/MT_A_So interfaces are described in Table 11-3.

Table 11-3 – Sn-X-L/MT_A_So interfaces

Input(s)	Output(s)
Each MT_CP: MT_CI_Data MT_CI_iPHB MT_CI_oPHB **SCC_CP**: SCC_CI_Data **Sn-X-L_AP**: Sn-X-L_AI_X_{AT} **Sn-X-L_TP**: Sn-X-L_TI_Clock Sn-X-L_TI_FrameStart **Sn-X-L/MT_A_So_MP**: Sn-X-L/MT_A_So_MI_SCCType Sn-X-L/MT_A_So_MI_Label[1…M] Sn-X-L/MT_A_So_MI_LSPType[1…M] Sn-X-L/MT_A_So_MI_CoS[1…M] Sn-X-L/MT_A_So_PHB2TCMapping[1…M] Sn-X-L/MT_A_So_MI_QoSEncodingMode[1…M] Sn-X-L/MT_A_So_MI_Mode[1…M] Sn-X-L/MT_A_So_MI _GAL_Enable[1…M]	**Sn-X-L_AP**: Sn-X-L_AI_Data Sn-X-L_AI_Clock Sn-X-L_AI_FrameStart

Processes

A process diagram of this function is shown in Figure 11-6.

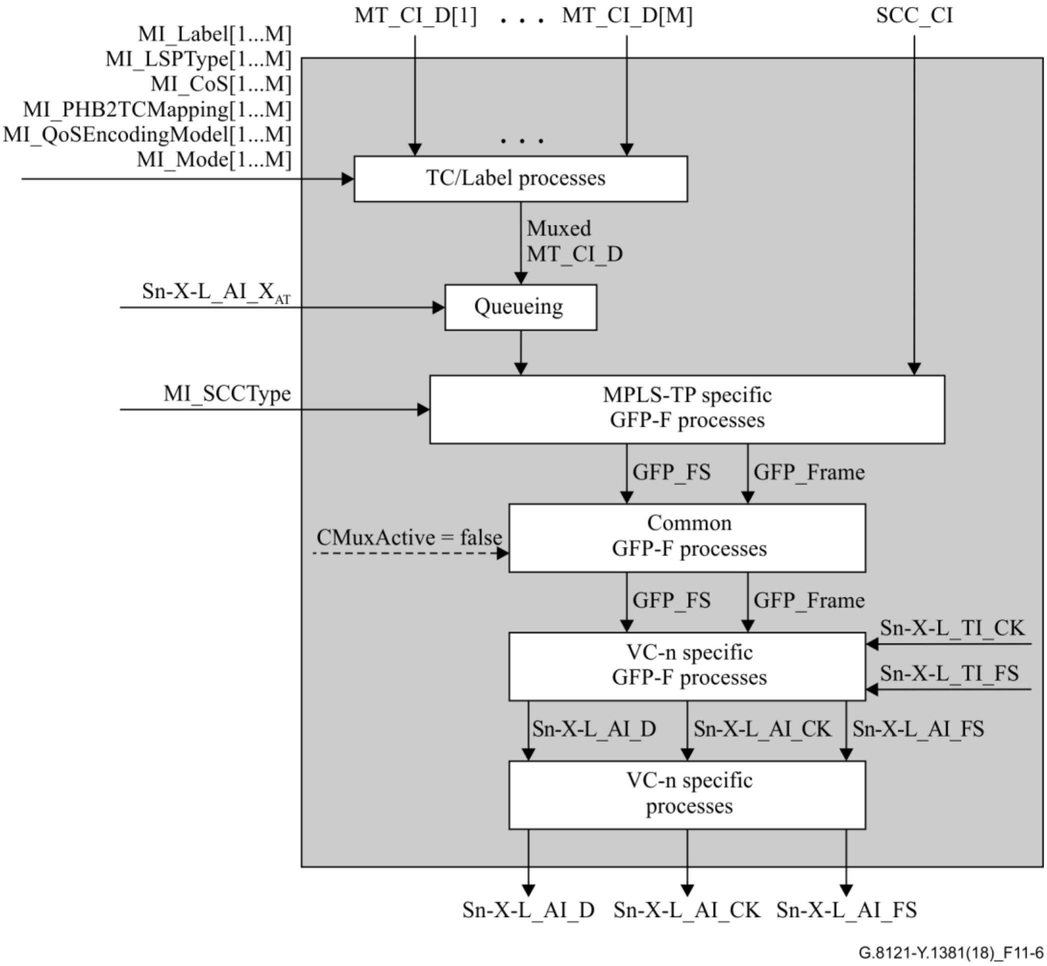


Figure 11-6 – Sn-X-L/MT_A_So process diagram

The processes have the same definition as in clause 11.1.1.1.

Defects: None.

Consequent actions: None.

Defect correlations: None.

Performance monitoring: For further study.

11.1.2.2 LCAS-capable VC-n to MPLS-TP adaptation sink function (S*n*-X-L/MT_A_Sk)

This function extracts MT_CI information from the Sn-X-L_AI signal (n=3, 4), delivering MT_CI.

Data at the Sn-X-L_AP is a VC-n-Xv (n=3, 4) but with indeterminate POH bytes J1, B3, G1, as per [ITU-T G.707].

Symbol

The Sn-X-L/MT_A_Sk function symbol is shown in Figure 11-7.

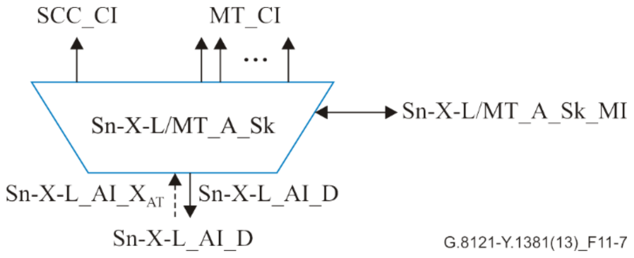


Figure 11-7 – Sn-X-L/MT_A_Sk symbol

Interfaces

The Sn-X-L/MT_A_Sk interfaces are described in Table 11-4.

Table 11-4 – Sn-X-L/MT_A_Sk interfaces

Input(s)	Output(s)
Sn-X-L_AP: Sn-X-L_AI_Data Sn-X-L_AI_ClocK Sn-X-L_AI_FrameStart Sn-X-L_AI_TSF Sn-X-L_AI_X_{AR} **Sn-X-L/MT_A_Sk_MP**: Sn-X-L/MT_A_Sk_MI_SCCType Sn-X-L/MT_A_Sk_MI_Label[1…M] Sn-X-L/MT_A_Sk_MI_LSPType[1…M] Sn-X-L/MT_A_Sk_MI_CoS[1…M] Sn-X-L/MT_A_Sk_MI_TC2PHBMapping[1…M] Sn-X-L/MT_A_Sk_MI_QoSDecodingMode[1…M] Sn-X-L/MT_A_Sk_MI_Mode[1…M] Sn-X-L/MT_A_Sk_MI_LCK_Period[1…M] Sn-X-L/MT_A_Sk_MI_LCK_CoS[1…M] Sn-X-L/MT_A_Sk_MI_Admin_State Sn-X-L/MT_A_Sk_MI_AIS_Period[1…M] Sn-X-L/MT_A_Sk_MI_AIS_CoS [1…M] Sn-X-L/MT_A_Sk_MI_GAL_Enable [1…M] Sn-X-L/MT_A_Sk_MI_LCK_OAM_Tool [1…M] Sn-X-L/MT_A_Sk_MI_AIS_OAM_Tool [1…M]	**Each MT_CP**: MT_CI_Data MT_CI_iPHB MT_CI_oPHB MT_CI_SSF MI_CI_LStack **SCC_CP**: SCC_CI_Data SCC_CI_SSF **Sn-X-L/MT_A_Sk_MP**: Sn-X-L/MT_A_Sk_MI_AcSL Sn-X-L/MT_A_Sk_MI_AcEXI Sn-X-L/MT_A_Sk_MI_LastInvalidUPI Sn-X-L/MT_A_Sk_MI_cPLM Sn-X-L/MT_A_Sk_MI_cLFD Sn-X-L/MT_A_Sk_MI_cEXM Sn-X-L/MT_A_Sk_MI_cUPM

Processes

A process diagram of this function is shown in Figure 11-8.

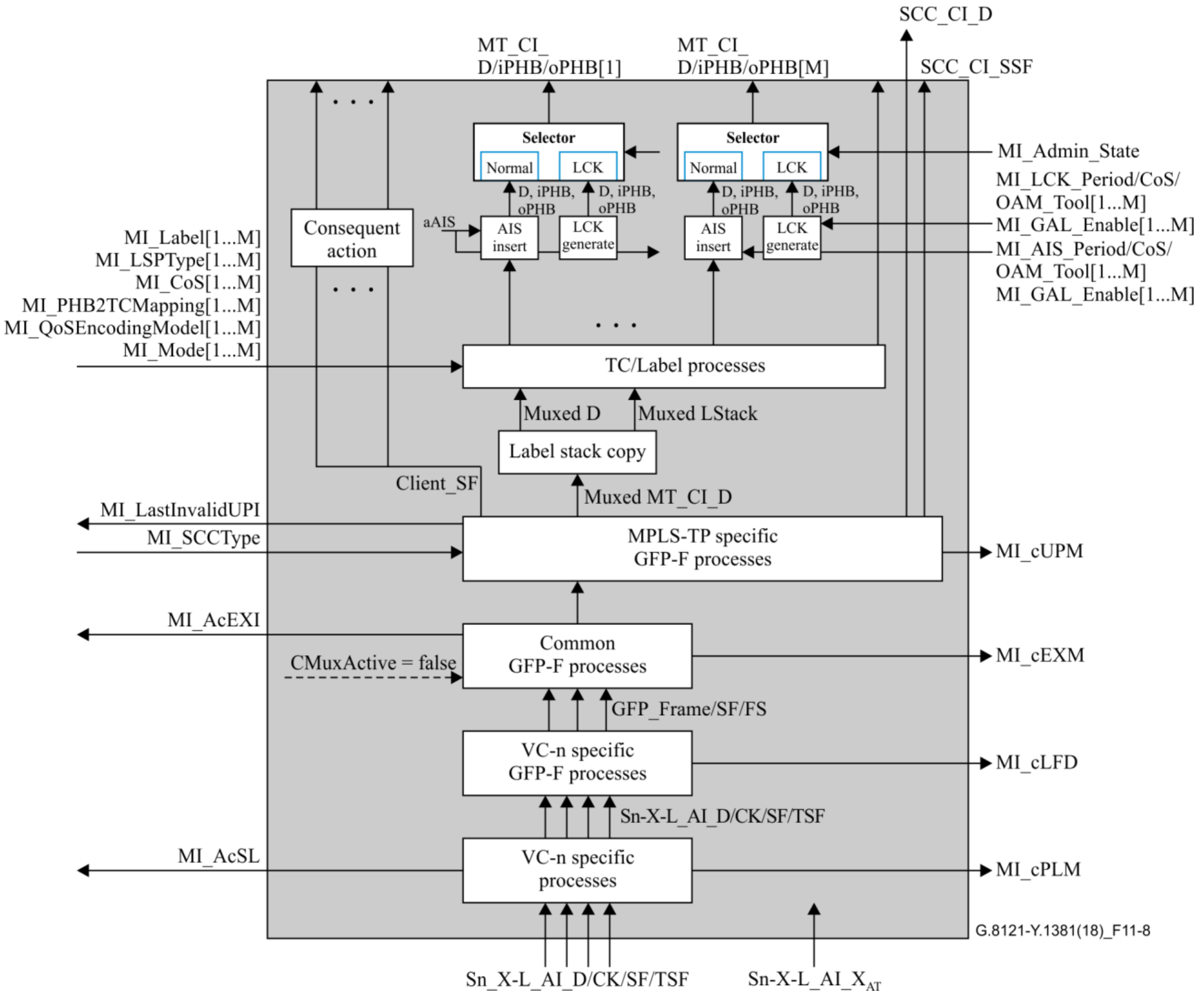


Figure 11-8 – Sn-X-L/MT_A_Sk process diagram

See process diagram and process description in clause 11.1.1.2. The additional Sn-X-L_AI_X_{AR} interface is not connected to any of the internal processes.

Defects:

dPLM – See clause 6.2.4.2 of [ITU-T G.806].

dLFD – See clause 6.2.5.2 of [ITU-T G.806].

dUPM – See clause 8.4.2.

dEXM – See clause 6.2.4.4 of [ITU-T G.806].

Consequent actions:

The function shall perform the following consequent actions:

aSSF ← AI_TSF or dPLM or dLFD or dUPM or dEXM

aAIS ← AI_TSF or dPLM or dLFD or dUPM or dEXM

Defect correlations:

The function shall perform the following defect correlations to determine the most probable fault cause (see clause 6.4 of [ITU-T G.806]). This fault cause shall be reported to the EMF.

cPLM ← dPLM and (not AI_TSF)

cLFD ← dLFD and (not dPLM) and (not AI_TSF)

cEXM ← dEXM and (not dPLM) and (not dLFD) and (not AI_TSF)

cUPM ← dUPM and (not dEXM) and (not dPLM) and (not dLFD) and (not AI_TSF)

Performance monitoring:

For further study.

11.1.3 VC-m to MPLS-TP adaptation functions (S*m*/MT_A; *m*=11, 11-X, 12, 12-X)

11.1.3.1 VC-m to MPLS-TP adaptation source function (S*m*/MT_A_So)

This function maps MT_CI information onto a Sm_AI signal (m=11, 11-X, 12, 12-X).

Data at the Sm_AP is a VC-m (m = 11, 11-X, 12, 12-X), having a payload as described in [ITU-T G.707], but with indeterminate POH bytes: J2, V5[1-4], V5[8].

Symbol

The Sm/MT_A_So function symbol is shown in Figure 11-9.

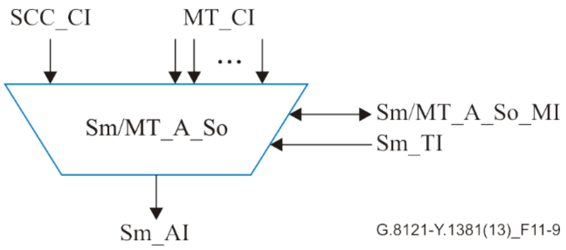


Figure 11-9 – Sm/MT_A_So symbol

Interfaces

The Sm/MT_A_So interfaces are described in Table 11-5.

Table 11-5 – Sm/MT_A_So interfaces

Input(s)	Output(s)
Each MT_CP: MT_CI_Data MT_CI_iPHB MT_CI_oPHB **SCC_CP**: SCC_CI_Data **Sm_TP**: Sm_TI_Clock Sm_TI_FrameStart	**Sm_AP**: Sm_AI_Data Sm_AI_Clock Sm_AI_FrameStart

Table 11-5 – Sm/MT_A_So interfaces

Input(s)	Output(s)
Sm/MT_A_So_MP: Sm/MT_A_So_MI_SCCType Sm/MT_A_So_MI_Label[1…M] Sm/MT_A_So_MI_LSPType[1…M] Sm/MT_A_So_MI_CoS[1…M] Sm/MT_A_So_PHB2TCMapping[1…M] Sm/MT_A_So_MI_QoSEncodingMode[1…M] Sm/MT_A_So_MI_Mode[1…M] Sm/MT_A_So_MI _GAL_Enable[1…M]	

Processes

A process diagram of this function is shown in Figure 11-10.

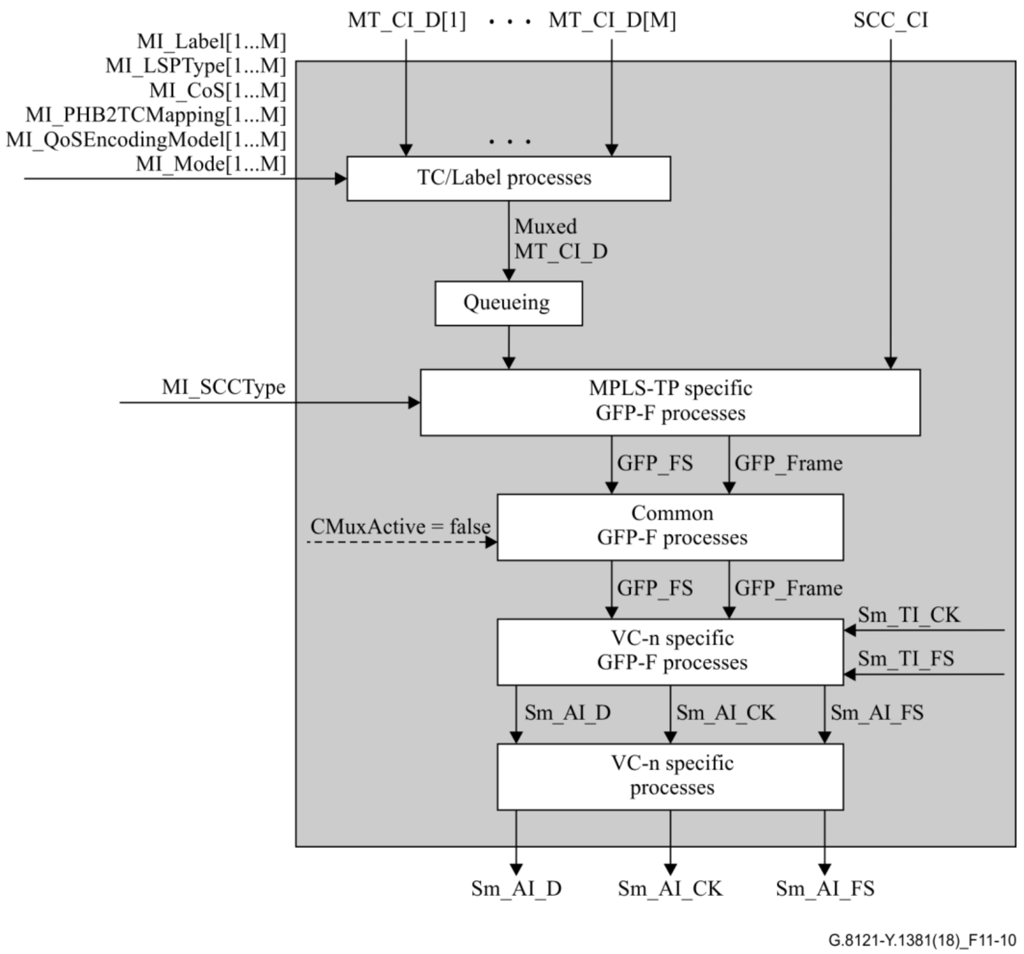


Figure 11-10 – Sm/MT_A_So process diagram

– *TC/Label processes:*

See clause 8.2.1.

– *Queuing process:*

See clause 8.3.

– *MPLS-TP-specific GFP-F source process:*

See clause 8.4.1.

– *Common GFP source process:*

See clause 8.5.3.1 of [ITU-T G.806]. GFP channel multiplexing is not supported (CMuxActive=false).

– *VC-m-specific GFP source process:*

See clause 8.5.2.1 of [ITU-T G.806]. The GFP frames are mapped into the VC-m payload area according to clause 10.6 of [ITU-T G.707].

– *VC-m-specific source process:*

V5[5-7] and K4[1]: Signal label information is derived directly from the Adaptation function type. The value for "GFP mapping" in Table 9-13 of [ITU-T G.707] is placed in the K4[1] extended signal label field as described in clause 8.2.3.2 of [ITU-T G.783].

K4[2]: For Sm/MT_A_So with m = 11, 12, the K4[2] bit is sourced as all-zeros.

NOTE 1 – For Sm/MT_A_So with m = 11-X, 12-X, the K4[2] bit is undefined at the Sm-X_AP output of this function (as per clause 13 of [ITU-T G.783]).

NOTE 2 – For Sm/MT_A_So with m = 11, 11-X, 12, 12-X, 2, the K4[3-8], V5[1-4] and V5[8] bits are undefined at the Sm-X_AP output of this function (as per clause 13 of [ITU-T G.783]).

Defects: None.

Consequent actions: None.

Defect correlations: None.

Performance monitoring: For further study.

11.1.3.2 VC-m to MPLS-TP adaptation sink function (Sm/MT_A_Sk)

This function extracts MT_CI information from the Sm_AI signal (m=11, 11-X, 12, 12-X), delivering MT_CI.

Data at the Sm _AP is a VC-m (m=11, 11-X, 12, 12-X) but with indeterminate POH bytes J2, V5[1-4], V5[8], as per [ITU-T G.707].

Symbol

The Sm/MT_A_Sk function symbol is shown in Figure 11-11.

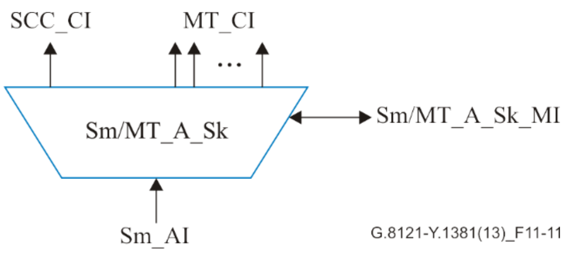


Figure 11-11 – Sm/MT_A_Sk symbol

Interfaces

The Sm/MT_A_Sk interfaces are described in Table 11-6.

Table 11-6 – Sm/MT_A_Sk interfaces

Input(s)	Output(s)
Sm_AP: Sm_AI_Data Sm_AI_ClocK Sm_AI_FrameStart Sm_AI_TSF **Sm/MT_A_Sk_MP**: Sm/MT_A_Sk_MI_SCCType Sm/MT_A_Sk_MI_Label[1…M] Sm/MT_A_Sk_MI_LSPType[1…M] Sm/MT_A_Sk_MI_CoS[1…M] Sm/MT_A_Sk_MI_TC2PHBMapping[1…M] Sm/MT_A_Sk_MI_QoSDecodingMode[1…M] Sm/MT_A_Sk_MI_Mode[1…M] Sm/MT_A_Sk_MI _LCK_Period[1…M] Sm/MT_A_Sk_MI _LCK_CoS[1…M] Sm/MT_A_Sk_MI _Admin_State Sm/MT_A_Sk_MI _AIS_Period[1…M] Sm/MT_A_Sk_MI _AIS_CoS[1…M] Sm/MT_A_Sk_MI _GAL_Enable[1…M] Sm/MT_A_Sk_MI _LCK_OAM_Tool [1…M] Sm/MT_A_Sk_MI _AIS_OAM_Tool[1…M]	**Each MT_CP**: MT_CI_Data MT_CI_iPHB MT_CI_oPHB MT_CI_SSF MI_CI_LStack **SCC_CP**: SCC_CI_Data SCC_CI_SSF **Sm/MT_A_Sk_MP**: Sm/MT_A_Sk_MI_AcSL Sm/MT_A_Sk_MI_AcEXI Sm/MT_A_Sk_MI_LastInvalidUPI Sm/MT_A_Sk_MI_cPLM Sm/MT_A_Sk_MI_cLFD Sm/MT_A_Sk_MI_cEXM Sm/MT_A_Sk_MI_cUPM

Processes

A process diagram of this function is shown in Figure 11-12.

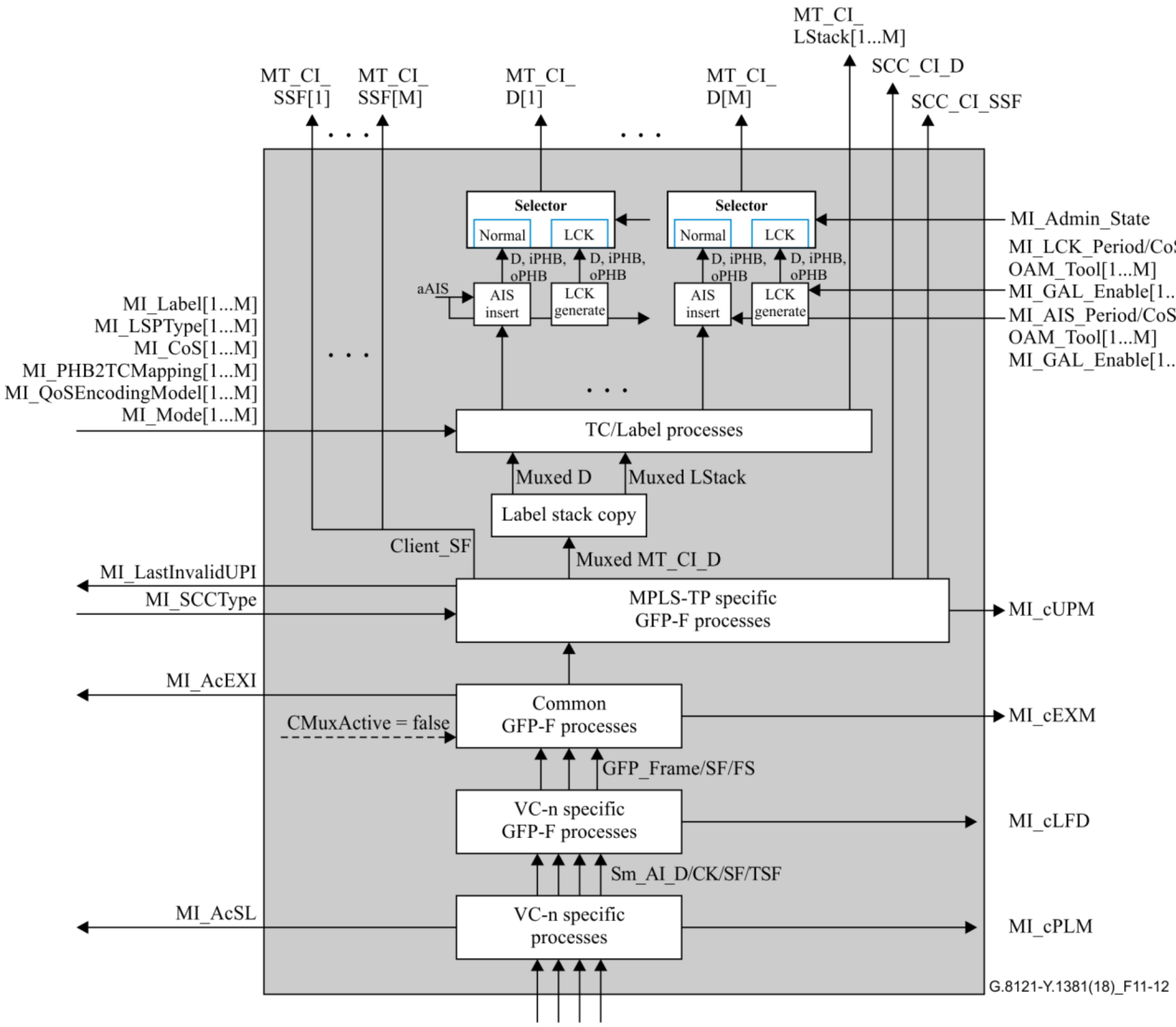


Figure 11-12 – Sm/MT_A_Sk process diagram

– *Selector generation process:*

See clause 8.6.1. The normal CI is blocked if Admin_State = LOCKED.

– *AIS insert process:*

See clause 8.6.2. There is a single AIS insert process for each MT.

– *LCK generation process:*

See clause 8.6.3. There is a single LCK generation process for each MT.

– *TC/Label sink processes:*

See clause 8.2.2.

– *Label stack copy process:*

See clause 8.2.3.

– *MPLS-TP specific GFP-F sink process:*

See clause 8.4.2.

– *Common GFP sink process:*

See clause 8.5.3.2 of [ITU-T G.806]. GFP channel multiplexing is not supported (CMuxActive=false).

– *VC-m-specific GFP sink process:*

See clause 8.5.2.2 of [ITU-T G.806]. The GFP frames are demapped from the VC-m payload area according to clause 10.6 of [ITU-T G.707].

– *VC-m-specific sink process:*

V5[5-7] and K4[1]: The signal label is recovered from the extended signal label position as described in clause 8.2.3.2 of [ITU-T G.783] and clause 6.2.4.2 of [ITU-T G.806]. The signal label for "GFP mapping" in Table 9-13 of [ITU-T G.707] shall be expected. The accepted value of the signal label is also available at the Sm/MT_A_Sk_MP.

Defects:

dPLM – See clause 6.2.4.2 of [ITU-T G.806].

dLFD – See clause 6.2.5.2 of [ITU-T G.806].

dUPM – See clause 8.4.2.

dEXM – See clause 6.2.4.4 of [ITU-T G.806].

Consequent actions:

The function shall perform the following consequent actions:

aSSF ← AI_TSF or dPLM or dLFD or dUPM or dEXM

aAIS ← AI_TSF or dPLM or dLFD or dUPM or dEXM

Defect correlations:

The function shall perform the following defect correlations to determine the most probable fault cause (see clause 6.4 of [ITU-T G.806]). This fault cause shall be reported to the EMF.

cPLM ← dPLM and (not AI_TSF)

cLFD ← dLFD and (not dPLM) and (not AI_TSF)

cEXM ← dEXM and (not dPLM) and (not dLFD) and (not AI_TSF)

cUPM ← dUPM and (not dEXM) and (not dPLM) and (not dLFD) and (not AI_TSF)

Performance monitoring:

For further study.

11.1.4 LCAS-capable VC-m to MPLS-TP adaptation functions (Sm-X-L/MT_A; *m*=11, 12)

11.1.4.1 LCAS-capable VC-m to MPLS-TP adaptation source function (Sm-X-L/MT_A_So)

This function maps MT_CI information onto a Sm-X-L_AI signal (m=11, 12).

Data at the Sm-X-L_AP is a VC-m-X (m = 11, 12), having a payload as described in [ITU-T G.707], but with indeterminate POH bytes: J2, V5[1-4], V5[8].

Symbol

The Sm-X-L/MT_A_So function symbol is shown in Figure 11-13.

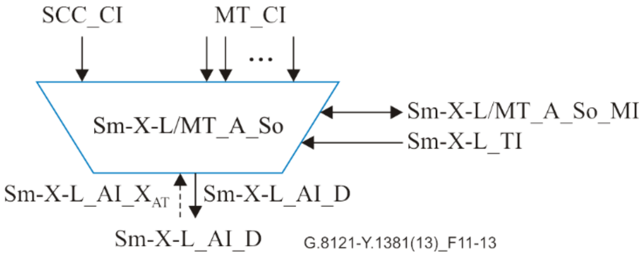


Figure 11-13 – Sm-X-L/MT_A_So symbol

Interfaces

The Sm-X-L/MT_A_So interfaces are described in Table 11-7.

Table 11-7 – Sm-X-L/MT_A_So interfaces

Input(s)	Output(s)
Each MT_CP: MT_CI_Data MT_CI_iPHB MT_CI_oPHB **SCC_CP**: SCC_CI_Data **Sm-X-L_AP:** Sm-X-L_AI_X_{AT} **Sm-X-L_TP**: Sm-X-L_TI_Clock Sm-X-L_TI_FrameStart **Sm-X-L/MT_A_So_MP**: Sm-X-L/MT_A_So_MI_SCCType Sm-X-L/MT_A_So_MI_Label[1…M] Sm-X-L/MT_A_So_MI_LSPType[1…M] Sm-X-L/MT_A_So_MI_CoS[1…M] Sm-X-L/MT_A_So_PHB2TCMapping[1…M] Sm-X-L/MT_A_So_MI_QoSEncodingMode[1…M] Sm-X-L/MT_A_So_MI_Mode[1…M] Sm-X-L/MT_A_So_MI_GAL_Enable[1…M]	**Sm-X-L_AP**: Sm-X-L_AI_Data Sm-X-L_AI_Clock Sm-X-L_AI_FrameStart

Processes

A process diagram of this function is shown in Figure 11-14.

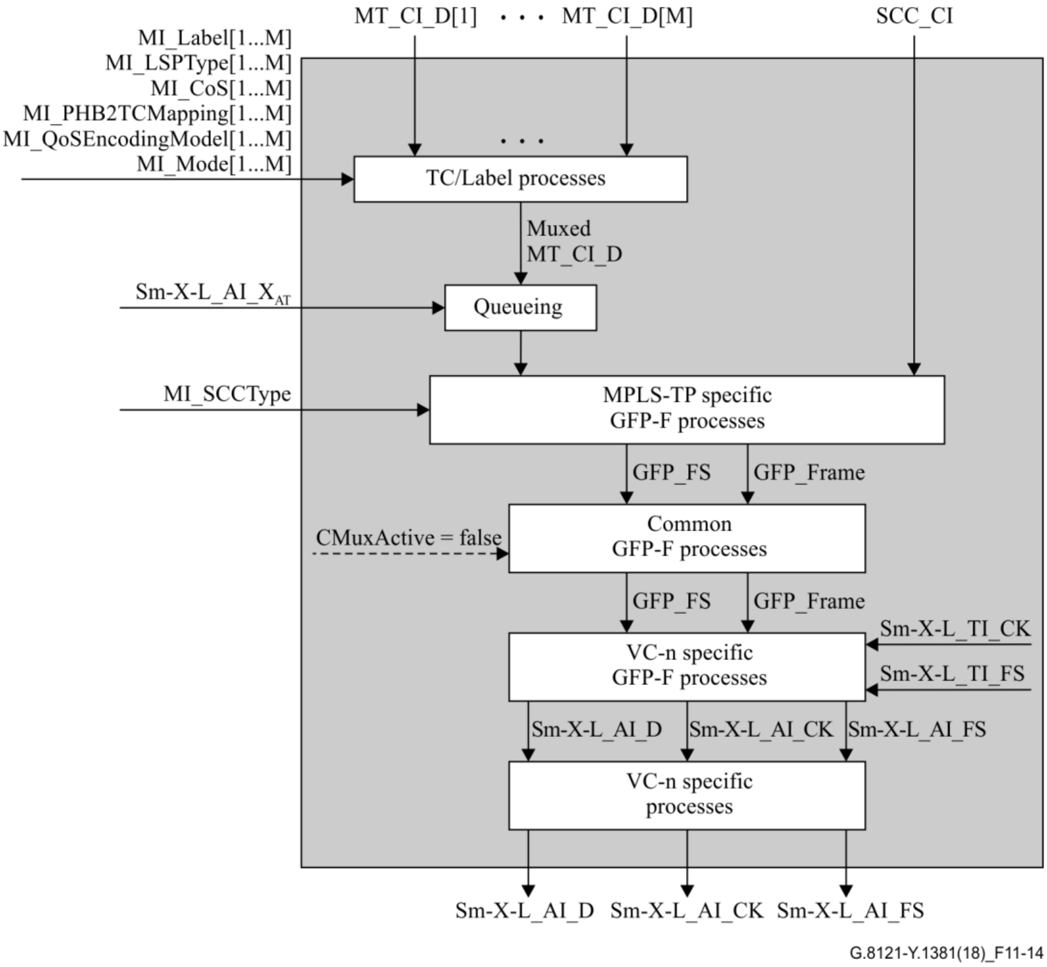


Figure 11-14 – Sm-X-L/MT_A_So process diagram

The processes have the same definition as in clause 11.1.1.1.

Defects: None.

Consequent actions: None.

Defect correlations: None.

Performance monitoring: *For further study*.

11.1.4.2 LCAS-capable VC-m to MPLS-TP adaptation sink function (Sm-X-L/MT_A_Sk)

This function extracts MT_CI information from the Sm-X-L_AI signal (m=11, 12), delivering MT_CI.

Data at the Sm-X-L_AP is a VC-m-Xv (m=11, 12) but with indeterminate POH bytes J2, V5[1-4], V5[8], as per [ITU-T G.707].

Symbol

The Sm-X-L/MT_A_Sk function symbol is shown in Figure 11-15.

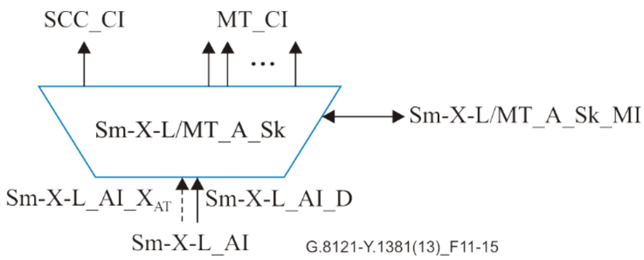


Figure 11-15 – Sm-X-L/MT_A_Sk symbol

Interfaces

The Sm-X-L/MT_A_Sk interfaces are described in Table 11-8.

Table 11-8 – Sm-X-L/MT_A_Sk interfaces

Input(s)	Output(s)
Sm-X-L_AP: Sm-X-L_AI_Data Sm-X-L_AI_ClocK Sm-X-L_AI_FrameStart Sm-X-L_AI_TSF Sm-X-L_AI_X_{AR} **Sm-X-L/MT_A_Sk_MP**: Sm-X-L/MT_A_Sk_MI_SCCType Sm-X-L/MT_A_Sk_MI_Label[1…M] Sm-X-L/MT_A_Sk_MI_LSPType[1…M] Sm-X-L/MT_A_Sk_MI_CoS[1…M] Sm-X-L/MT_A_Sk_MI_TC2PHBMapping[1…M] Sm-X-L/MT_A_Sk_MI_QoSDecodingMode[1…M] Sm-X-L/MT_A_Sk_MI_Mode[1…M] Sm-X-L/MT_A_Sk_MI_LCK_Period[1…M] Sm-X-L/MT_A_Sk_MI_LCK_CoS[1…M] Sm-X-L/MT_A_Sk_MI_Admin_State Sm-X-L/MT_A_Sk_MI_AIS_Period[1…M] Sm-X-L/MT_A_Sk_MI_AIS_CoS[1…M] Sm-X-L/MT_A_Sk_MI_GAL_Enable[1…M] Sm-X-L /MT_A_Sk_MI_LCK_OAM_Tool [1…M] Sm-X-L /MT_A_Sk_MI_AIS_OAM_Tool [1…M]	**Each MT_CP**: MT_CI_Data MT_CI_iPHB MT_CI_oPHB MT_CI_SSF MI_CI_LStack **SCC_CP**: SCC_CI_Data SCC_CI_SSF **Sm-X-L/MT_A_Sk_MP**: Sm-X-L/MT_A_Sk_MI_AcSL Sm-X-L/MT_A_Sk_MI_AcEXI Sm-X-L/MT_A_Sk_MI_LastInvalidUPI Sm-X-L/MT_A_Sk_MI_cPLM Sm-X-L/MT_A_Sk_MI_cLFD Sm-X-L/MT_A_Sk_MI_cEXM Sm-X-L/MT_A_Sk_MI_cUPM

Processes

A process diagram of this function is shown in Figure 11-16.

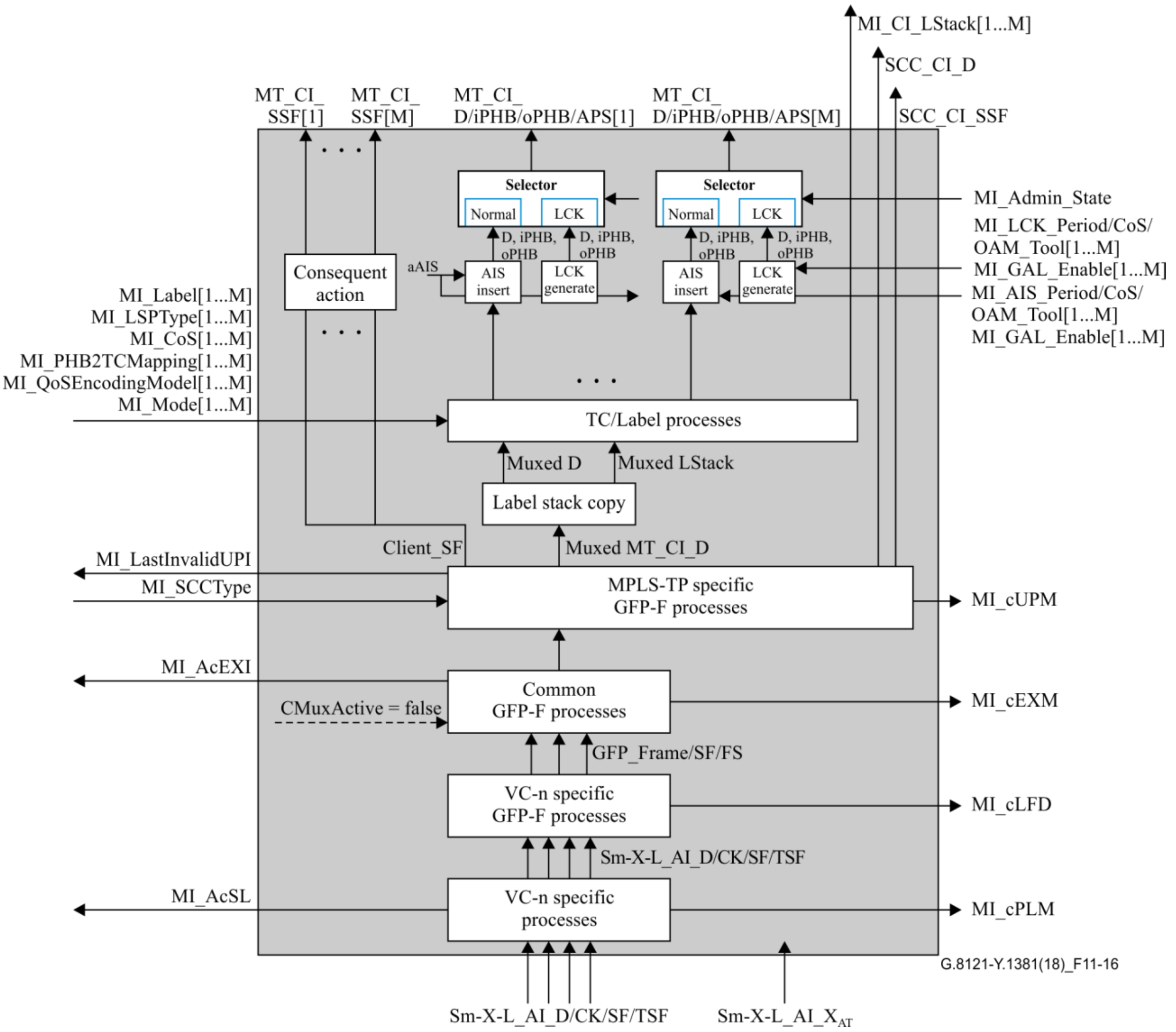


Figure 11-16 – Sm-X-L/MT_A_Sk process diagram

See process diagram and process description in clause 11.1.1.2. The additional Sm-X-L_AI_X$_{AR}$ interface is not connected to any of the internal processes.

Defects:

dPLM – See clause 6.2.4.2 of [ITU-T G.806].

dLFD – See clause 6.2.5.2 of [ITU-T G.806].

dUPM – See clause 8.4.2.

dEXM – See clause 6.2.4.4 of [ITU-T G.806].

Consequent actions:

The function shall perform the following consequent actions:

aSSF ← AI_TSF or dPLM or dLFD or dUPM or dEXM

aAIS ← AI_or dPLM or dLFD or dUPM or dEXM

Defect correlations:

The function shall perform the following defect correlations to determine the most probable fault cause (see clause 6.4 of [ITU-T G.806]). This fault cause shall be reported to the EMF.

cPLM ← dPLM and (not AI_TSF)

cLFD ← dLFD and (not dPLM) and (not AI_TSF)

cEXM ← dEXM and (not dPLM) and (not dLFD) and (not AI_TSF)

cUPM ← dUPM and (not dEXM) and (not dPLM) and (not dLFD) and (not AI_TSF)

Performance monitoring:

For further study.

11.2 OTH to MPLS-TP adaptation function (O/MT_A)

11.2.1 ODU*k* to MPLS-TP adaptation functions

11.2.1.1 ODUk to MPLS-TP adaptation source function (ODUkP/MT_A_So)

The ODUkP/MT_A_So function creates the ODUk signal from a free running clock. It maps the MT_CI information into the payload of the OPUk, adds OPUk overhead (RES, PT) and default ODUk overhead.

Symbol

The ODUkP/MT_A_So function symbol is shown in Figure 11-17.

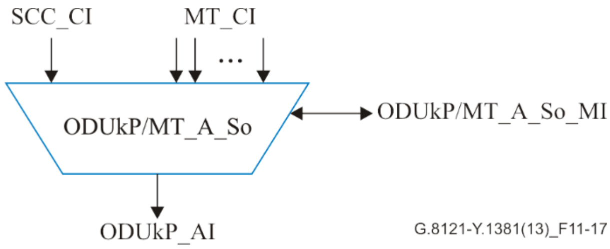


Figure 11-17 – ODUkP/MT_A_So symbol

Interfaces

The ODUkP/MT_A_So interfaces are described in Table 11-9.

Table 11-9 – ODUkP/MT_A_So interfaces

Input(s)	Output(s)
Each MT_CP: MT_CI_Data MT_CI_iPHB MT_CI_oPHB **SCC_CP**: SCC_CI_Data **ODUkP/MT_A_So_MP**: ODUkP/MT_A_So_MI_SCCType ODUkP/MT_A_So_MI_Label[1…M] ODUkP/MT_A_So_MI_LSPType[1…M]	**ODUkP_AP**: ODUkP_AI_Data ODUkP_AI_Clock ODUkP_AI_FrameStart ODUkP_AI_MultiFrameStart

Table 11-9 – ODUkP/MT_A_So interfaces

Input(s)	Output(s)
ODUkP/MT_A_So_MI_CoS[1…M] ODUkP/MT_A_So_PHB2TCMapping[1…M] ODUkP/MT_A_So_MI_QoSEncodingMode[1…M] ODUkP/MT_A_So_MI_Mode[1…M] ODUkP/MT_A_So_MI _GAL_Enable[1…M]	

Processes

A process diagram of this function is shown in Figure 11-18.

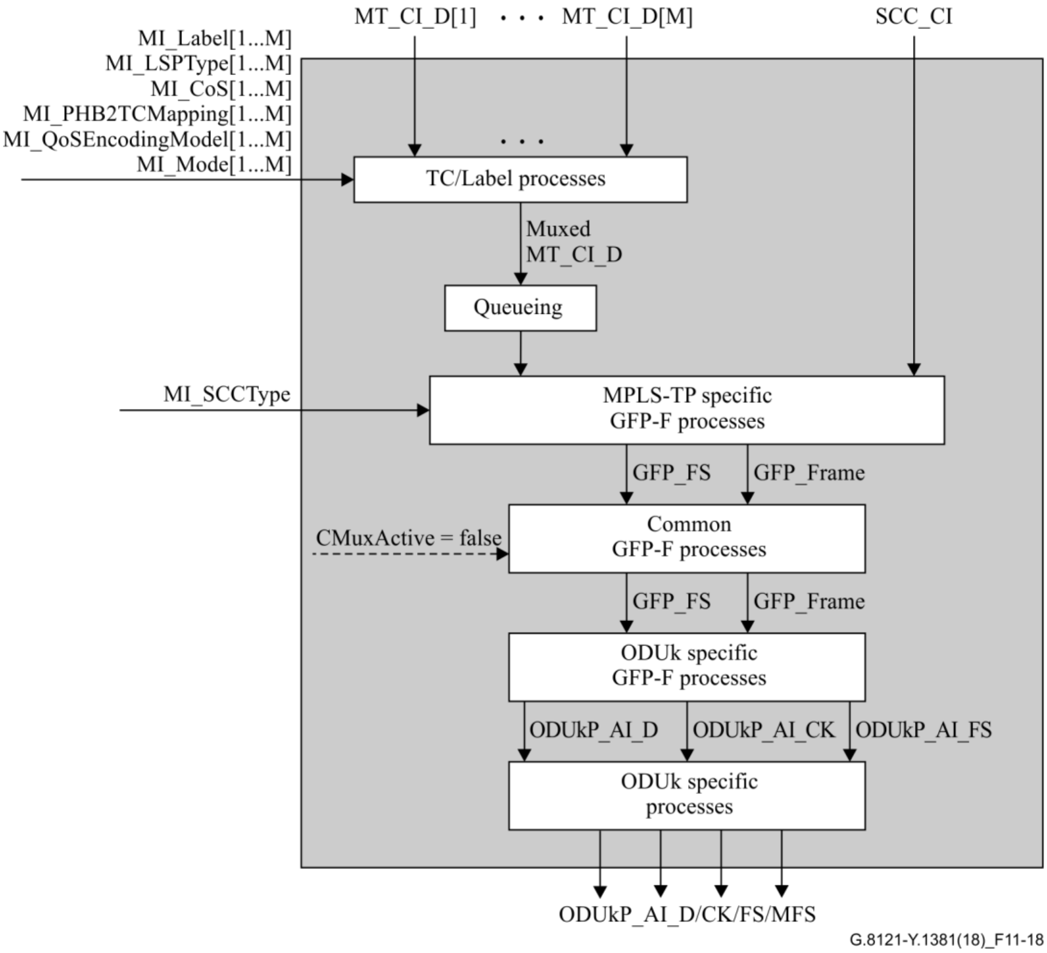


Figure 11-18 – ODUkP/MT_A_So process diagram

– *TC/Label processes:*

See clause 8.2.1.

– *Queuing process:*

See clause 8.3.

– *MPLS-TP-specific GFP-F source process:*

See clause 8.4.1.

– *Common GFP source process:*

See clause 8.5.3.1 of [ITU-T G.806]. GFP channel multiplexing is not supported (CMuxActive=false).

– *ODUk specific GFP source process:*

See clause 8.5.2.1 of [ITU-T G.806]. The GFP frames are mapped into the ODUk payload area according to clause 17.3 of [ITU-T G.709].

– *ODUk specific source process:*

See Figure 11-19.

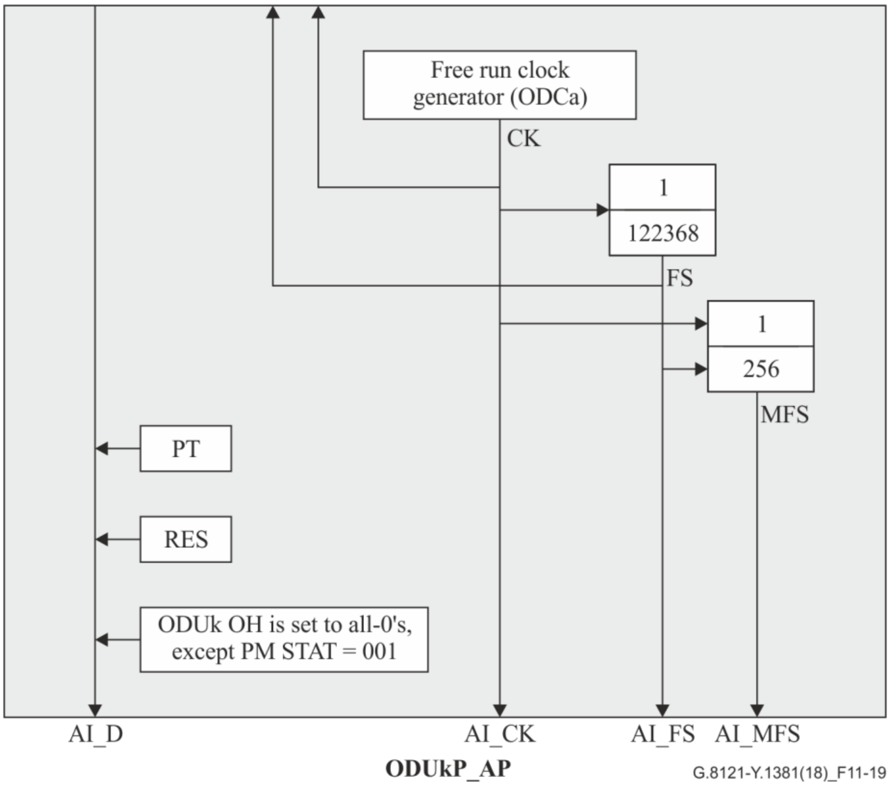


Figure 11-19 – ODUkP specific source processes

Clock and (multi)frame start signal generation: The function shall generate a local ODUk clock (ODUkP_AI_CK) of "239/(239 – k) × $4^{(k-1)}$ × 2 488 320 kHz ± 20 ppm" from a free running oscillator. The jitter and wander requirements as defined in Annex A of [ITU-T G.8251] (ODCa clock) apply.

The function shall generate the (multi)frame start reference signals AI_FS and AI_MFS for the ODUk signal. The AI_FS signal shall be active once per 122 368 clock cycles. AI_MFS shall be active once every 256 frames.

PT: The payload type information is derived directly from the Adaptation function type. The value for "GFP mapping" shall be inserted into the PT byte position of the payload structure indication (PSI) overhead as defined in clause 15.9.2.1.1 of [ITU-T G.709].

RES: The function shall insert all-0's into the RES bytes.

All other bits of the ODUk overhead should be sourced as "0"s, except the ODUk-PM STAT field which should be set to the value "normal path signal" (001).

Defects: None.

Consequent actions: None.

Defect correlations: None.

Performance monitoring: For further study.

11.2.1.2 ODUk to MPLS-TP adaptation sink function (ODUkP/MT_A_Sk)

The ODUkP/MT_A_Sk extracts MT_CI information from the ODUkP payload area. It extracts the OPUk overhead (PT and RES) and monitors the reception of the correct payload type.

Symbol

The ODUkP/MT_A_Sk function symbol is shown in Figure 11-20.

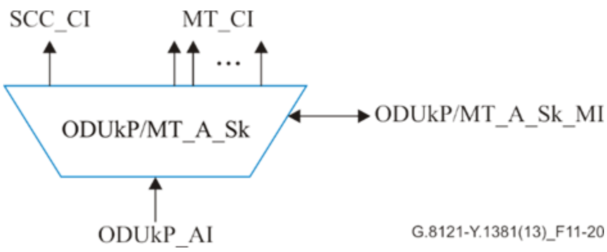


Figure 11-20 – ODUkP/MT_A_Sk symbol

Interfaces

The ODUkP/MT_A_Sk interfaces are described in Table 11-10.

Table 11-10 – ODUkP/MT_A_Sk interfaces

Input(s)	Output(s)
ODUkP_AP:	**Each MT_CP**:
ODUkP_AI_Data	MT_CI_Data
ODUkP_AI_ClocK	MT_CI_iPHB
ODUkP_AI_FrameStart	MT_CI_oPHB
ODUkP_AI_MultiFrameStart	MT_CI_SSF
ODUkP_AI_TSF	MT_CI_LStack
ODUkP/MT_A_Sk_MP:	**SCC_CP**:
ODUkP/MT_A_Sk_MI_SCCType	SCC_CI_Data
ODUkP/MT_A_Sk_MI_Label[1…M]	SCC_CI_SSF
ODUkP/MT_A_Sk_MI_LSPType[1…M]	**ODUkP/MT_A_Sk_MP**:
ODUkP/MT_A_Sk_MI_CoS[1…M]	ODUkP/MT_A_Sk_MI_AcPT
ODUkP/MT_A_Sk_MI_TC2PHBMapping[1…M]	ODUkP/MT_A_Sk_MI_AcEXI
ODUkP/MT_A_Sk_MI_QoSDecodingMode[1…M]	ODUkP/MT_A_Sk_MI_LastInvalidUPI
ODUkP/MT_A_Sk_MI_Mode[1…M]	ODUkP/MT_A_Sk_MI_cPLM
ODUkP/MT_A_Sk_MI_LCK_Period[1…M]	ODUkP/MT_A_Sk_MI_cLFD
ODUkP/MT_A_Sk_MI_LCK_CoS[1…M]	ODUkP/MT_A_Sk_MI_cEXM
ODUkP/MT_A_Sk_MI_Admin_State	ODUkP/MT_A_Sk_MI_cUPM
ODUkP/MT_A_Sk_MI_AIS_Period[1…M]	
ODUkP/MT_A_Sk_MI_AIS_CoS[1…M]	
ODUkP/MT_A_Sk_MI_GAL_Enable[1…M]	
ODUkP/MT_A_Sk_MI_LCK_Tool[1…M]	
ODUkP/MT_A_Sk_MI_AIS_Tool[1…M]	

Processes

A process diagram of this function is shown in Figure 11-21.

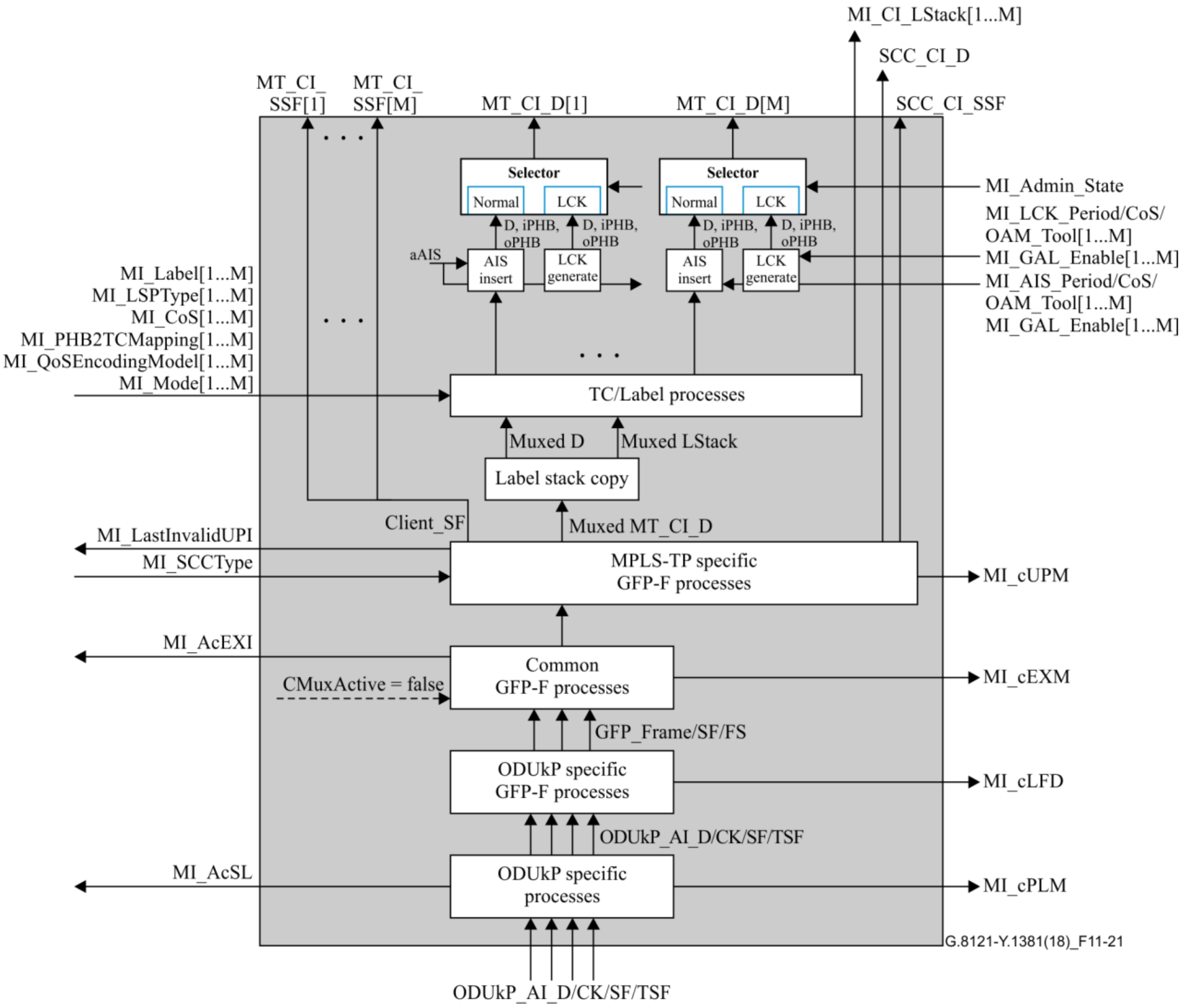


Figure 11-21 – ODUkP/MT_A_Sk process diagram

– *Selector generation process:*

See clause 8.6.1. The normal CI is blocked if Admin_State = LOCKED.

– *AIS insert process:*

See clause 8.6.2. There is a single AIS insert process for each MT.

– *LCK generation process:*

See clause 8.6.3. There is a single LCK generation process for each MT.

– *TC/Label sink processes:*

See clause 8.2.2.

– *Label stack copy process:*

See 8.2.3.

– *MPLS-TP-specific GFP-F sink process:*

See clause 8.4.2.

– *Common GFP sink process:*

See clause 8.5.3.2 of [ITU-T G.806]. GFP channel multiplexing is not supported (CMuxActive=false).

– *ODUk specific GFP sink process:*

See clause 8.5.2.2 of [ITU-T G.806]. The GFP frames are demapped from the ODUk payload area according to clause 17.3 of [ITU-T G.709].

– *ODUk-specific sink process:*

The ODUkP specific sink processes are shown in Figure 11-22.

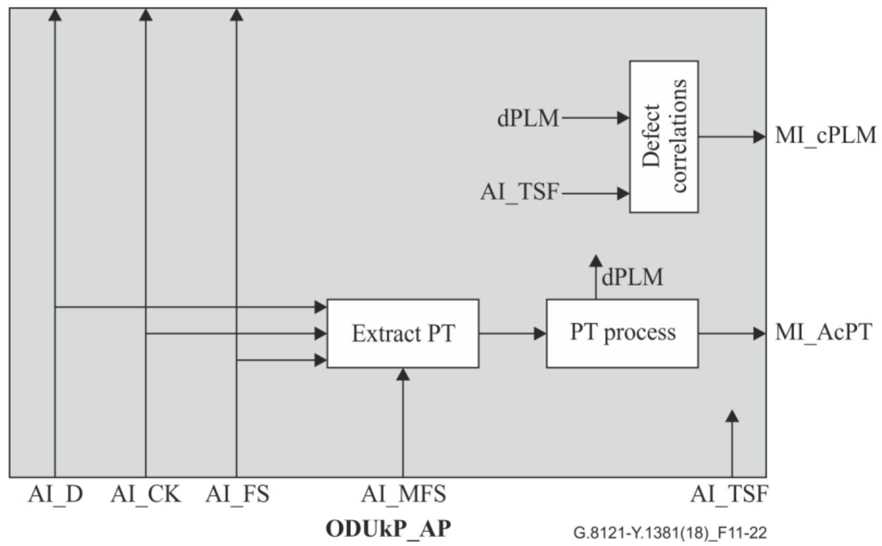


Figure 11-22 – ODUkP specific sink processes

PT: The function shall extract the PT byte from the PSI overhead as defined in clause 8.7.1 of [ITU-T G.798]. The payload type value for "GFP mapping" in clause 15.9.2.1.1 of [ITU-T G.709] shall be expected. The accepted PT value is available at the MP (MI_AcPT) and is used for payload mismatch (PLM) defect detection.

RES: The value in the RES bytes shall be ignored.

Defects:

dPLM – See clause 6.2.4.1 of [ITU-T G.798].

dLFD – See clause 6.2.5.2 of [ITU-T G.806].

dEXM – See clause 6.2.4.4 of [ITU-T G.806].

dUPM – See clause 8.4.2.

Consequent actions:

The function shall perform the following consequent actions:

aSSF ← AI_TSF or dPLM or dLFD or dUPM or dEXM

aAIS ← AI_TSF or dPLM or dLFD or dUPM or dEXM

Defect correlations:

The function shall perform the following defect correlations to determine the most probable fault cause (see clause 6.4 of [ITU-T G.806]). This fault cause shall be reported to the EMF.

cPLM ← dPLM and (not AI_TSF)

cLFD ← dLFD and (not dPLM) and (not AI_TSF)

cEXM ← dEXM and (not dPLM) and (not dLFD) and (not AI_TSF)

cUPM ← dUPM and (not dEXM) and (not dPLM) and (not dLFD) and (not AI_TSF)

Performance monitoring:

For further study.

11.2.2 HAO capable ODUk to MPLS-TP adaptation functions (ODUkP-h/MT_A; k=ODUflex)

11.2.2.1 HAO capable ODUk to MPLS-TP adaptation source function (ODUkP-h/MT_A_So)

The ODUkP-h/MT_A_So function creates the ODUk signal from a free running clock. It maps the MT_CI information into the payload of the OPUk (k=flex), adds OPUk Overhead (RES, PT, RCOH) and default ODUk overhead.

Symbol

The ODUkP-h/MT_A_So function symbol is shown in Figure 11-23.

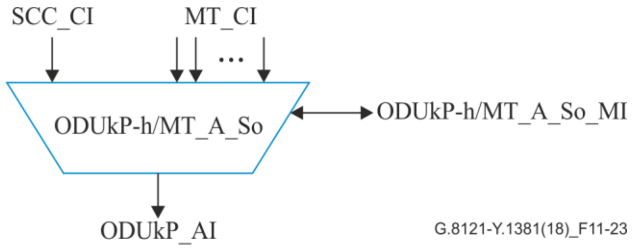


Figure 11-23 – ODUkP-h/MT_A_So symbol

Interfaces

The ODUkP-h/MT_A_So interfaces are described in Table 11-11.

Table 11-11 – ODUkP-h/MT_A_So interfaces

Input(s)	Output(s)
Each MT_CP: MT_CI_Data MT_CI_iPHB MT_CI_oPHB **ODUkP_RP**: ODUkP_RI_RP ODUkP_RI_TSCC ODUkP_RI_NCS **SCC_CP:** SCC_CI_Data **ODUkP-h/MT_A_So_MP**: ODUkP-h/MT_A_So_MI_SCCType ODUkP-h/MT_A_So_MI_Label[1…M]	**ODUkP_AP**: ODUkP_AI_Data ODUkP_AI_Clock ODUkP_AI_FrameStart ODUkP_AI_MultiFrameStart ODUkP_(A/M)I_RP ODUkP_(A/M)I_TSCC **ODUkP-h/MT_A_So_MP**: ODUkP-h/MT_A_So_MI_ADJSTATE

Table 11-11 – ODUkP-h/MT_A_So interfaces

Input(s)	Output(s)
ODUkP-h/MT_A_So_MI_LSPType[1…M] ODUkP-h/MT_A_So_MI_CoS[1…M] ODUkP-h/MT_A_So_PHB2TCMapping[1…M] ODUkP-h/MT_A_So_MI_QoSEncodingMode[1…M] ODUkP-h/MT_A_So_MI_Mode[1…M] ODUkP-h/MT_A_So_MI_GAL_Enable[1…M] ODUkP-h/MT_A_So_MI_Increase ODUkP-h/MT_A_So_MI_Decrease ODUkP-h/MT_A_So_MI_TSNUM ODUkP-h/MT_A_So_MI_ODUflexRate	
NOTE – (A/M)I_xxx indicates that the xxx signal may either be an AI_xxx or a MI_xxx signal.	

Processes

A process diagram of this function is shown as Figure 11-18 in clause 11.2.1.1. An ODU specific process for HAO capable adaptation function is described as below. Other processes are described in clause 11.2.1.1.

– *ODUk specific source process:*

Figure 11-24 illustrates ODUkP (k=flex) specific source processes.

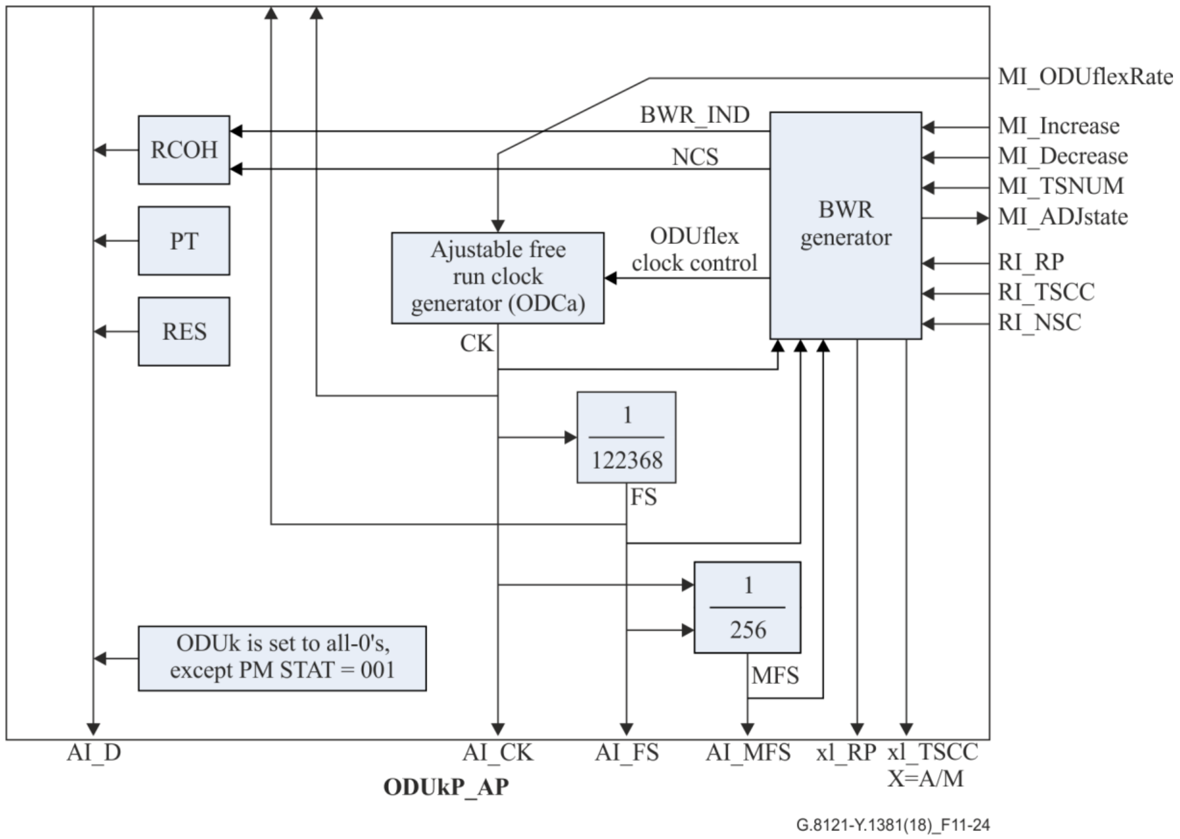


Figure 11-24 – ODUkP (k=flex) specific source processes

Clock and (multi)frame start signal generation: The function shall generate a local ODUk clock with a clock rate within the minimum to maximum clock rate of the ODUflex signal as given in Table 14-2 of [ITU-T G.798]. The jitter and wander requirements as defined in Annex A of [ITU-T G.8251] (ODCa clock) apply.

The function shall generate the (multi)frame start reference signals AI_FS and AI_MFS for the ODUk signal. The AI_FS signal shall be active once per 122 368 clock cycles. AI_MFS shall be active once every 256 frames.

PT: The payload type information is derived directly from the Adaptation function type. The value for "GFP mapping" shall be inserted into the PT byte position of the PSI overhead as defined in clause 15.9.2.1.1 of [ITU-T G.709]. The PT value of a HAO capable adaptation function is the same as a non-HAO capable one.

RES: The function shall insert all-0's into the RES bytes.

All other bits of the ODUk overhead should be sourced as "0"s, except the ODUk-PM STAT field which should be set to the value "normal path signal" (001).

RCOH generator: This process inserts network connectivity status (NCS) generated by HAO process into the NCS field of the resize control overhead (RCOH) in OPUflex.

BWR_Generator: This process is used for bandwidth resize (BWR) protocol adjustment processing and generation of BWR protocol overhead. It contains the following processes as shown in Figure 11-25.

Adjustment activation: When MI_Increase or MI_Decrease is true, BWR protocol is activated and RI processing is started.

Rate adjustment control: Generates ODUflex clock control signal. Original ODUflex clock rate will gradually change to new ODUflex clock rate such that no GMP buffer overflow or underflow will occur in the ODUflex network connection. Refer to [ITU-T G.7044].

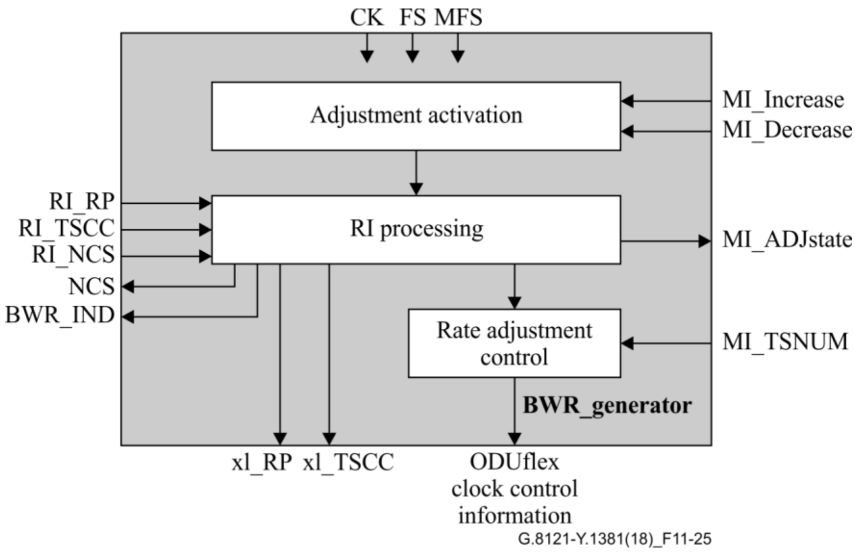


Figure 11-25 – BWR_Generator process

RI processing: This process performs BWR protocol according to RI_RP, RI_TSCC, RI_NCS signals received from the BWR_Receiver process.

– When RI processing is activated, xI_RP and xI _TSCC (x is A or M) signals are set to one (1).

– The value of the NCS signal is set to ACK(1) when receiving RI_RP=1 and the value of RI_TSCC is changed from 0 to 1.

– Rate adjustment control is activated when receiving RI_RP=1 and RI_TSCC=1 and RI_NCS=ACK(1).
– BWR_IND is set to "1" x µs before ODUflex signal's bit rate adjustment starts, and is set to "0" y µs before ODUflex signal's bit rate adjustment completes. x is almost equal to y and shall be in the range of 125 to 250 µs.
– The value of xI_TSCC signal is set to 0 when rate adjustment is completed.
– The value of NCS signal is set to NACK(0) when receiving RI_RP=1 and the value of RI_TSCC is changed from 1 to 0.
– The value of RP signal is set to 0 when receiving RI_NCS=NACK(0) and sending NCS=NACK(0).
– The completion of the resize process is reported to network management system (NMS) when receiving RI_RP=0.

Defects: None.

Consequent actions: None.

Defect correlations: None.

Performance monitoring: For further study.

11.2.2.2 HAO capable ODUk to MPLS-TP adaptation sink function (ODUkP-h/MT_A_Sk)

The ODUkP-h/MT_A_Sk extracts MT_CI information from the ODUkP payload area. It extracts the OPUk overhead (PT, RCOH and RES) and monitors the reception of the correct payload type.

Symbol

The ODUkP-h/MT_A_Sk function symbol is shown in Figure 11-26.

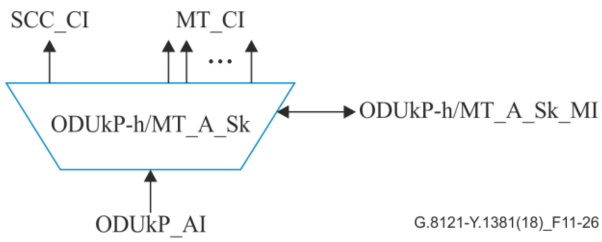


Figure 11-26 – ODUkP-h/MT_A_Sk symbol

Interfaces

The ODUkP-h/MT_A_Sk interfaces are described in Table 11-12.

Table 11-12 – ODUkP-h/MT_A_Sk interfaces

Input(s)	Output(s)
ODUkP_AP: ODUkP_AI_Data ODUkP_AI_ClocK ODUkP_AI_FrameStart ODUkP_AI_MultiFrameStart ODUkP_AI_TSF	**Each MT_CP**: MT_CI_Data MT_CI_iPHB MT_CI_oPHB MT_CI_SSF MT_CI_LStack **SCC_CP**:

Table 11-12 – ODUkP-h/MT_A_Sk interfaces

Input(s)	Output(s)
ODUkP_(A/M)I_RP ODUkP_(A/M)I_TSCC **ODUkP-h/MT_A_Sk_MP**: ODUkP-h /MT_A_Sk_MI_SCCType ODUkP-h /MT_A_Sk_MI_Label[1…M] ODUkP-h /MT_A_Sk_MI_LSPType[1…M] ODUkP-h /MT_A_Sk_MI_CoS[1…M] ODUkP-h /MT_A_Sk_MI_TC2PHBMapping[1…M] ODUkP-h /MT_A_Sk_MI_QoSDecodingMode[1…M] ODUkP/MT_A_Sk_MI_Mode[1…M] ODUkP-h /MT_A_Sk_MI_LCK_Period[1…M] ODUkP-h /MT_A_Sk_MI_LCK_CoS[1…M] ODUkP-h /MT_A_Sk_MI_LCK_Tool[1…M] ODUkP-h /MT_A_Sk_MI_AIS_Tool[1…M] ODUkP-h /MT_A_Sk_MI_Admin_State ODUkP-h /MT_A_Sk_MI_AIS_Period[1…M] ODUkP-h /MT_A_Sk_MI_AIS_CoS[1…M] ODUkP-h /MT_A_Sk_MI_LCK_Tool[1…M] ODUkP-h /MT_A_Sk_MI_AIS_Tool[1…M] ODUkP-h /MT_A_Sk_MI_GAL_Enable[1…M] ODUkP-h/MT_A_Sk_MI_Increase ODUkP-h/MT_A_Sk_MI_Decrease	SCC_CI_Data SCC_CI_SSF **ODUkP_RP**: ODUkP_RI_RP ODUkP_RI_TSCC ODUkP_RI_NCS **ODUkP-h/MT_A_Sk_MP**: ODUkP-h/MT_A_Sk_MI_AcPT ODUkP-h/MT_A_Sk_MI_AcEXI ODUkP-h/MT_A_Sk_MI_LastInvalidUPI ODUkP-h/MT_A_Sk_MI_cPLM ODUkP-h/MT_A_Sk_MI_cLFD ODUkP-h/MT_A_Sk_MI_cEXM ODUkP-h/MT_A_Sk_MI_cUPM

Processes

A process diagram of this function is shown as Figure 11-21 in clause 11.2.1.2. An ODU specific process for HAO capable adaptation function is described as below. Other processes are described in clause 11.2.1.2.

– *ODUk-specific sink process:*

The ODUkP (k=flex) specific sink process is illustrated in Figure 11-27.

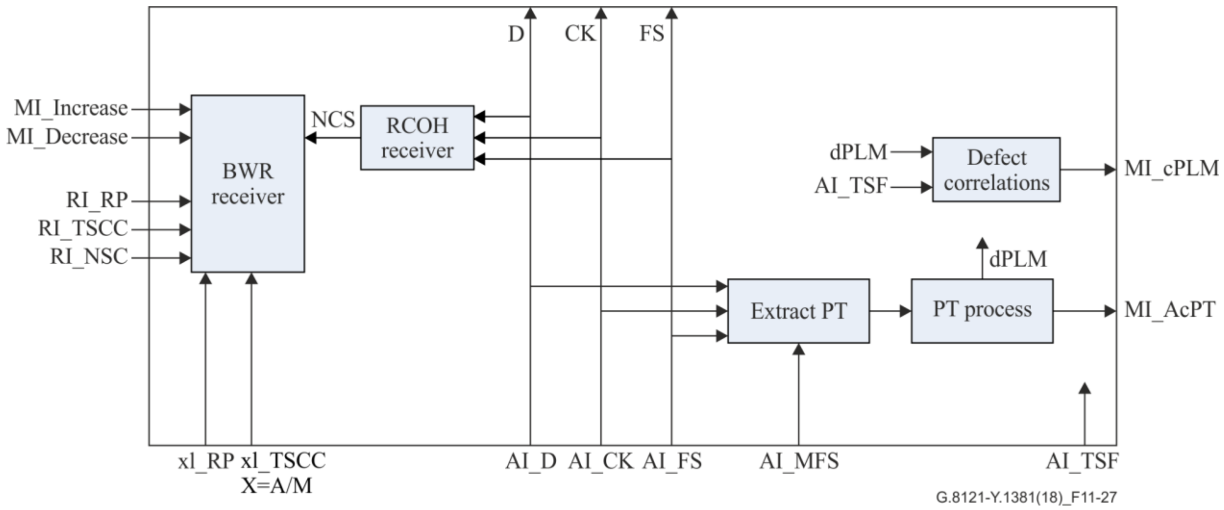


Figure 11-27 – ODUkP (k=flex) specific sink processes

PT: The function shall extract the PT byte from the PSI overhead as defined in clause 8.7.1 of [ITU-T G.798]. The payload type value for "GFP mapping" in clause 15.9.2.1.1 of [ITU-T G.709] shall be expected. The accepted PT value is available at the MP (MI_AcPT) and is used for PLM defect detection. The PT value of a HAO capable adaptation function is the same as a non-HAO capable one.

RES: The value in the RES bytes shall be ignored.

RCOH receiver: This process extracts NCS from RCOH overhead area, and then forwards it to the BWR_Receiver.

BWR_Receiver: This process extracts and detects the BWR protocol overhead, with the exception of the BWR_IND signal. It is shown in Figure 11-28.

When MI_Increase or MI_Decrease is true, the BWR protocol is activated and starts to receive AI_RP/MI_RP, AI_TSCC/MI_TSCC from BWR_RELAY_Receiver process and NCS from Extract NCS process. Then the detected value of RP, TSCC and NCS are sent to BWR generator.

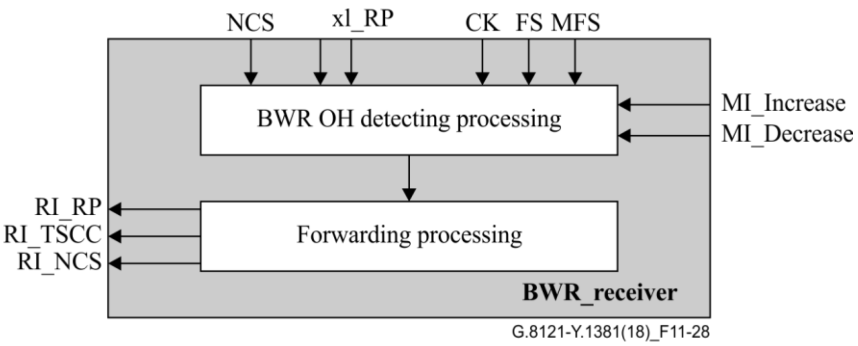


Figure 11-28 – BWR_Receiver process

Defects:

dPLM – See clause 6.2.4.1 of [ITU-T G.798].

dLFD – See clause 6.2.5.2 of [ITU-T G.806].

dEXM – See clause 6.2.4.4 of [ITU-T G.806].

dUPM – See clause 8.4.2.

Consequent actions:

The function shall perform the following consequent actions:

aSSF ← AI_TSF or dPLM or dLFD or dUPM or dEXM

aAIS ← AI_TSF or dPLM or dLFD or dUPM or dEXM

Defect correlations:

The function shall perform the following defect correlations to determine the most probable fault cause (see clause 6.4 of [ITU-T G.806]). This fault cause shall be reported to the EMF.

cPLM ← dPLM and (not AI_TSF)

cLFD ← dLFD and (not dPLM) and (not AI_TSF)

cEXM ← dEXM and (not dPLM) and (not dLFD) and (not AI_TSF)

cUPM ← dUPM and (not dEXM) and (not dPLM) and (not dLFD) and (not AI_TSF)

Performance monitoring:

For further study.

11.3 PDH to MPLS-TP adaptation function (P/MT_A)

11.3.1 *Pq* to MPLS-TP adaptation functions (P*q*/MT_A; q = 11s, 12s, 31s, 32e)

11.3.1.1 *Pq* to MPLS-TP adaptation source function (P*q*/MT_A_So)

This function maps MT_CI information onto a Pq_AI signal (q = 11s, 12s, 31s, 32e).

Data at the Pq_AP is a Pq (q = 11s, 12s, 31s, 32e), having a payload as described in [ITU-T G.7043] with a value of N=1. The VLI byte is reserved and not used for payload data.

Symbol

The Pq/MT_A_So function symbol is shown in Figure 11-29.

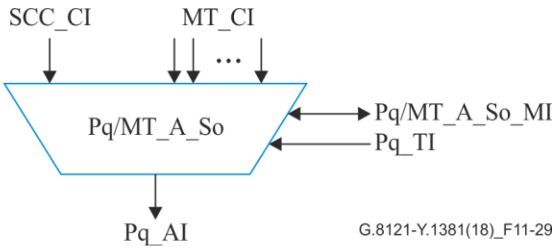


Figure 11-29 – Pq/MT_A_So symbol

Interfaces

The Pq/MT_A_So interfaces are described in Table 11-13.

Table 11-13 – Pq/MT_A_So interfaces

Input(s)	Output(s)
Each MT_CP: MT_CI_Data MT_CI_iPHB MT_CI_oPHB **SCC_CP:** SCC_CI_Data **Pq_TP:** Pq_TI_Clock Pq_TI_FrameStart **Pq/MT_A_So_MP:** Pq/MT_A_So_MI_SCCType Pq/MT_A_So_MI_Label[1…M] Pq/MT_A_So_MI_LSPType[1…M] Pq/MT_A_So_MI_CoS[1…M] Pq/MT_A_So_PHB2TCMapping[1…M] Pq/MT_A_So_MI_QoSEncodingMode[1…M] Pq/MT_A_So_MI_Mode[1…M] Pq/MT_A_So_MI_GAL_Enable[1…M]	**Pq_AP:** Pq_AI_Data Pq_AI_Clock Pq_AI_FrameStart

Processes

A process diagram of this function is shown in Figure 11-30.

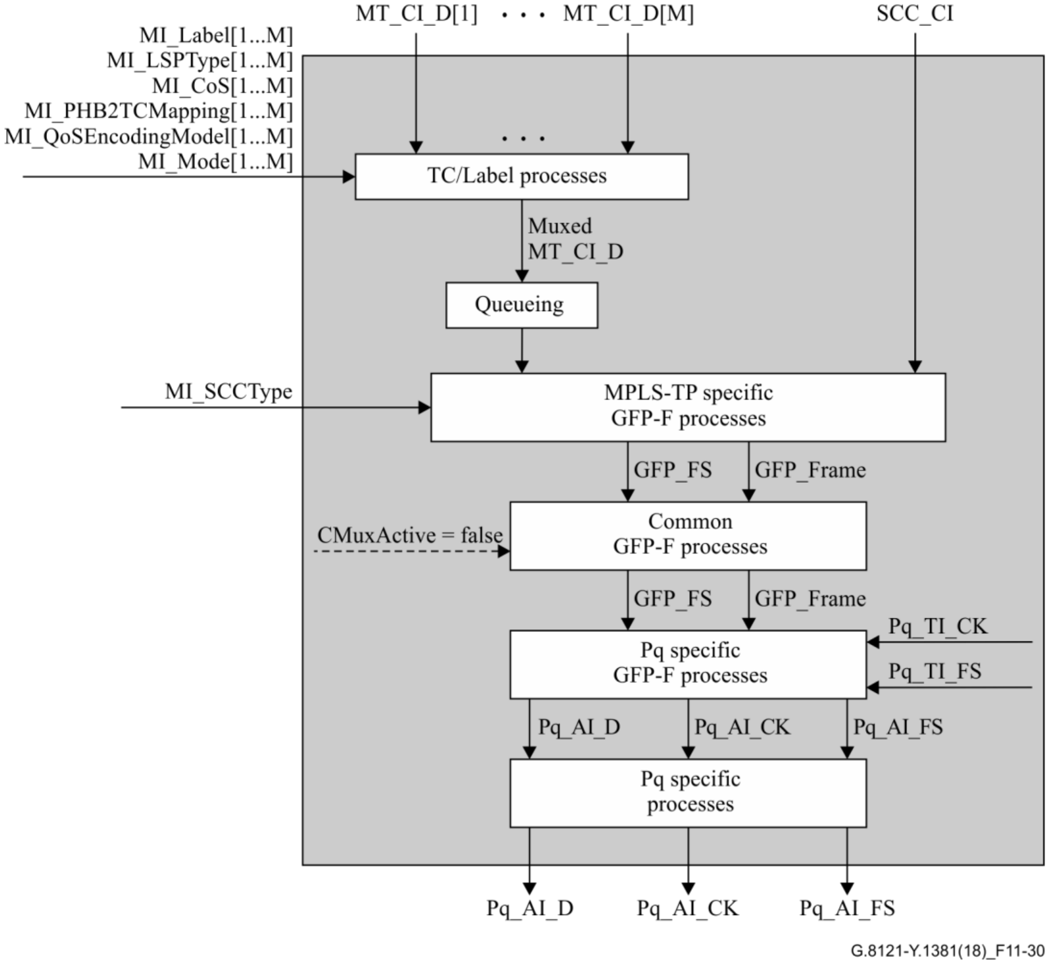


Figure 11-30 – Pq/MT_A_So process diagram

TC/Label processes:

See clause 8.2.1.

Queuing process:

See clause 8.3.

MPLS-TP-specific GFP-F source process:

See clause 8.4.1.

Common GFP source process:

See clause 8.5.3.1 of [ITU-T G.806]. GFP channel multiplexing is not supported (CMuxActive=false).

Pq specific GFP source process:

See clause 8.5.2.1 of [ITU-T G.806]. The GFP frames are mapped into the Pq payload area according to [ITU-T G.8040].

Pq specific source process:

NOTE – The VLI byte is fixed and equal to 0x00 at the Pq_AP output of this function.

P31s specific:

MA: Signal label information is derived directly from the Adaptation function type. The value for "GFP mapping" in clause 2.1 of [ITU-T G.832] is placed in the "Payload type" field of the MA byte.

Defects: None.

Consequent actions: None.

Defect correlations: None.

Performance monitoring: For further study.

11.3.1.2 Pq to MPLS-TP adaptation sink function (Pq/MT_A_Sk)

This function extracts MT_CI information from the Pq_AI signal (q = 11s, 12s, 31s, 32e), delivering MT_CI.

Data at the Pq _AP is a Pq (q = 11s, 12s, 31s, 32e), having a payload as described in [ITU-T G.7043] with a value of N=1. The VLI byte is reserved and not used for payload data.

Symbol

The Pq/MT_A_Sk function symbol is shown in Figure 11-31.

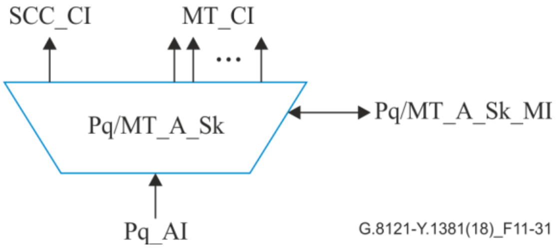


Figure 11-31 – Pq/MT_A_Sk symbol

Interfaces

The Pq/MT_A_Sk interfaces are described in Table 11-14.

Table 11-14 – Pq/MT_A_Sk interfaces

Input(s)	Output(s)
Pq_AP: Pq_AI_Data Pq_AI_ClocK Pq_AI_FrameStart Pq_AI_TSF **Pq/MT_A_Sk_MP:** Pq/MT_A_Sk_MI_SCCType Pq/MT_A_Sk_MI_Label[1…M] Pq/MT_A_Sk_MI_LSPType[1…M] Pq/MT_A_Sk_MI_CoS[1…M] Pq/MT_A_Sk_MI_TC2PHBMapping[1…M] Pq/MT_A_Sk_MI_QoSDecodingMode[1…M] Pq/MT_A_Sk_MI_Mode[1…M]	**Each MT_CP:** MT_CI_Data MT_CI_iPHB MT_CI_oPHB MT_CI_SSF MT_CI_LStack **SCC_CP:** SCC_CI_Data SCC_CI_SSF **Pq/MT_A_Sk_MP:** Pq/MT_A_Sk_MI_AcSL Pq/MT_A_Sk_MI_AcEXI Pq/MT_A_Sk_MI_LastInvalidUPI Pq/MT_A_Sk_MI_cPLM Pq/MT_A_Sk_MI_cLFD

Table 11-14 – Pq/MT_A_Sk interfaces

Input(s)	Output(s)
Pq/MT_A_Sk_MI_LCK_Period[1…M] Pq/MT_A_Sk_MI_LCK_CoS[1…M] Pq/MT_A_Sk_MI_Admin_State Pq/MT_A_Sk_MI_AIS_Period[1…M] Pq/MT_A_Sk_MI_AIS_CoS[1…M] Pq/MT_A_Sk_MI_GAL_Enable [1…M] Pq/MT_A_Sk_MI_LCK_Tool[1…M] Pq/MT_A_Sk_MI_AIS_Tool[1…M]	Pq/MT_A_Sk_MI_cEXM Pq/MT_A_Sk_MI_cUPM

Processes

A process diagram of this function is shown in Figure 11-32.

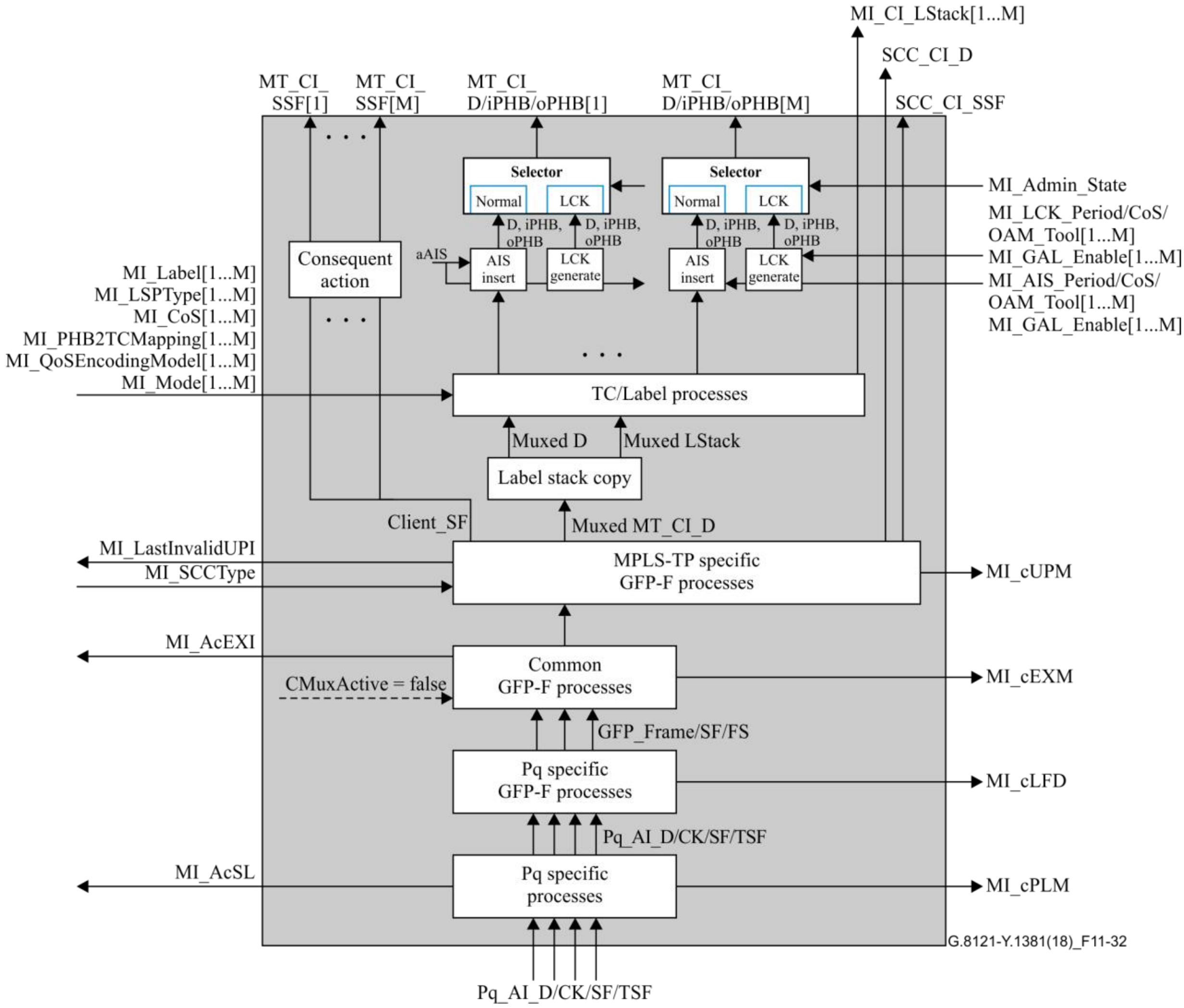


Figure 11-32 – Pq/MT_A_Sk process diagram

– *Selector generation process:*

See clause 8.6.1. The normal CI is blocked if Admin_State = LOCKED.

– *AIS insert process:*

See clause 8.6.2. There is a single AIS insert process for each MT.

– *LCK generation process:*

See clause 8.6.3. There is a single LCK generation process for each MT.

***TC/Label sink processes**:*

See clause 8.2.2.

– *Label stack copy process:*

See clause 8.2.3.

***MPLS-TP specific GFP-F sink process**:*

See clause 8.4.2.

***Common GFP sink process**:*

See clause 8.5.3.1 of [ITU-T G.806]. GFP channel multiplexing is not supported (CMuxActive=false).

***Pq specific GFP sink process**:*

See clause 8.5.2.1 of [ITU-T G.806]. The GFP frames are demapped from the Pq payload area according to [ITU-T G.8040].

***Pq specific sink process**:*

NOTE – The VLI byte at the Pq_AP input of this function is ignored.

P31s specific:

MA: The signal label is recovered from the "Payload type" field in the MA byte as per clause 6.2.4.2 of [ITU-T G.806]. The signal label for "GFP mapping" in clause 2.1 of [ITU-T G.832] shall be expected. The accepted value of the signal label is also available at the P31s/ETH_A_Sk_MP.

Defects:

dPLM – See clause 6.2.4.2 of [ITU-T G.806].

dLFD – See clause 6.2.5.2 of [ITU-T G.806]

dUPM – See clause 8.4.2

dEXM – See clause 6.2.4.4 of [ITU-T G.806].

NOTE – dPLM is only defined for q = 31s. dPLM is assumed to be false for q = 11s, 12s, 32e.

Consequent actions:

The function shall perform the following consequent actions:

aSSF ← AI_TSF or dPLM or dLFD or dUPM or dEXM

aAIS ← AI_TSF or dPLM or dLFD or dUPM or dEXM

Defect correlations:

The function shall perform the following defect correlations to determine the most probable fault cause (see clause 6.4 of [ITU-T G.806]). This fault cause shall be reported to the EMF.

cPLM ← dPLM and (not AI_TSF)

cLFD ← dLFD and (not dPLM) and (not AI_TSF)

cEXM ← dEXM and (not dPLM) and (not dLFD) and (not AI_TSF)

cUPM ← dUPM and (not dEXM) and (not dPLM) and (not dLFD) and (not AI_TSF)

Performance monitoring:

For further study.

11.3.2 LCAS-capable Pq to MPLS-TP adaptation functions (Pq-X-L/MT_A; q=11s, 12s, 31s, 32e)

11.3.2.1 LCAS-capable Pq to MPLS-TP adaptation source function (Pq-X-L/MT_A_So)

This function maps MT_CI information onto a Pq-X-L_AI signal (q=11s, 12s, 31s, 32e).

Data at the Pq-X-L_AP is a Pq-X (q = 11s, 12s, 31s, 32e), having a payload as described in [ITU-T G.7043].

Symbol

The Pq-X-L/MT_A_So function symbol is shown in Figure 11-33.

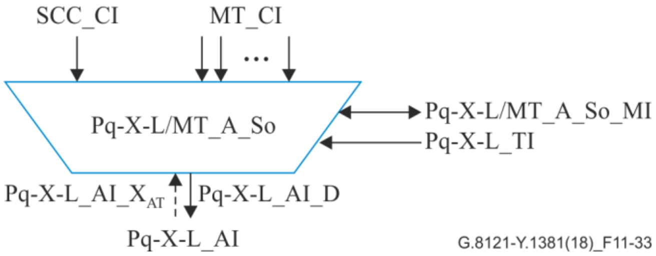


Figure 11-33 – Pq-X-L/MT_A_So symbol

Interfaces

The Pq-X-L/MT_A_So interfaces are described in Table 11-15.

Table 11-15 – Pq-X-L/MT_A_So interfaces

Input(s)	Output(s)
Each MT_CP: MT_CI_Data MT_CI_iPHB MT_CI_oPHB **SCC_CP**: SCC_CI_Data **Pq-X-L_AP**: Pq-X-L_AI_X_{AT} **Pq-X-L_TP**: Pq-X-L_TI_Clock Pq-X-L_TI_FrameStart **Pq-X-L/MT_A_So_MP**: Pq-X-L/MT_A_So_MI_SCCType Pq-X-L/MT_A_So_MI_Label[1…M] Pq-X-L/MT_A_So_MI_LSPType[1…M] Pq-X-L/MT_A_So_MI_CoS[1…M] Pq-X-L/MT_A_So_PHB2TCMapping[1…M] Pq-X-L/MT_A_So_MI_QoSEncodingMode[1…M]	**Pq-X-L_AP**: Pq-X-L_AI_Data Pq-X-L_AI_Clock Pq-X-L_AI_FrameStart

Table 11-15 – Pq-X-L/MT_A_So interfaces

Input(s)	Output(s)
Pq-X-L/MT_A_So_MI_Mode[1…M] Pq-X-L/MT_A_So_MI_GAL_Enable[1…M]	

Processes

A process diagram of this function is shown in Figure 11-34.

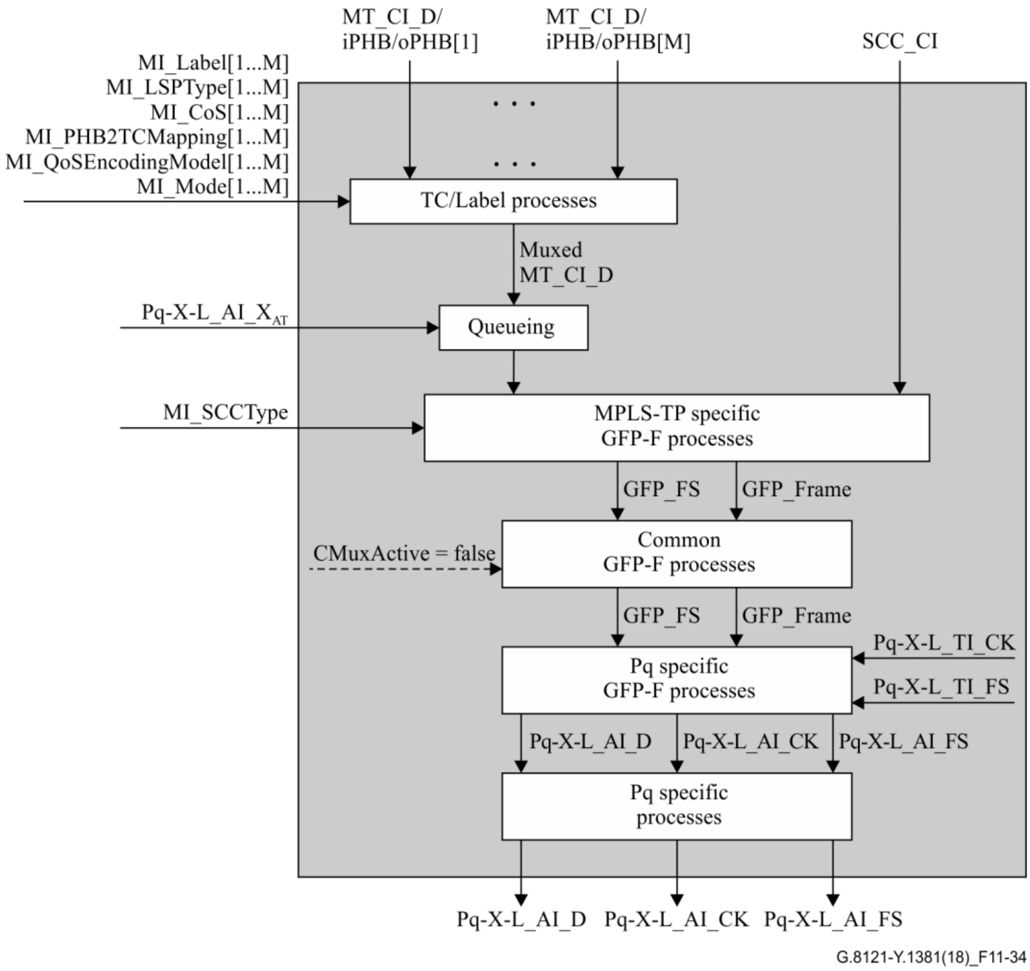


Figure 11-34 – Pq-X-L/MT_A_So process diagram

The processes have the same definition as in clause11.1.1.1.

Defects: None.

Consequent actions: None.

Defect correlations: None.

Performance monitoring: For further study.

11.3.2.2 LCAS-capable Pq to MPLS-TP adaptation sink function (Pq-X-L/MT_A_Sk)

This function extracts MT_CI information from the Pq-X-L_AI signal (q = 11s, 12s, 31s, 32e), delivering MT_CI.

Data at the Pq-X-L_AP is a Pq-Xv (q = 11s, 12s, 31s, 32e), having a payload as described in [ITU-T G.7043].

Symbol

The Pq-X-L/MT_A_Sk function symbol is shown in Figure 11-35.

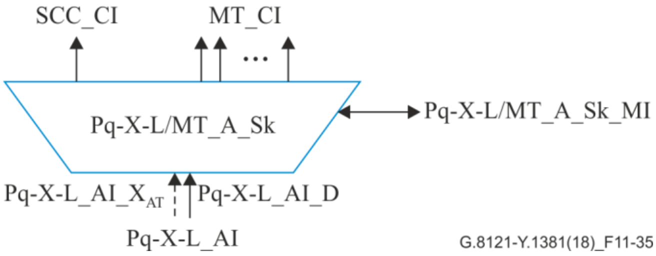


Figure 11-35 – Pq-X-L/MT_A_Sk symbol

Interfaces

The Pq-X-L/MT_A_Sk interfaces are described in Table 11-16.

Table 11-16 – Pq-X-L/MT_A_Sk interfaces

Input(s)	Output(s)
Pq-X-L_AP: Pq-X-L_AI_Data Pq-X-L_AI_ClocK Pq-X-L_AI_FrameStart Pq-X-L_AI_TSF Pq-X-L_AI_X_{AR} **Pq-X-L/MT_A_Sk_MP:** Pq-X-L/MT_A_Sk_MI_SCCType Pq-X-L/MT_A_Sk_MI_Label[1…M] Pq-X-L/MT_A_Sk_MI_LSPType[1…M] Pq-X-L/MT_A_Sk_MI_CoS[1…M] Pq-X-L/MT_A_Sk_MI_TC2PHBMapping[1…M] Pq-X-L/MT_A_Sk_MI_QoSDecodingMode[1…M] Pq-X-L/MT_A_Sk_MI_Mode[1…M] Pq-X-L//MT_A_Sk_MI_LCK_Period[1…M] Pq-X-L//MT_A_Sk_MI_LCK_P[1…M] Pq-X-L//MT_A_Sk_MI_Admin_State Pq-X-L//MT_A_Sk_MI_AIS_Period[1…M] Pq-X-L//MT_A_Sk_MI_AIS_P[1…M] Pq-X-L//MT_A_Sk_MI_GAL_Enable[1…M] Pq-X-L/MT_A_Sk_MI_LCK_Tool[1…M] Pq-X-L/MT_A_Sk_MI_AIS_Tool[1…M]	**Each MT_CP:** MT_CI_Data MT_CI_iPHB MT_CI_oPHB MT_CI_SSF MT_CI_LStack **SCC_CP:** SCC_CI_Data SCC_CI_SSF **Pq-X-L/MT_A_Sk_MP:** Pq-X-L/MT_A_Sk_MI_AcSL Pq-X-L/MT_A_Sk_MI_AcEXI Pq-X-L/MT_A_Sk_MI_LastInvalidUPI Pq-X-L/MT_A_Sk_MI_cPLM Pq-X-L/MT_A_Sk_MI_cLFD Pq-X-L/MT_A_Sk_MI_cEXM Pq-X-L/MT_A_Sk_MI_cUPM

Processes

The Pq-X-L/MT_A_Sk process diagram is illustrated in Figure 11-36.

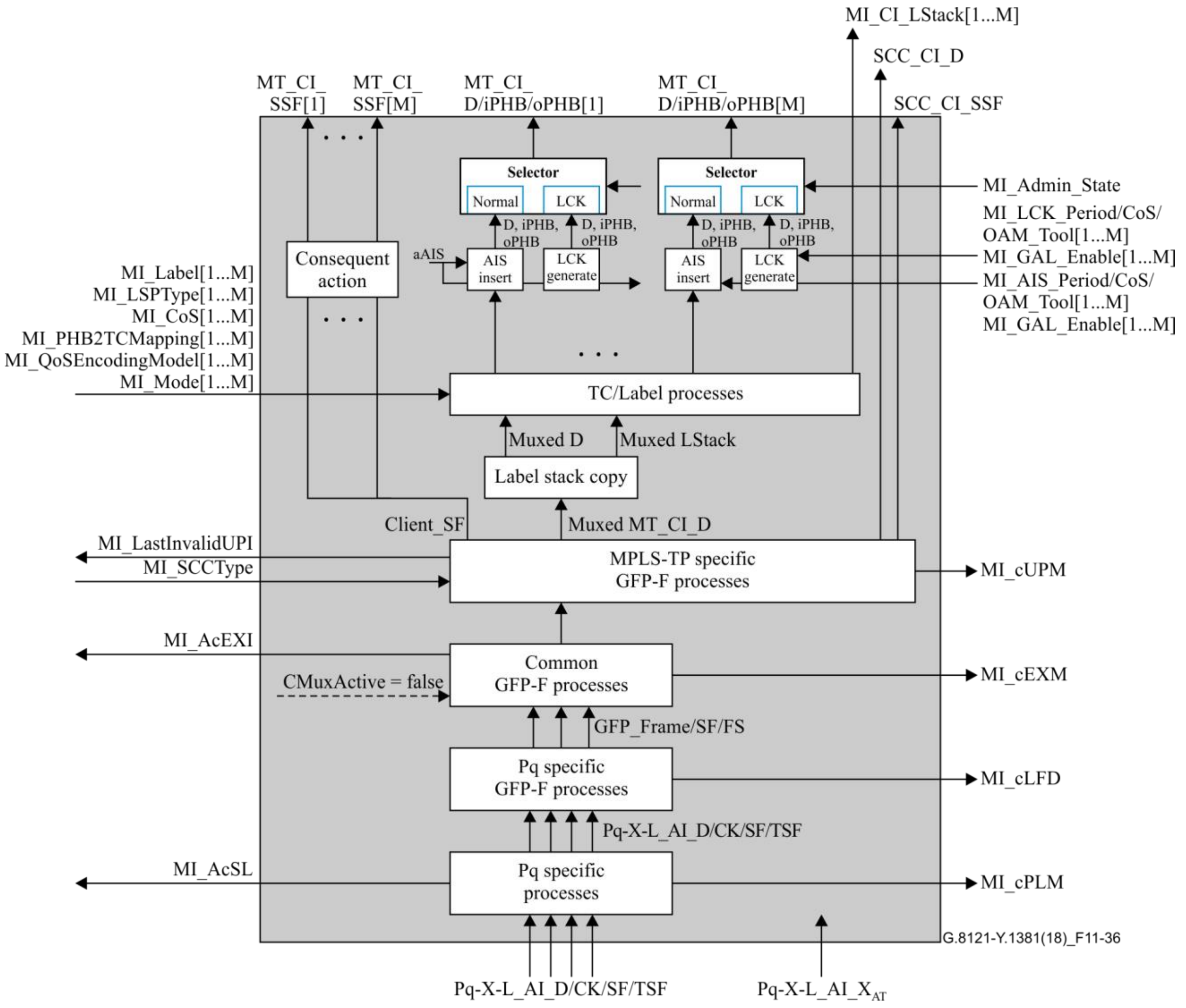


Figure 11-36 – Pq-X-L/MT_A_Sk process diagram

See process diagram and process description in clause 11.1.1.2. The additional Pq-X-L_AI_X_{AR} interface is not connected to any of the internal processes.

Defects:

dPLM – See clause 6.2.4.2 of [ITU-T G.806].

dLFD – See clause 6.2.5.2 of [ITU-T G.806].

dUPM – See clause 8.4.2.

dEXM – See clause 6.2.4.4 of [ITU-T G.806].

NOTE: dPLM is only defined for q = 31s. dPLM is assumed to be false for q = 11s, 12s, 32e.

Consequent actions:

The function shall perform the following consequent actions:

aSSF ← AI_TSF or dPLM or dLFD or dUPM or dEXM

aAIS ← AI_TSF or dPLM or dLFD or dUPM or dEXM

Defect correlations:

The function shall perform the following defect correlations to determine the most probable fault cause (see clause 6.4 of [ITU-T G.806]). This fault cause shall be reported to the EMF.

cPLM ← dPLM and (not AI_TSF)

cLFD ← dLFD and (not dPLM) and (not AI_TSF)

cEXM ← dEXM and (not dPLM) and (not dLFD) and (not AI_TSF)

cUPM ← dUPM and (not dEXM) and (not dPLM) and (not dLFD) and (not AI_TSF)

Performance monitoring:

For further study.

11.4 Ethernet to MPLS-TP adaptation function

11.4.1 ETH to MPLS-TP adaptation function (ETH/MT_A)

11.4.1.1 ETH to MPLS-TP adaptation source function (ETH/MT_A_So)

Symbol

The ETH/MT_A_So function symbol is shown in Figure 11-37.

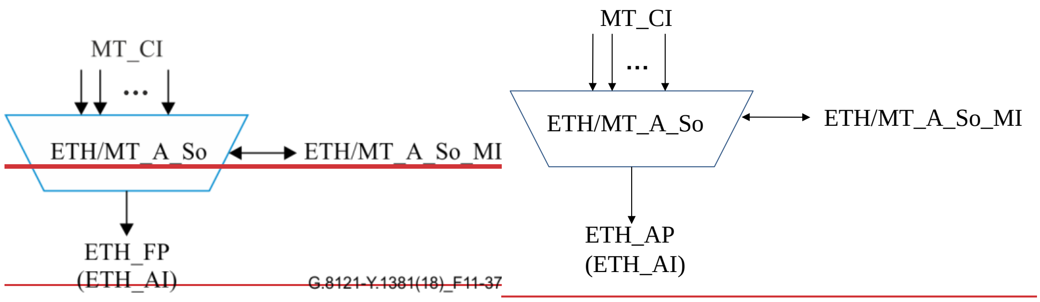


Figure 11-37 – ETH/MT_A_So symbol

Interfaces

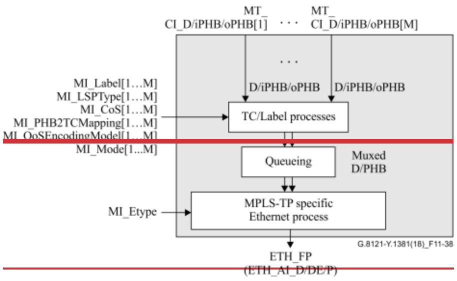
The ETH/MT_A_So interfaces are described in Table 11-17.

Table 11-17 – ETH/MT_A_So interfaces

Input(s)	Output(s)
Each MT_CP: MT_CI_Data[1…M] MT_CI_iPHB[1…M] MT_CI_oPHB[1…M] **ETH/MT_A_So_MP:** ETH/MT_A_So_MI_Label[1…M] ETH/MT_A_So_MI_LSPType[1…M] ETH/MT_A_So_MI_CoS[1…M] ETH/MT_A_So_PHB2TCMapping[1…M] ETH/MT_A_So_MI_QoSEncodingMode[1…M] ETH/MT_A_So_MI_Mode[1…M] ETH/MT_A_So_MI_Etype ETH/MT_A_So_MI_GAL_Enable[1…M]	**ETHH~~Yn~~_AP:** ETH_AI_Data ETH_AI_P ETH_AI_DE

Processes

A process diagram of this function is shown in Figure 11-38.



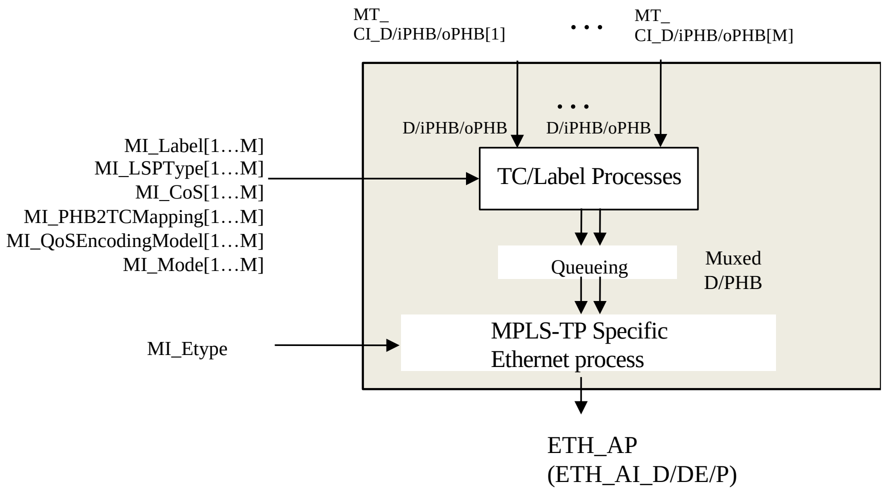


Figure11-38 – ETH/MT_A_So process

– *TC/Label processing*

See clause 8.2.1.

– *Queuing process*:

See clause 8.3.

– *MPLS-TP specific Ethernet process*:

This process inserts the Ethertype for MPLS-TP packets according to [IETF RFC 5332].

Defects: None.

Consequent actions: None.

Defect correlations: None.

Performance monitoring: For further study.

11.4.1.2 ETH to MPLS-TP adaptation sink function (ETH/MT_A_Sk)

Symbol

The ETH/MT_A_Sk function symbol is shown in Figure 11-39.

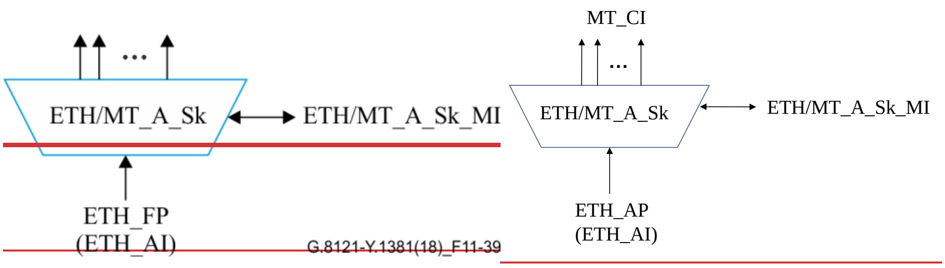


Figure 11-39 – ETH/MT_A_Sk symbol

Interfaces

The ETH/MT_A_Sk interfaces are described in Table 11-18.

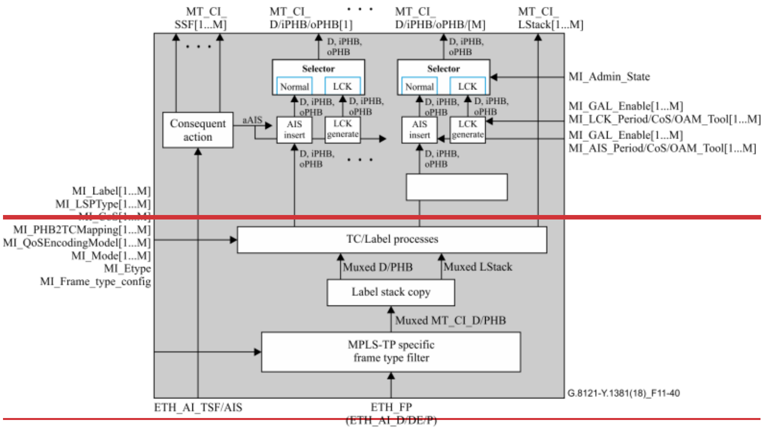
Table 11-18 – ETH/MT_A_Sk interfaces

Input(s)	Output(s)
ETH_AP: ETH_AI_Data ETH_AI_P ETH_AI_DE ETH_AI_TSF ETH_AI_AIS **ETH/MT_A_Sk_MP:** ETH/MT_A_Sk_MI_Etype ETH/MT_A_Sk_MI_Frame_Type_Config ETH/MT_A_Sk_MI_Label[1…M] ETH/MT_A_Sk_MI_LSPType[1…M] ETH/MT_A_Sk_MI_CoS[1…M] ETH/MT_A_Sk_MI_TC2PHBMapping[1…M] ETH/MT_A_Sk_MI_QoSDecodingMode[1…M] ETH/MT_A_Sk_MI_Mode[1…M]	**Each MT_CP:** MT_CI_Data[1...M] MT_CI_iPHB[1...M] MT_CI_oPHB[1...M] MI_CI_Lstack[1...M]

Input(s)	Output(s)
ETH/MT_A_Sk_MI_GAL_Enable[1…M] ETH/MT_A_Sk_MI_Admin_State ETH/MT_A_Sk_MI_LCK_Period[1…M] ETH/MT_A_Sk_MI_LCK_CoS[1…M] ETH/MT_A_Sk_MI_AIS_Period[1…M] ETH/MT_A_Sk_MI_AIS_CoS[1…M] ETH/MT_A_Sk_MI_LCK_OAM_Tool[1…M] ETH/MT_A_Sk_MI_AIS_OAM_Tool[1…M]	

Processes

A process diagram of this function is shown in Figure 11-40.



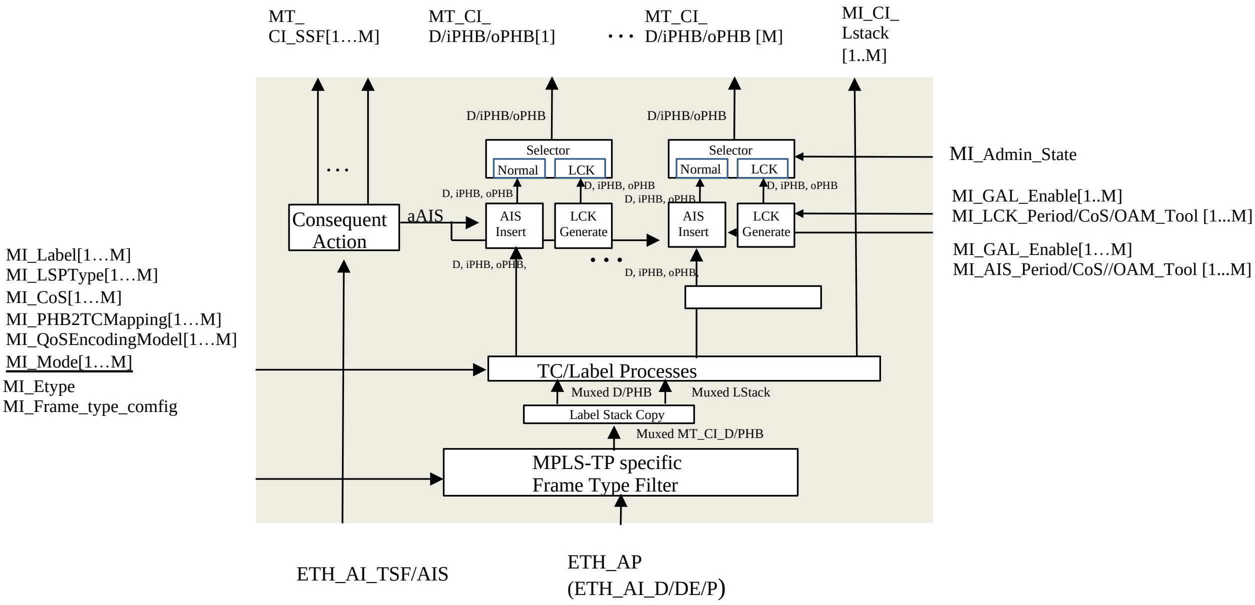


Figure 11-40 – ETH/MT_A_Sk process

– *Selector generation process*:

See clause 8.6.1. The normal CI is blocked if Admin_State = LOCKED.

– *AIS insert process*:

See clause 8.6.2. There is a single AIS insert process for each MT.

– *LCK generation process*:

See clause 8.6.3. There is a single LCK generation process for each MT.

– *TC/Label sink processes*:

See clause 8.2.2.

– *Label stack copy process*:

See clause 8.2.3.

– *MPLS-TP specific Filter process:*

This process is for the reception process of the Ethertype for MPLS-TP packets according to [IETF RFC 5332].

Defects: None.

Consequent actions: For further study.

Defect correlations: None.

Performance monitoring: For further study.

Annex A

Mapping MPLS-TP packets to OTN using IMP

(This annex forms an integral part of this Recommendation.)

Per clause 17.11 of [ITU-T G.709], packet clients other than Ethernet can be mapped to OTN using IMP (Idle Mapping Procedure) by first mapping the packet client to Ethernet, and then mapping the Ethernet signal with IMP.

Figure A.1 illustrates the compound function ODUflexP/MT-imp_A that consists of ODUflexP/ETH-imp_A function, ETHnull_FT function and ETH/MT_A function.

The ODUflexP/ETH-imp_A function is specified in cluaue 14.3.21 of [ITU-T G.798]. The ETHnull_FT function is specified as ETHx_FT in clause 9.2.1 of [ITU-T G.8021] without OAM processes. ETH/MT_A function is specified in clause 9.4 of this Recommendation.

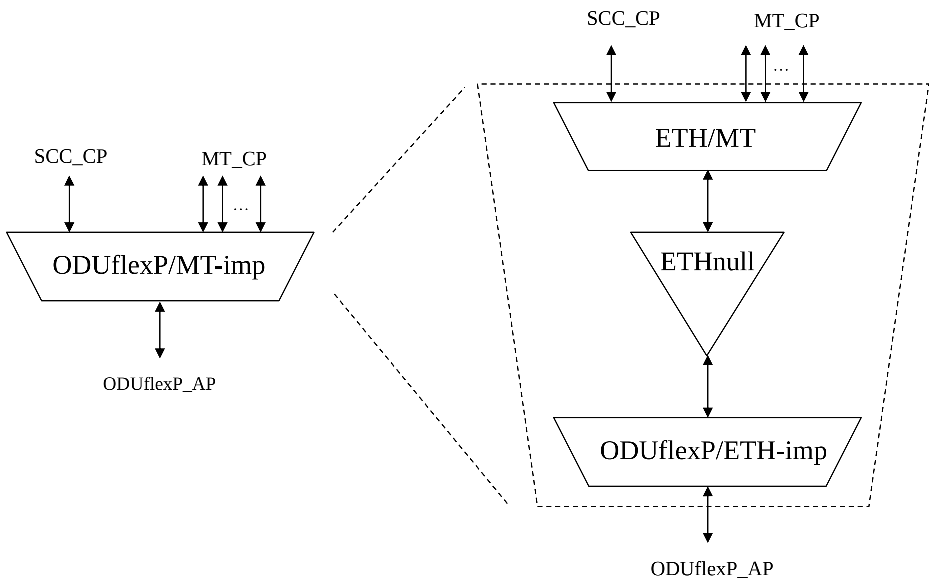


Figure A.1 – Expression of ODUflexP/MT-imp_A as a compound function

Appendix I

Examples of processing of packets with expired TTL

(This appendix does not form an integral part of this Recommendation.)

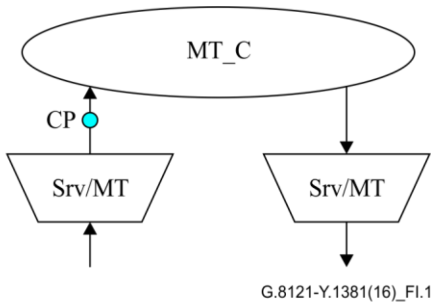
MPLS-TP packets received with an expired TTL shall not be forwarded. However, MPLS-TP OAM packets received with an expired TTL can be processed and their processing can happen at different locations (i.e., from different atomic functions) within a MPLS-TP equipment.

The proper behaviour depends on the MPLS-TP connection configuration within the node. The following examples are considered and are described:

– Intermediate node with no MIPs
– Intermediate node – interface MIPs
– Intermediate node – node MIP
– Terminating node – down MEP or node MEP
– Terminating node – up MEP (with interface MIP)

NOTE – As indicated in clause 9.4.2.2.2, the MI_DS_MP_Type parameter should be properly configured by the EMF and not exposed to the operator as a configuration parameter of the NE management. The examples described in this appendix provide guidelines on how the EMF can properly configure the MI_DS_MP_Type.

Figure I.1 describes the behaviuor of an intermediate node with no MIPs using the atomic functions defined in this Recommendation:

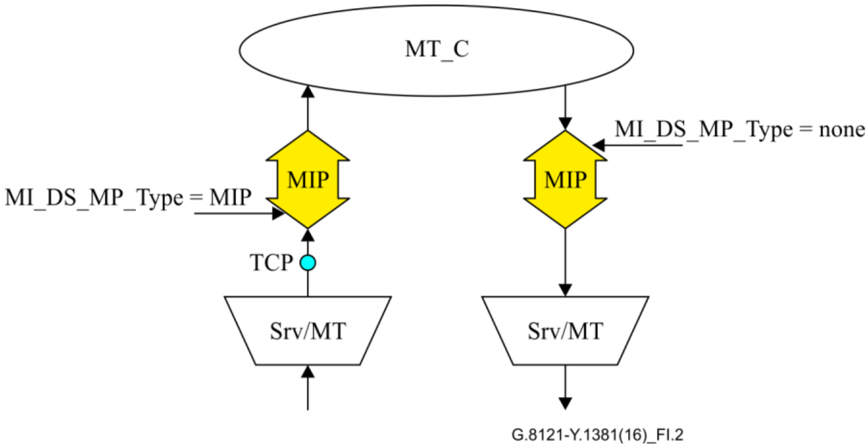


NOTE – Srv can be any server (MT or non-MT).

Figure I.1 – Intermediate node with no MIPs

The Server/MT_A_Sk is connected to the MT_C via a MT_CP. Therefore, the TTL decrement process, as defined in clause 8.2.2, will discard all the MPLS-TP packets (user data or OAM) that are received with an expired TTL.

Figure I.2 describes the behaviour of an intermediate node supporting per-interface MIPs using the atomic functions defined in this Recommendation:



NOTE – Srv can be any server (MT or non-MT).

Figure I.2 – Intermediate node with per-interface MIPs

The Server/MT_A_Sk is connected to ingress MIP via a MT_TCP. Therefore, the TTL decrement process, as defined in clause 8.2.2, will forward all the MPLS-TP packets (user data or OAM) that are received with an expired TTL to the ingress MIP.

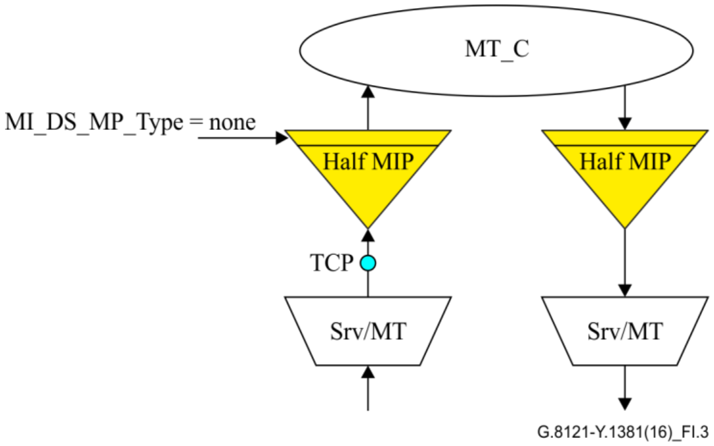
The MTDi_TT_Sk atomic function within the ingress MIP will process all the MPLS-TP OAM packets received with an expired TTL and which are targeted to the ingress MIP.

The TTL check process in the MTDi/MT_A_Sk within the ingress MIP, as defined in clause 9.4.2.2.2, is properly configured by the EMF (MI_DS_MP_Type=MIP) to drop all the MPLS-TP user data packets received with an expired TTL and to forward all the MPLS-TP OAM packets received with an expired TTL together (i.e., with fate share) with all the MPLS-TP packets received with a non-expired TTL.

These packets are forwarded up to the egress MIP where the MTDi_TT_Sk atomic function will process all the MPLS-TP OAM packets received with an expired TTL and which are targeted to the egress MIP.

The TTL check process in the MTDi/MT_A_Sk within the egress MIP, as defined in clause 9.4.2.2.2, is properly configured by the EMF (MI_DS_MP_Type=none) to drop all the MPLS-TP packets received with an expired TTL. Although MPLS-TP user data packets with an expired TTL will never arrive at this point, this check will ensure also that any MPLS-TP OAM packet with an expired TTL is not forwarded.

Figure I.3 describes the behaviour of an intermediate node with a per-node MIP using the atomic functions defined in this Recommendation. The per-node MIP is modelled as being composed by two half-MIPs on each side of the MT_C:



NOTE – Srv can be any server (MT or non-MT).

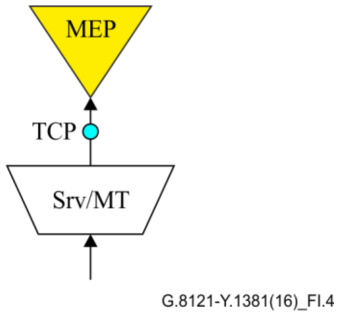
Figure I.3 – Intermediate node with a per-node MIP

The Server/MT_A_Sk is connected to ingress MIP via a MT_TCP. Therefore, the TTL decrement process, as defined in clause 8.2.2, will forward all the MPLS-TP packets (user data or OAM) that are received with an expired TTL to the ingress half-MIP.

The MTDi_TT_Sk atomic function within the ingress half-MIP will process all the MPLS-TP OAM packets received with an expired TTL and which are targeted to the node MIP.

The TTL check process in the MTDi/MT_A_Sk, as defined in clause 9.4.2.2.2, is properly configured by the EMF (MI_DS_MP_Type=none) to drop all the MPLS-TP packets (user data or OAM) that are received with an expired TTL.

Figure I.4 describes the behaviour of a terminating node with a down MEP or a per-node MEP using the atomic functions defined in this Recommendation. These two cases are modelled in the same way:



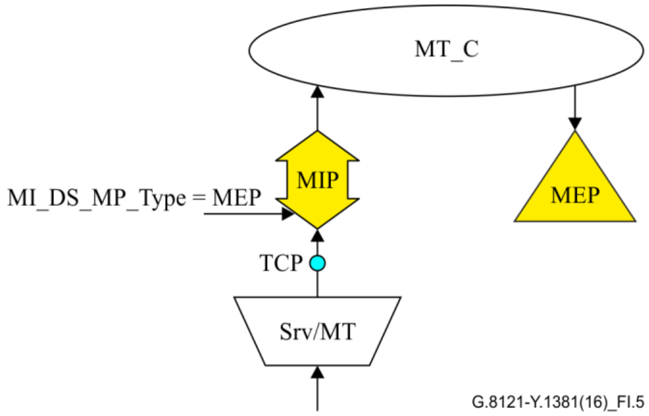
NOTE – Srv can be any server (MT or non-MT).

Figure I.4 – Terminating node with a down MEP or node MEP

The Server/MT_A_Sk is connected to MEP via a MT_TCP. Therefore, the TTL decrement process, as defined in clause 8.2.2, will forward all the MPLS-TP packets (user data or OAM) that are received with an expired TTL to the MEP.

The MEP terminates the MPLS-TP trail and processes all the MPLS-TP packets it receives regardless of whether the TTL has expired or not.

Figure I.5 describes the behaviour of a terminating node with an Up MEP, and therefore a per-interface ingress MIP, using the atomic functions defined in this Recommendation:



NOTE – Srv can be any server (MT or non-MT).

Figure I.5 – Terminating node with an Up MEP (and a per-interface MIP)

The Server/MT_A_Sk is connected to ingress MIP via a MT_TCP. Therefore, the TTL decrement process, as defined in clause 8.2.2, will forward all the MPLS-TP packets (user data or OAM) that are received with an expired TTL to the ingress MIP.

The MTDi_TT_Sk atomic function within the ingress MIP will process all the MPLS-TP OAM packets received with an expired TTL and which are targeted to the ingress MIP.

The TTL check process in the MTDi/MT_A_Sk within the ingress MIP, as defined in clause 9.4.2.2.2, is properly configured by the EMF (MI_DS_MP_Type=MEP) to forward all the MPLS-TP packets (user data or OAM) that are received with an expired TTL together (i.e., with fate share) with all the MPLS-TP packets received with a non-expired TTL.

These packets are forwarded up to the Up MEP that terminates the MPLS-TP trail and processes all the MPLS-TP packets it receives regardless of whether the TTL has expired or not.

Appendix II

Flow of PHB information through MEP and MIP

(This appendix does not form an integral part of this Recommendation.)

This Recommendation describes the various atomic functions that comprise MIPs and MEPs:

– MEP: MT/MT_A, MT_TT, MTDe/MT_A, MTDe_TT

– MIP: Two MHFs each comprising MTDi_TT, MTDi/MT_A

The handling of PHB values is described in clauses 8.2, 9.2, 9.3 and 9.4. The PHB information is passed between the atomic functions in characteristic information (CI) and adapted information (AI).

As described in clause 10 of [ITU-T G.8110.1], the MPLS-TP Diffserv architecture supports two models: the "Short Pipe" model, and the "Uniform" model. To support this, it is necessary in certain cases to pass two PHB values in the CI and the AI, referred to as the incoming PHB (iPHB) and outgoing PHB (oPHB). In other cases, only a single PHB values needs to be passed between the atomic functions.

Figures 10-1 and 10-2 of [ITU-T G.8110.1] are the reference diagrams showing how PHB values are used in the two models. However, these do not show all of the atomic functions defined in this Recommendation that comprise MEPs and MIPs. So, in particular, it is unclear at a first glance why in some cases the AI carries a single PHB value, while in other cases it carries separate iPHB and oPHB values.

Figure II.1 and II.2 show a MEP and a MIP and illustrate the flow of PHB information through them. The iPHB values are shown in green, oPHB values in red, and where only a single PHB value is used, this is shown in black.

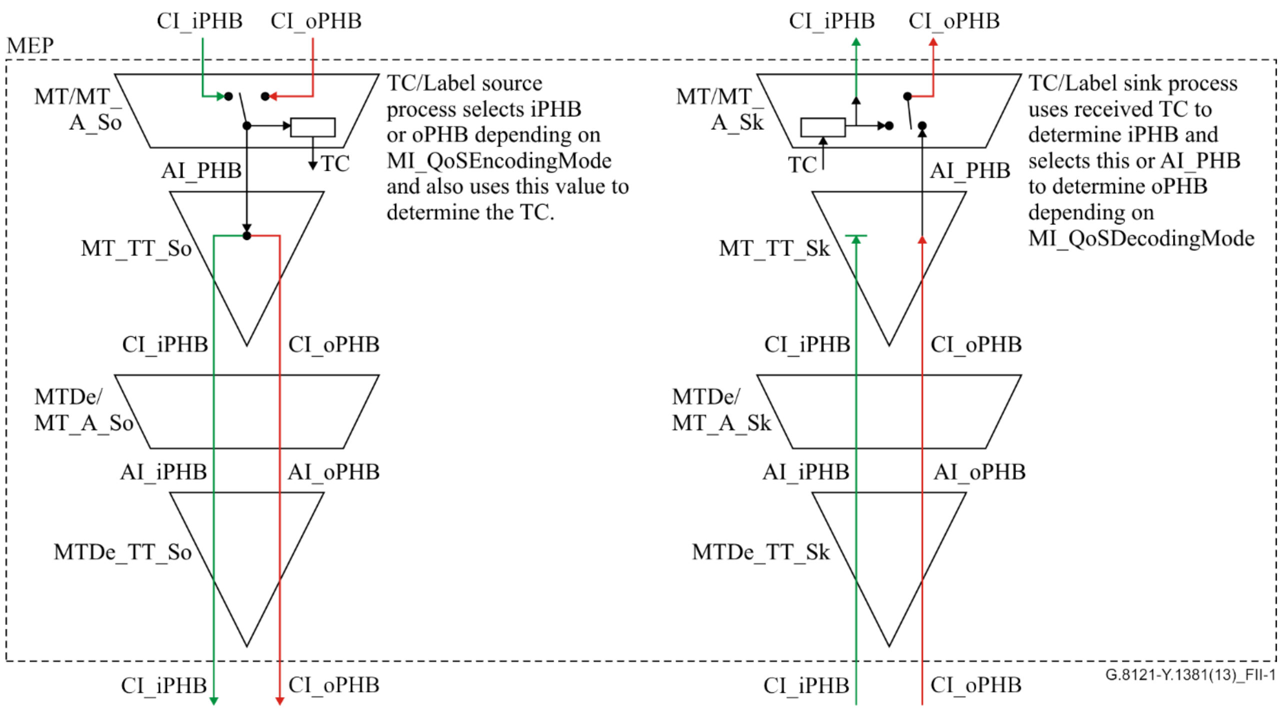


Figure II.1 – Flow of PHB information through a MEP

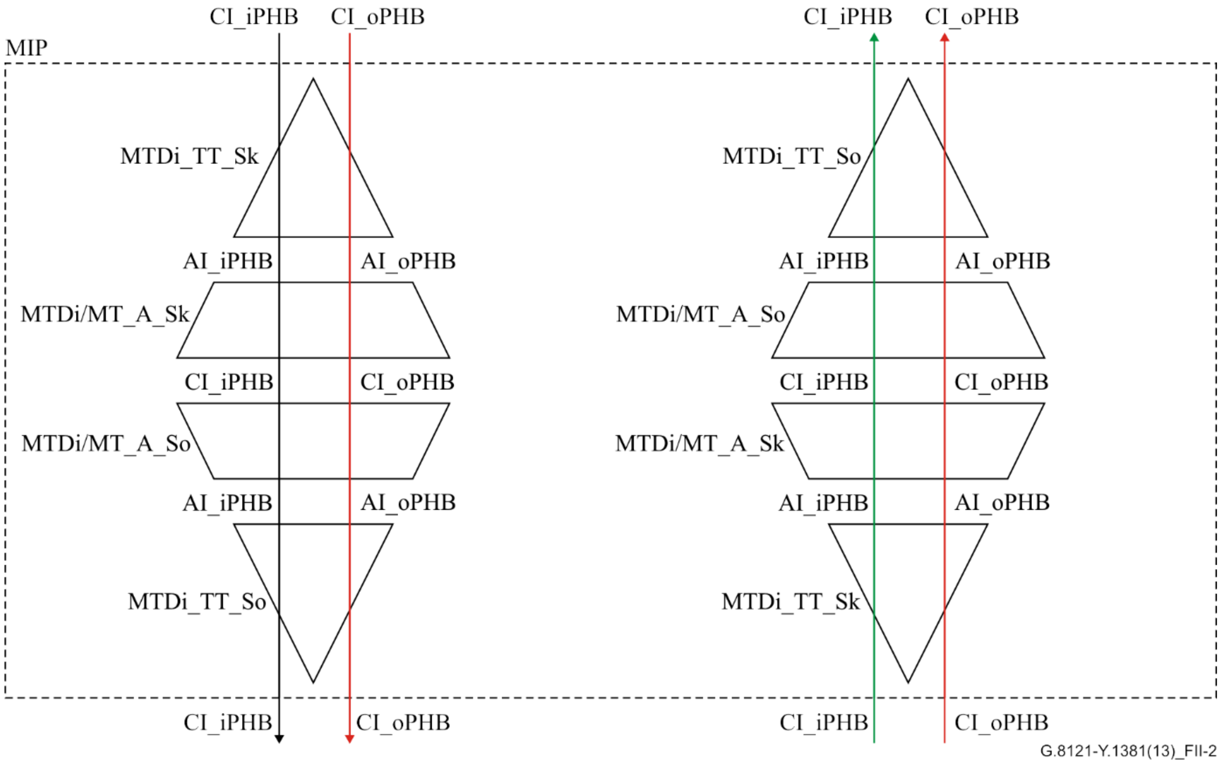


Figure II.2 – Flow of PHB information through a MIP

By considering a case where the sink side of one MEP is connected to the source side of another MEP, via a MIP, it becomes clear why both the iPHB and oPHB values must be passed through the MIP without modification. This is illustrated in Figure II.3. The same logic would apply if the MTDe_TT and MTDe/MT_A atomic functions were used without their associated MT_TT and MT/MT_A functions.

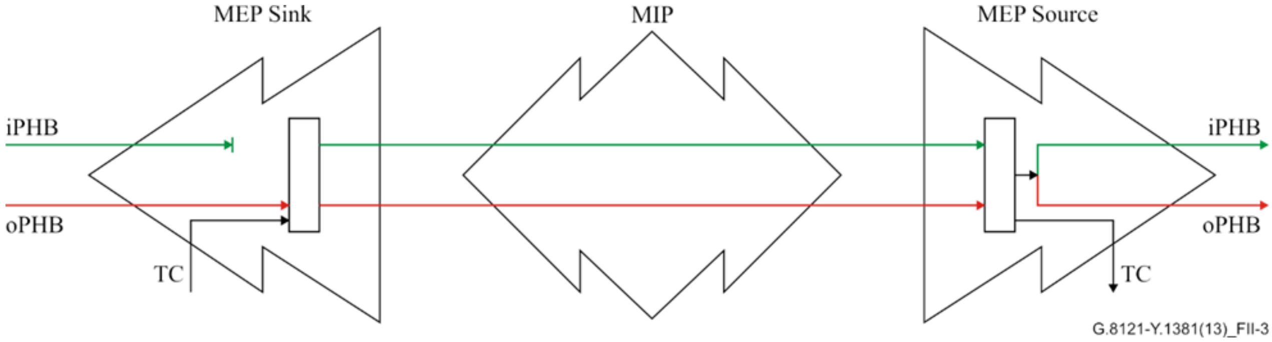


Figure II.3 – Example showing two MEPs and a MIP

Appendix III

SDL descriptions

(This appendix does not form an integral part of this Recommendation.)

In this Recommendation, detailed characteristics of equipment functional blocks are described with specification and description language (SDL) diagrams specified in [b-ITU-T Z.100]. The SDLdiagrams use the following conventions, as shown in Figure III.1.

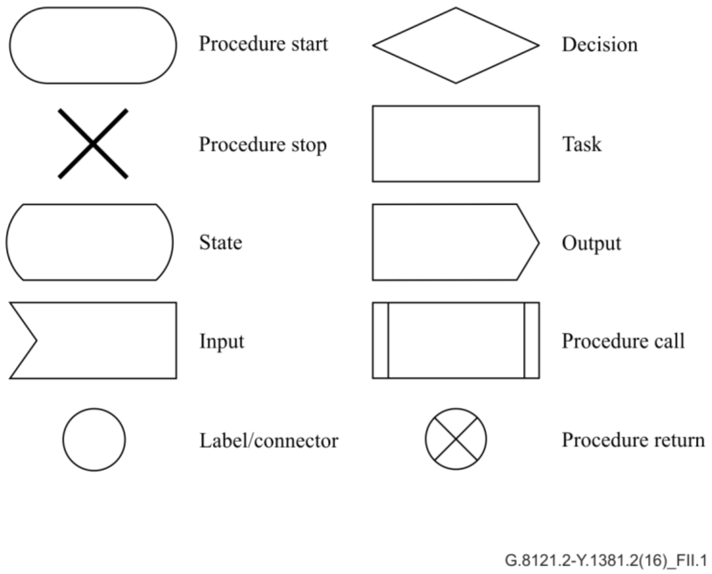


Figure III.1 – SDL symbols

Bibliography

[b-ITU-T G.780] Recommendation ITU-T G.780/Y.1351 (2010), *Terms and definitions for synchronous digital hierarchy (SDH) networks.*

[b-ITU-T G.8151] Recommendation ITU-T G.8151/Y.1374 (2018), *Management aspects of the MPLS-TP network element*.

[b-ITU-T Z.100] Recommendation ITU-T Z.100 (2016), *Specification and Description Language- Overview of SDL-2010.*

[b-IETF RFC 6378] IETF RFC 6378 (2011), *MPLS Transport Profile (MPLS-TP) Linear Protection.*

ITU-T Y-SERIES RECOMMENDATIONS

GLOBAL INFORMATION INFRASTRUCTURE, INTERNET PROTOCOL ASPECTS, NEXT-GENERATION NETWORKS, INTERNET OF THINGS AND SMART CITIES

GLOBAL INFORMATION INFRASTRUCTURE	
General	Y.100–Y.199
Services, applications and middleware	Y.200–Y.299
Network aspects	Y.300–Y.399
Interfaces and protocols	Y.400–Y.499
Numbering, addressing and naming	Y.500–Y.599
Operation, administration and maintenance	Y.600–Y.699
Security	Y.700–Y.799
Performances	Y.800–Y.899
INTERNET PROTOCOL ASPECTS	
General	Y.1000–Y.1099
Services and applications	Y.1100–Y.1199
Architecture, access, network capabilities and resource management	Y.1200–Y.1299
Transport	**Y.1300–Y.1399**
Interworking	Y.1400–Y.1499
Quality of service and network performance	Y.1500–Y.1599
Signalling	Y.1600–Y.1699
Operation, administration and maintenance	Y.1700–Y.1799
Charging	Y.1800–Y.1899
IPTV over NGN	Y.1900–Y.1999
NEXT GENERATION NETWORKS	
Frameworks and functional architecture models	Y.2000–Y.2099
Quality of Service and performance	Y.2100–Y.2199
Service aspects: Service capabilities and service architecture	Y.2200–Y.2249
Service aspects: Interoperability of services and networks in NGN	Y.2250–Y.2299
Enhancements to NGN	Y.2300–Y.2399
Network management	Y.2400–Y.2499
Network control architectures and protocols	Y.2500–Y.2599
Packet-based Networks	Y.2600–Y.2699
Security	Y.2700–Y.2799
Generalized mobility	Y.2800–Y.2899
Carrier grade open environment	Y.2900–Y.2999
FUTURE NETWORKS	Y.3000–Y.3499
CLOUD COMPUTING	Y.3500–Y.3999
INTERNET OF THINGS AND SMART CITIES AND COMMUNITIES	
General	Y.4000–Y.4049
Definitions and terminologies	Y.4050–Y.4099
Requirements and use cases	Y.4100–Y.4249
Infrastructure, connectivity and networks	Y.4250–Y.4399
Frameworks, architectures and protocols	Y.4400–Y.4549
Services, applications, computation and data processing	Y.4550–Y.4699
Management, control and performance	Y.4700–Y.4799
Identification and security	Y.4800–Y.4899
Evaluation and assessment	Y.4900–Y.4999

For further details, please refer to the list of ITU-T Recommendations.

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series D	Tariff and accounting principles and international telecommunication/ICT economic and policy issues
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	**Transmission systems and media, digital systems and networks**
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Cable networks and transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Environment and ICTs, climate change, e-waste, energy efficiency; construction, installation and protection of cables and other elements of outside plant
Series M	Telecommunication management, including TMN and network maintenance
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling, and associated measurements and tests
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	Data networks, open system communications and security
Series Y	**Global information infrastructure, Internet protocol aspects, next-generation networks, Internet of Things and smart cities**
Series Z	Languages and general software aspects for telecommunication systems