International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

G.8021/Y.1341

Amendment 1

(10/2012)

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

Packet over Transport aspects – Ethernet over Transport aspects

SERIES Y: GLOBAL INFORMATION INFRASTRUCTURE, INTERNET PROTOCOL ASPECTS AND NEXT-GENERATION NETWORKS

Internet protocol aspects – Transport

Characteristics of Ethernet transport network equipment functional blocks

Amendment 1

Recommendation ITU-T G.8021/Y.1341 (2012) – Amendment 1

ITU-T

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
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.8021/Y.1341

Characteristics of Ethernet transport network equipment functional blocks

Amendment 1

Summary

Amendment 1 to Recommendation ITU-T G.8021/Y.1341 (2012) presents the support of the intermediate retrieval of on-demand performance measurement results, and enhancements concerning the modelling of delay measurement and client signal failure functions.

History

Edition	Recommendation	Approval	Study Group
1.0	ITU-T G.8021/Y.1341	2004-08-22	15
1.1	ITU-T G.8021/Y.1341 (2004) Amd. 1	2006-06-06	15
2.0	ITU-T G.8021/Y.1341	2007-12-22	15
2.1	ITU-T G.8021/Y.1341 (2007) Amd. 1	2009-01-13	15
2.2	ITU-T G.8021/Y.1341 (2007) Amd. 2	2010-02-22	15
3.0	ITU-T G.8021/Y.1341	2010-10-22	15
3.1	ITU-T G.8021/Y.1341 (2010) Amd .1	2011-07-22	15
4.0	ITU-T G.8021/Y.1341	2012-05-07	15
4.1	ITU-T G.8021/Y.1341 (2012) Amd.1	2012-10-29	15

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 not 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/.

Recommendation ITU-T G.8021/Y.1341

Characteristics of Ethernet transport network equipment functional blocks

Amendment 1

1) Clause 8.1.9, Loss measurement (LM) processes

1.1) Clause 8.1.9.1

Update clause 8.1.9.1, Overview, as indicated below:

Figure 8-35 shows the different processes inside MEPs and MIPs that are involved in the on-demand loss measurement protocol.

The MEP on-demand OAM source insertion process is defined in clause 9.4.1.1, the MEP on-demand OAM sink extraction process in clause 9.4.1.2, the MIP on-demand OAM sink extraction process in clause 9.4.2.2, and the MIP on-demand OAM source insertion process in clause 9.4.2.1. In summary, they insert and extract ETH_CI OAM signals into and from the stream of ETH_CI_D traffic units together with the complementing P and D signals going through an MEP and MIP; the extraction is based on MEL and Opcode. Furthermore, the insertion process inserts the correct MEL and SA values into the OAM traffic units.

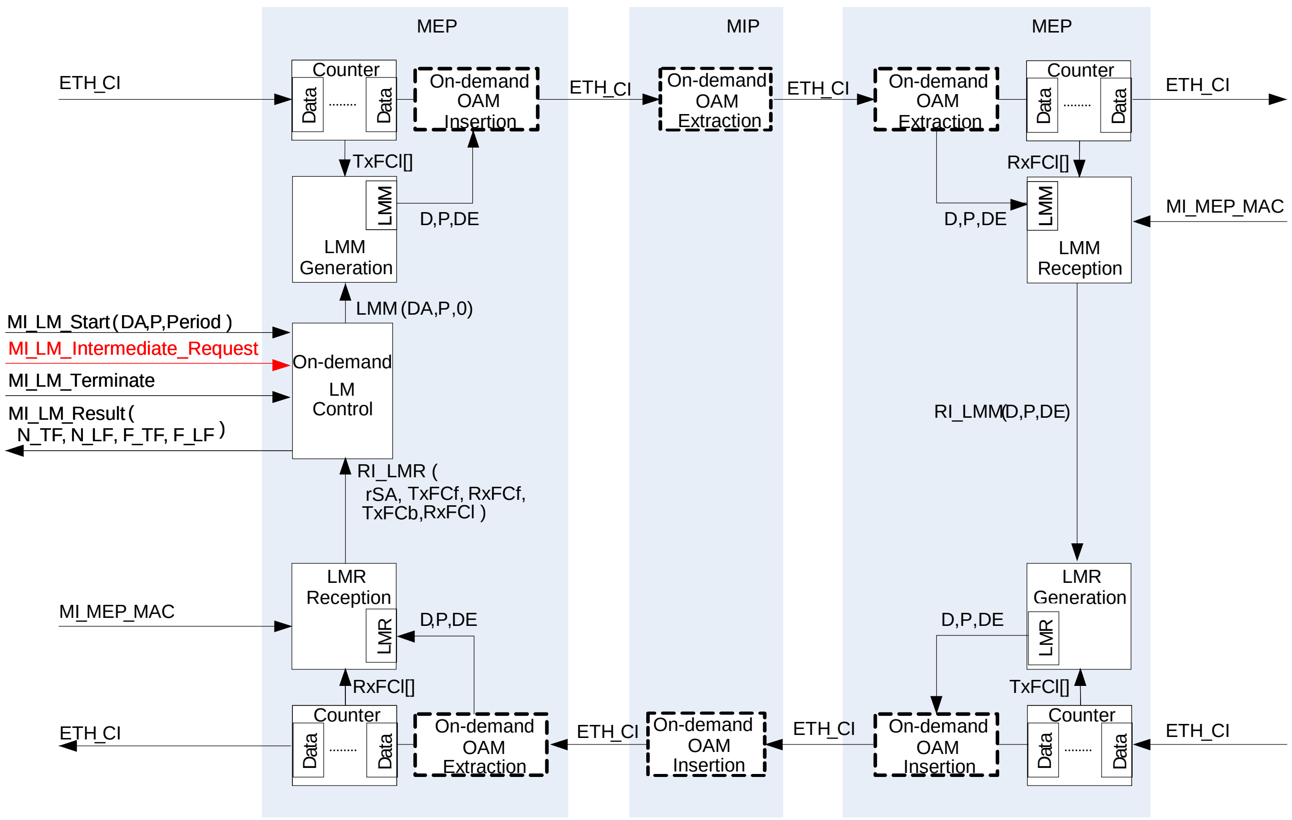

Figure 8-35 – Overview of processes involved with on-demand loss measurement

The on-demand LM control process controls the on-demand LM protocol. The protocol is activated upon receipt of the MI_LM_Start(DA,P,Period) signal and remains activated until the MI_LM_Terminate signal is received.

The result is communicated via the MI_LM_Result(N_TF, N_LF, F_TF, F_LF) signal when the process is terminated by the MI_LM_Terminate signal or when an intermediate result is requested via the MI_LM_Intermediate_Request signal. If the on-demand LM control process activates the multiple monitoring on different CoS levels simultaneously, each result is independently managed per CoS level.

The LMM generation process generates an LMM traffic unit that passes transparently through MIPs, but that will be processed by the LMM reception process in MEPs. The LMR generation process generates an LMR traffic unit in response to the receipt of an LMM traffic unit. The LMR reception process receives and processes the LMR traffic units.

…

1.2) Clause 8.1.9.2

Update clause 8.1.9.2, LM control process, as indicated below:

The behaviour of the on-demand LM control process is defined in Figure 8-37.

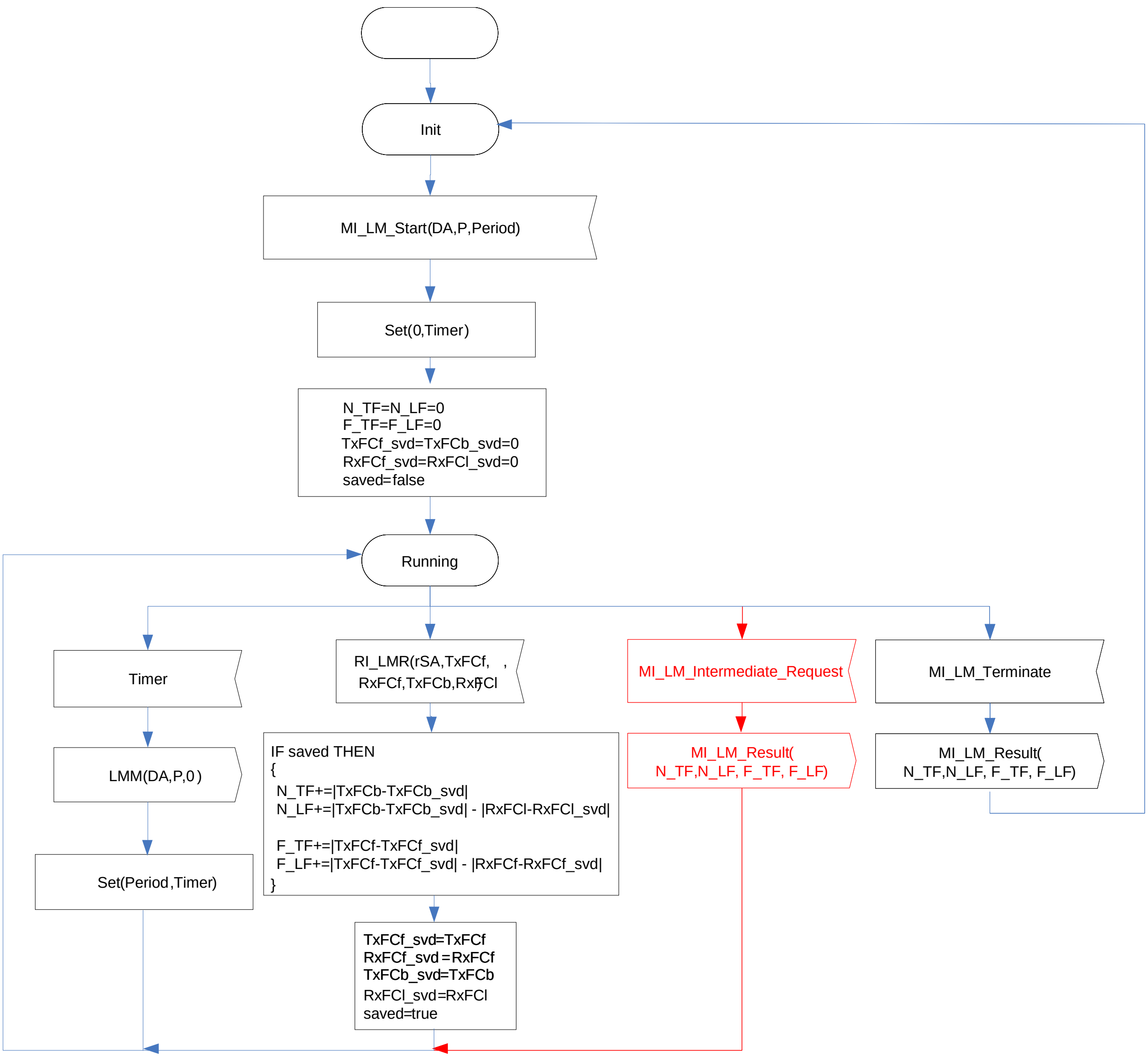

Figure 8-37 – On-demand LM control behaviour

Upon receipt of the MI_LM_Start(DA,P,Period), the LM protocol is started. Every period the generation of an LMM frame is triggered (using the LMM(DA,P,0) signal), until the MI_LM_Terminate signal is received.

The received counters are used to count the near end and far end transmitted and lost frames. This result is reported using the MI_LM_Result(N_TF, N_LF, F_TF, F_LF) signal after the receipt of the MI_LM_Terminate signal or of the MI_LM_Intermediate_Request signal.

2) Clause 8.1.10, Delay measurement (DM) processes

2.1) Clause 8.1.10.1

Update clause 8.1.10.1, Overview, as indicated below:

Figure 8-47 shows the different processes inside MEPs and MIPs that are involved in the on-demand delay measurement protocol.

The MEP on-demand OAM source insertion process is defined in clause 9.4.1.1, the MEP on-demand OAM sink extraction process in clause 9.4.1.2, the MIP on-demand OAM sink extraction process in clause 9.4.2.2, and the MIP on-demand OAM source insertion process in clause 9.4.2.1. In summary, they insert and extract ETH_CI OAM signals into and from the stream of ETH_CI_D traffic units and the complementing P and D signals going through a MEP and MIP; the extraction is based on MEL and Opcode. Furthermore, the insertion process inserts the correct MEL and SA values into the OAM traffic units.

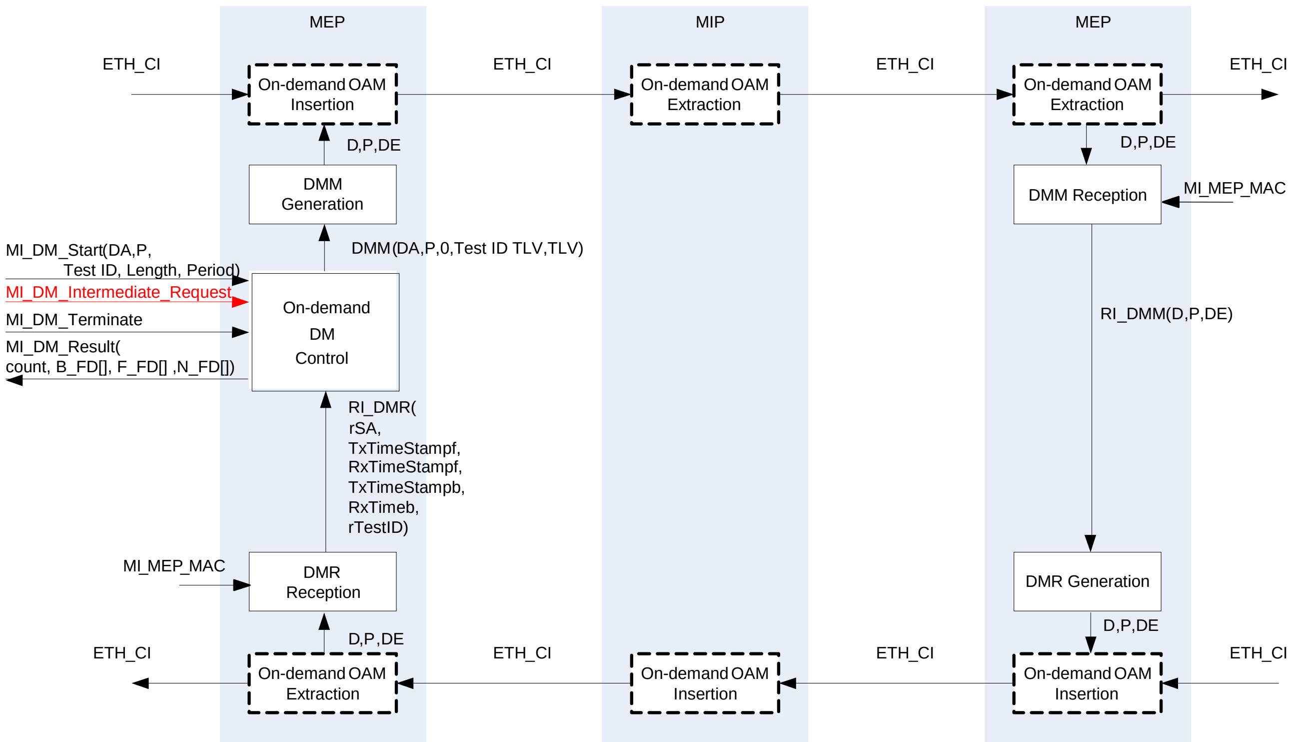

Figure 8-47 – Overview of processes involved with on-demand delay measurement

The on-demand DM control process controls the on-demand DM protocol. The protocol is activated upon receipt of the MI_DM_Start(DA,P,Test ID,Length,Period) signal and remains activated until the MI_DM_Terminate signal is received. The result is communicated via the MI_DM_Result(count, B_FD[], F_FD[] ,N_FD[]) signal when the process is terminated by the MI_DM_Terminate signal or when an intermediate result is requested via the MI_DM_Intermediate_Request signal. If the on-demand DM control process activates the multiple monitoring on different CoS levels simultaneously, each result is independently managed per CoS level. Optional test ID TLV can be utilized to distinguish each measurement if multiple measurements are simultaneously activated in an ME. If the protocol is used in multipoint-to-multipoint environments, the multicast class 1 address can be used for a DA and the test result is independently managed per peer node.

The DMM generation process generates DMM traffic units that pass through MIPs transparently, but are received and processed by DMM reception processes in MEPs. The DMR generation process may generate a DMR traffic unit in response. This DMR traffic unit also passes transparently through MIPs, but is received and processed by DMR reception processes in MEPs.

At the source MEP side, the DMM generation process stamps the value of the local time to the TxTimeStampf field in the DMM message when the first bit of the frame is transmitted. Note well that at the sink MEP side, the DMM reception process stamps the value of the local time to the RxTimeStampf field in the DMM message when the last bit of the frame is received.

The DMR generation and reception process stamps with the same way as the DMM generation and reception process.

…

2.2) Clause 8.1.10.2

Update clause, 8.1.10.2, DM control process, as indicated below:

The behaviour of the on-demand DM control process is defined in Figure 8-49.

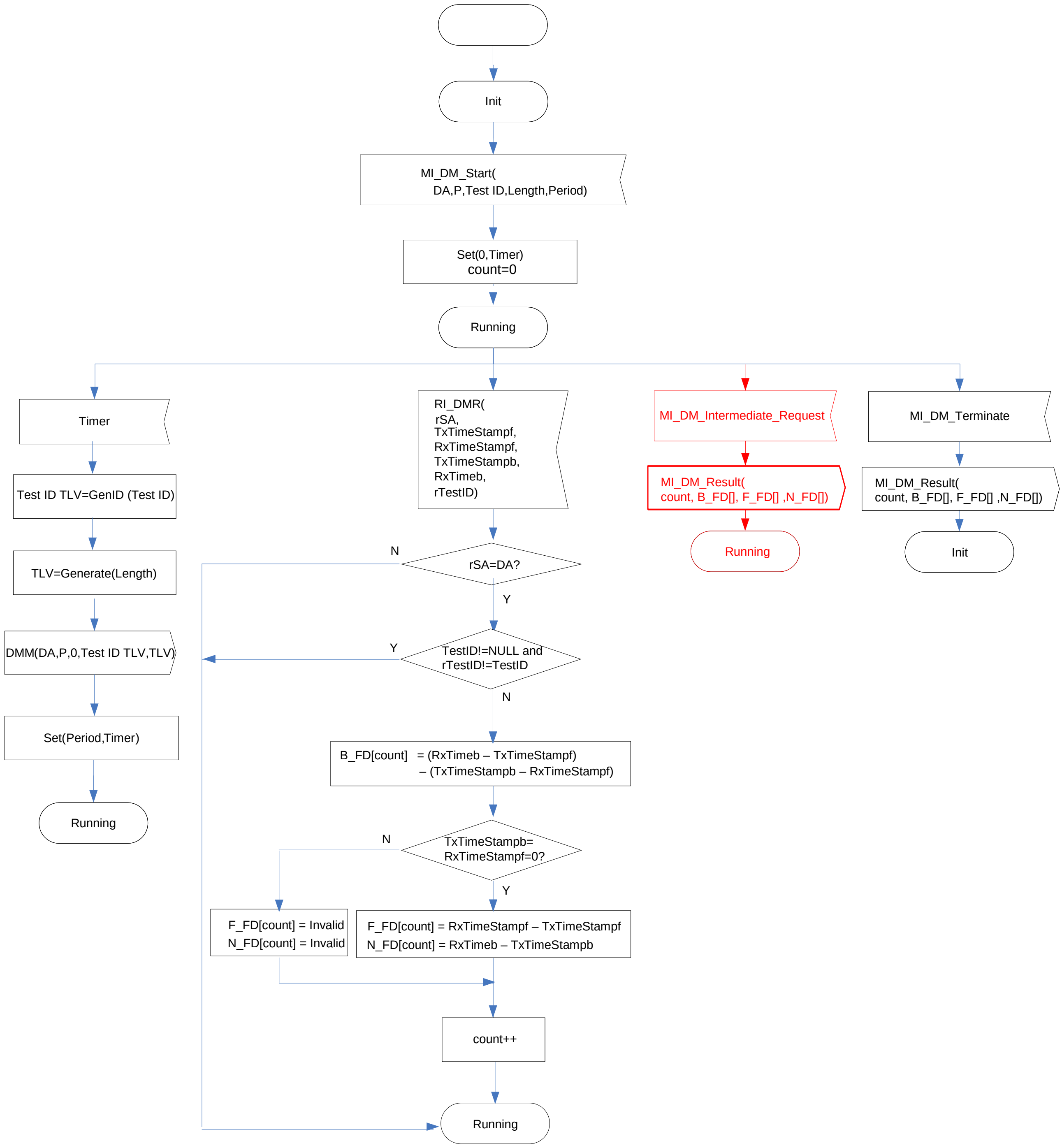

Figure 8-49 – On-demand DM control behaviour

Upon receipt of the MI_DM_Start(DA,P,Test ID,Length,Period), the DM protocol is started. Every period the generation of a DMM frame is triggered (using the DMM(DA,P,0,Test ID TLV,TLV) signal), until the MI_DM_Terminate signal is received. The TLV field of the DMM frames can have two types of TLVs. The first one is the test ID TLV, which is optionally used for a discriminator of each test and the value Test ID is included in the TLV. The second one is the data TLV, which is determined by the Generate(Length) function. Generate(Length) generates a Data TLV with length "'Length" of an arbitrary bit pattern to be included in the DMM frame.

Upon receipt of a DMR traffic unit the delay value recorded by this particular DMR traffic unit is calculated. This result is reported using the MI_DM_Result(count, B_FD[], F_FD[] ,N_FD[]) signal after the receipt of the MI_DM_Terminate signal or of the MI_DM_Intermediate_Request signal.

Note that the measurements of F_FD and N_FD are not supported by the peer MEP if both TxTimeStampb and TxTimeStampf are zero.

3) Clause 8.1.11, One-way delay measurement (1DM) processes

3.1) Clause 8.1.11.1

Update clause 8.1.11.1, Overview, as follows:

Figure 8-57 shows the different processes inside MEPs and MIPs that are involved in the on-demand one-way delay measurement protocol.

The MEP on-demand OAM source insertion process is defined in clause 9.4.1.1, the MEP on-demand OAM sink extraction process in clause 9.4.1.2, and the MIP on-demand OAM sink extraction process in clause 9.4.2.2. In summary, they insert and extract ETH_CI OAM signals into and from the stream of ETH_CI_D traffic units and the complementing P and DE signals going through a MEP and MIP; the extraction is based on MEL and Opcode. Furthermore, the insertion process inserts the correct MEL and SA values into the OAM traffic units.

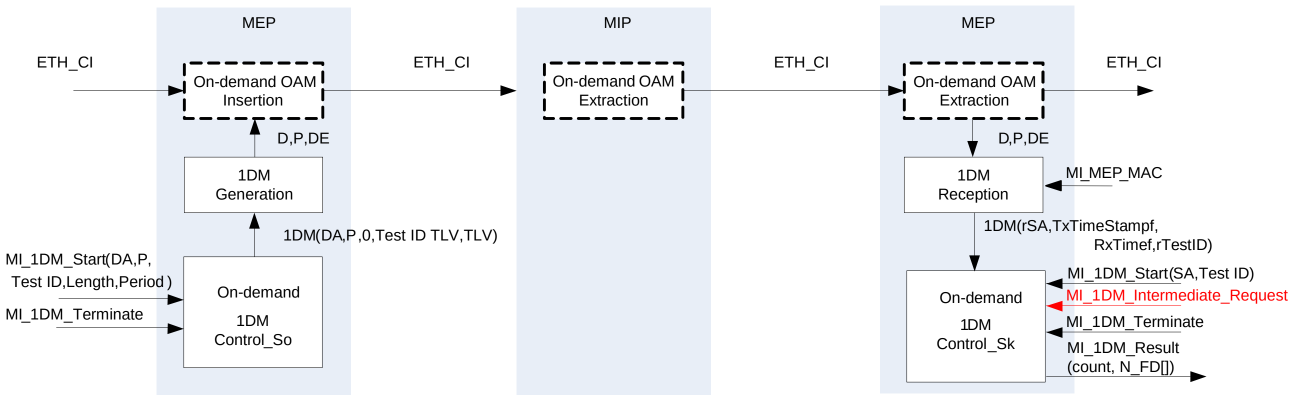

Figure 8-57 – Overview of processes involved with on-demand one-way delay measurement

The on-demand 1DM protocol is controlled by the on-demand 1DM Control_So and 1DM Control_Sk processes. The on-demand 1DM Control_So process triggers the generation of 1DM traffic units upon the receipt of an MI_1DM_Start(DA,P,Test ID,Length,Period) signal. The on-demand 1DM Control_Sk process processes the information from received 1DM traffic units after receiving the MI_1DM_Start(SA,Test ID) signal. The result is communicated by the sink MEP when the on-demand 1DM control_Sk process is terminated by the MI_1DM_terminate signal or when an intermediate result is requested via the MI_1DM_intermediate_request signal.

The 1DM generation process generates 1DM messages that pass transparently through MIPs and are received and processed by the 1DM reception process in MEPs.

At the source MEP side, the 1DM generation process stamps the value of the local time to the TxTimeStampf field in the 1DM message when the first bit of the frame is transmitted. Note well that at the sink MEP side, the 1DM reception process records the value of the local time when the last bit of the frame is received.

…

3.2) Clause 8.1.11.5

Update clause 8.1.11.5, 1DM Control_Sk process, as follows:

Figure 8-64 shows the behaviour of the on-demand 1DM Control_Sk process. The MI_1DM_Start(SA) signal starts the processing of 1DM messages coming from an MEP with SA as the MAC address. The protocol runs until the receipt of the MI_1DM_Terminate signal.

While running, the process processes the received 1DM(rSA,TxTimeStampf,RxTimef,rTestID) information. First the rSA is compared with the SA from the MI_1DM_Start (SA) signal. If the rSA is not equal to this SA, the information is ignored. Next the rTestID is compared with the TestID from the MI_1DM_Start (Test ID) signal. If the MI_1DM_Start (Test ID) signal is configured and rTestID is available but both values are different, the information is ignored. Otherwise the delay from the single received 1DM traffic unit is calculated. This result is reported using the MI_1DM_Result(count, N_FD[]) signal after the receipt of the MI_1DM_Terminate signal or of the MI_1DM_Intermediate_Request signal.

Figure 8-64 – On-demand 1DM Control_Sk process

4) Clause 8.1.14, Synthetic loss measurement (SL) processes

4.1) Clause 8.1.14.1

Update clause 8.1.14.1, Overview, as follows:

Figure 8-81 shows the different processes inside MEPs and MIPs that are involved in the on-demand synthetic loss measurement protocol.

The MEP on-demand OAM insertion process is defined in clause 9.4.1.1, the MEP OAM on-demand extraction process in clause 9.4.1.2, the MIP OAM extraction process in clause 9.4.2.1, and the MIP OAM insertion process in clause 9.4.2.2. In summary, they insert and extract ETH_CI OAM signals into and from the stream of ETH_CI_D traffic units and the complementing P and D signals going through a MEP and MIP; the extraction is based on MEL and Opcode. Furthermore, the insertion process inserts the correct MEL and SA values into the OAM traffic units.

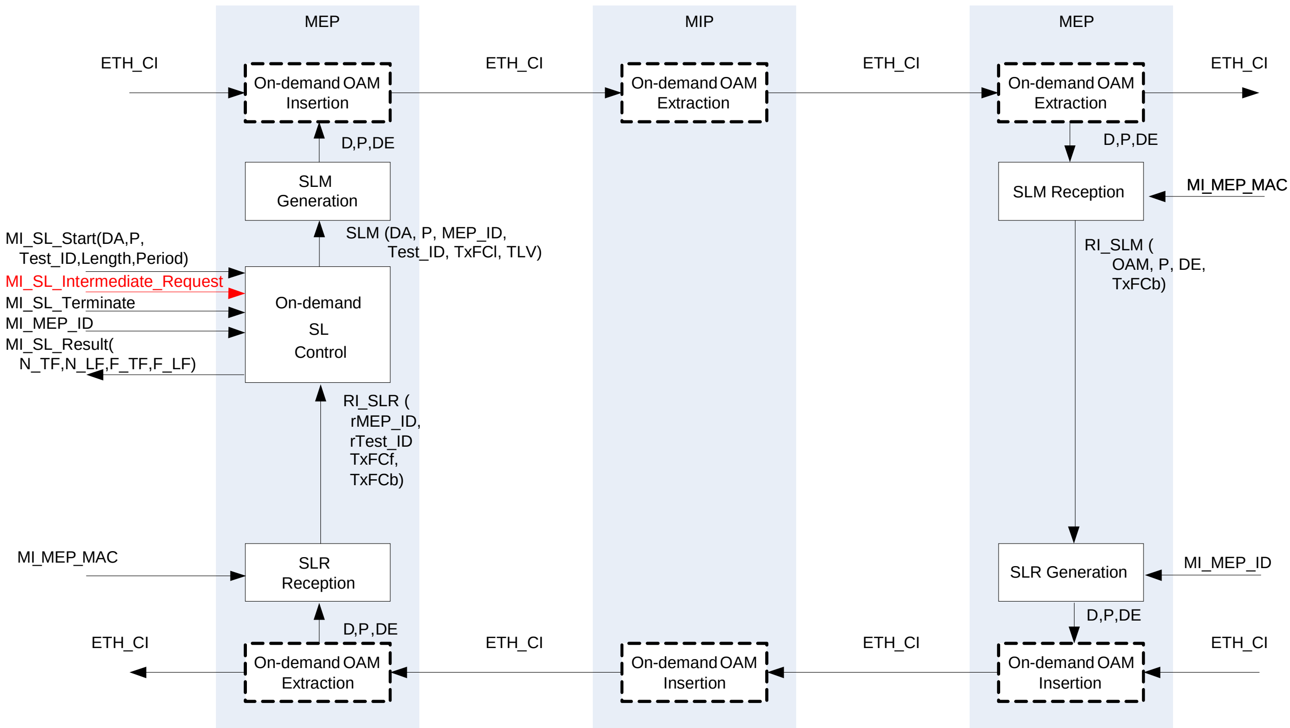

Figure 8-81 – Overview of processes involved with on-demand synthetic loss measurement protocol

The SL protocol is controlled by the on-demand SL control process.

The on-demand SL control process is activated upon receipt of the MI_SL_Start(DA,P,Test_ID,Length,Period) signal and remains activated until the MI_SL_Terminate signal is received. The measured synthetic loss values are output ~~after the MI_SL_Terminate signal~~ via the MI_SL_Result(N_TF,N_LF,F_TF,F_LF) signal when the process is terminated by the MI_SL_Terminate signal or when an intermediate result is requested via the MI_SL_Intermediate_Request signal.

The SLM generation process generates SLM traffic units that pass through MIPs transparently, but are received and processed by SLM reception processes in MEPs. The SLR generation process may generate an SLR traffic unit in response. This SLR traffic unit also passes transparently through MIPs, but is received and processed by SLR reception processes in MEPs.

…

4.2) Clause 8.1.14.2

Update clause 8.1.14.2, SL control process, as follows:

The behaviour of the on-demand SL control process is defined in Figure 8-83. There are multiple instances of the on-demand SL control process, each handling an independent stream of SLM frames.

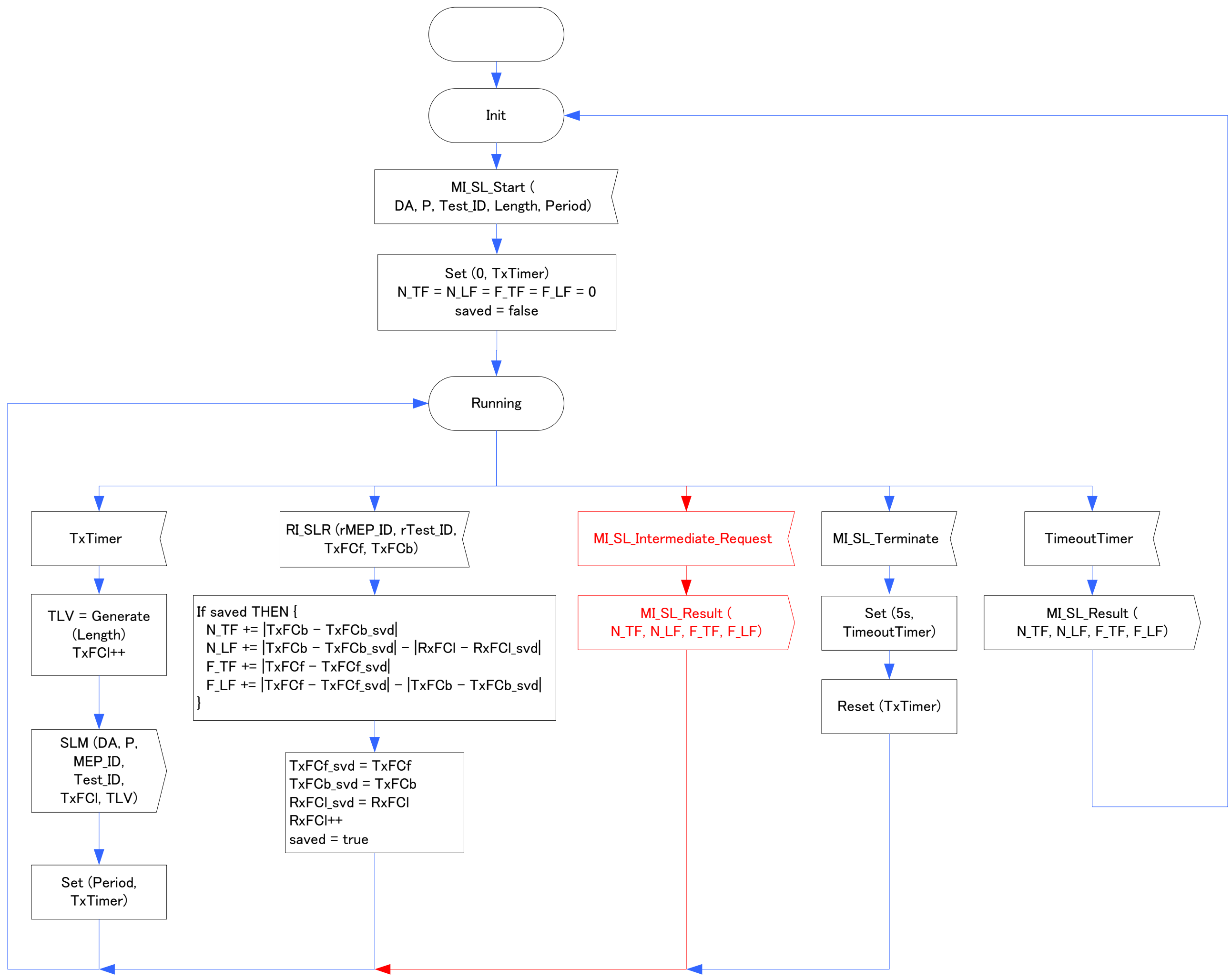

Figure 8-83 – On-demand SL control behaviour

Upon receipt of the MI_SL_Start(DA,P,Test ID,Length,Period), the SL protocol is started. Every designated period the generation of an SLM frame is triggered (using the SLM(DA,P,MEP_ID,Test_ID,TxFCl,TLV) signal), until the MI_SL_Terminate signal is received. The MEP_ID is the MI_MEP_ID of the MEP itself. The TLV field of the SLM frames is determined by the Generate(Length) function. Generate(Length) generates a Data TLV with length 'Length' of arbitrary bit pattern, as described in clause 8.1.8.2. If the length is 0, the TLV is set to NULL.

Upon receipt of an SLR traffic unit, the received counter values are used to count the near-end and far-end transmitted and lost synthetic frames. This result is reported using the MI_SL_Result(N_TF,N_LF,F_TF,F_LF) signal after the receipt of the MI_SL_Terminate signal or of the MI_SL_Intermediate_Request signal.

5) Clause 8.1.15, One-way synthetic loss measurement (1SL) processes

5.1) Clause 8.1.15.1

Update clause 8.1.15.1, Overview, as follows:

Figure 8-91 shows the different processes inside MEPs and MIPs that are involved in the on-demand one way synthetic loss measurement protocol.

The MEP on-demand OAM source insertion process is defined in clause 9.4.1.1, the MEP on-demand OAM sink extraction process in clause 9.4.1.2, the MIP on-demand OAM sink extraction process in clause 9.4.2.2. In summary, they insert and extract ETH_CI OAM signals into and from the stream of ETH_CI_D traffic units and the complementing P and DE signals going through a MEP and MIP; the extraction is based on MEL and Opcode. Furthermore, the insertion process inserts the correct MEL and SA values into the OAM traffic units.

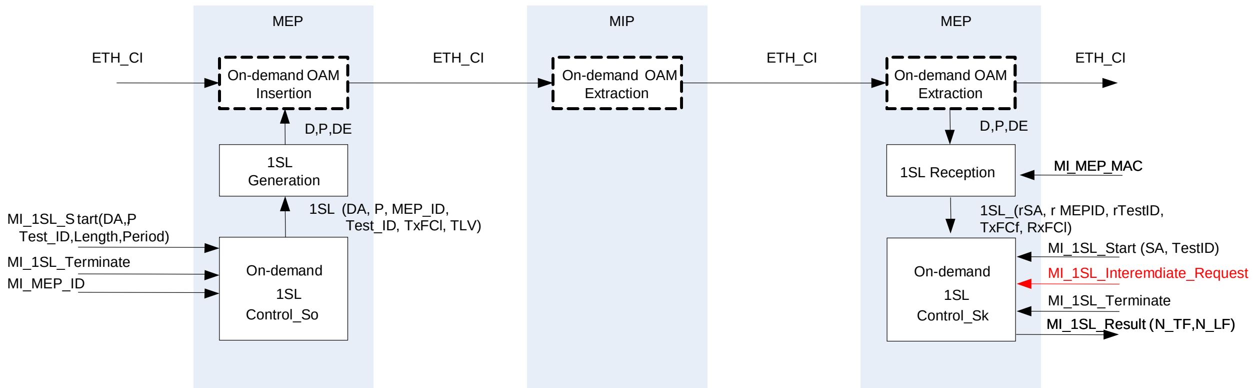

Figure 8-91 – Overview of processes involved with on-demand one-way synthetic loss measurement

The on-demand 1SL protocol is controlled by the on-demand 1SL Control_So and 1SL Control_Sk processes. The on-demand 1SL Control_So process triggers the generation of 1SL traffic units upon the receipt of an MI_1SL_Start(DA,P, Test ID,Length,Period) signal. The on-demand 1SL Control_Sk process processes the information from received 1SL traffic units after receiving the MI_1SL_Start(SA,Test ID) signal. The result is communicated by the sink MEP when the process is terminated by the MI_1SL_Terminate signal or when an intermediate result is requested via the MI_1SL_Intermediate_Request signal.

The 1SL generation process generates 1SL messages that pass transparently through MIPs and are received and processed by the 1SL reception process in MEPs.

…

5.2) Clause 8.1.15.5

Update clause 8.1.15.5., 1SL Control_Sk process1, as follows:

Figure 8-98 shows the behaviour of the on-demand 1SL Control_Sk process. The MI_1SL_Start(SA,Test ID) signal starts the processing of 1SL messages coming from a MEP with SA as MAC address. The protocol runs until the receipt of the MI_1SL_Terminate signal.

While running the process processes the received 1SL(rSA, rMEP_ID, rTest_ID, TxTCf, RxTCl) information. First the rSA is compared with the SA from the MI_1SL_Start (SA,Test ID) signal. If the rSA is not equal to this SA, the information is ignored. Next the rTestID is compared with the TestID from the MI_1SL_Start (SA,Test_ID) signal. If the Test_ID signal is configured and rTest_ID is available but both values are different, the information is ignored. Otherwise the loss from the single received 1SL traffic unit is calculated. This result is reported using the MI_1SL_Result(N_TF, N_LF) signal after the receipt of the MI_1SL_Terminate signal or of the MI_1SL_Intermediate_Request signal.

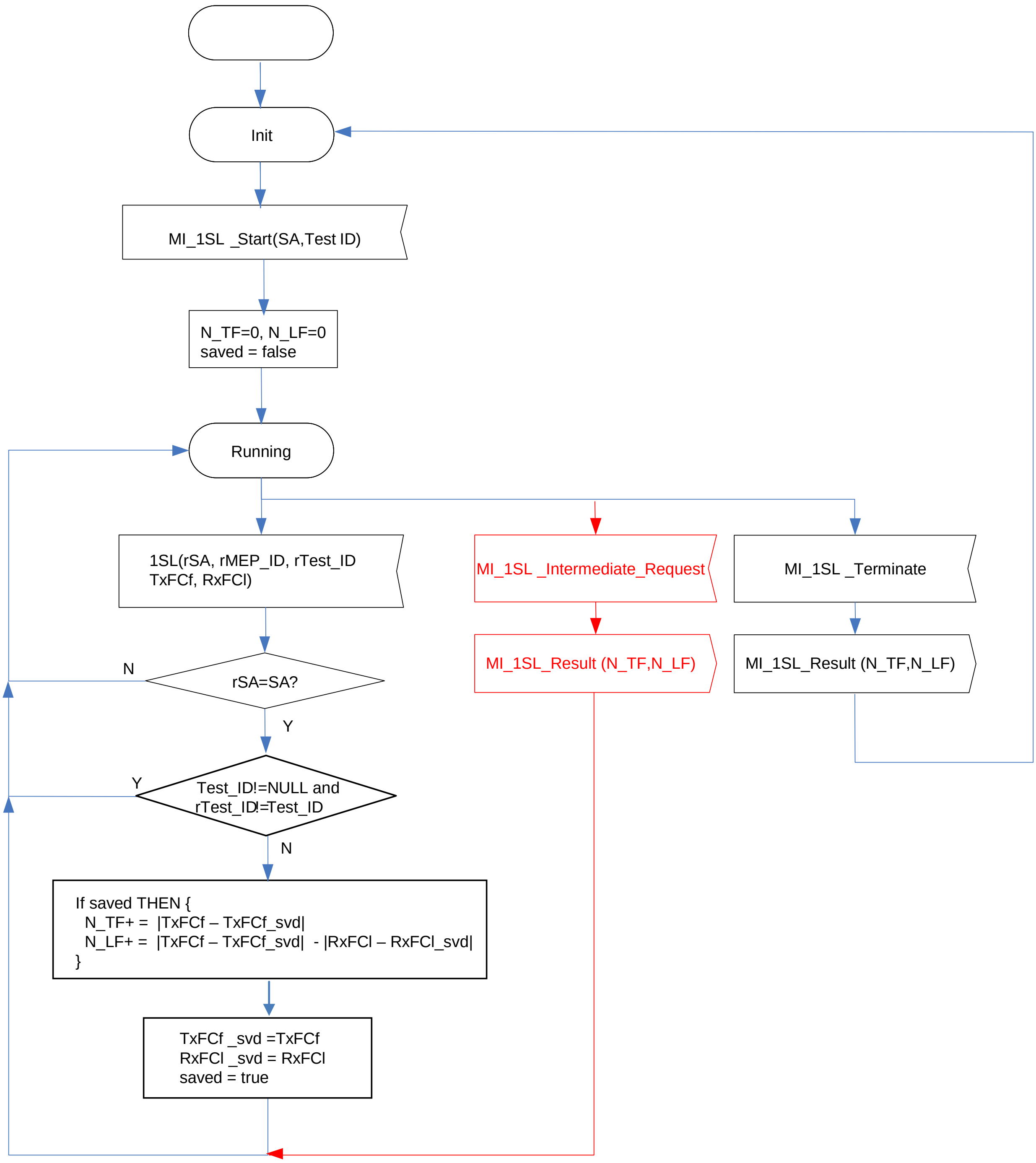

Figure 8-98 – On-demand 1SL Control_Sk process

6) Clause 8.1.16, CSF insert process

Update clause 8.1.16 as follows:

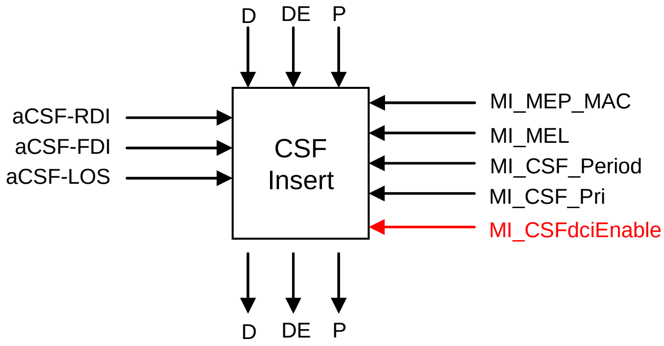

Figure 8-100 – CSF insert process

Figure 8-100 shows the CSF insert process symbol and Figure 8-101 defines the behaviour. If ~~the aCSF signal is true~~any of the aCSF-RDI, aCSF-FDI or aCSF-LOS signals are true, the CSF Insert process continuously generates ETH_CI traffic units where the ETH_CI_D signal contains the CSF ~~signal until the aCSF signal is false~~traffic unit until the condition no longer holds, i.e., all of aCSF-RDI, aCSF-FDI and aCSF-LOS are false. At this point, CSF traffic unit(s) with DCI (Defect Clear Information) are generated indicating that the defect has been cleared, if MI_CSFdciEnable = True.

NOTE 1 – Figure 8-101 shows a case where a single CSF traffic unit with DCI is generated. However, the detail transmission condition (e.g., transmission period, the number of traffic units) is out of the scope of this Recommendation.

The generated CSF traffic units are inserted in the incoming stream, i.e., the output stream contains the incoming traffic units and the generated CSF traffic units.

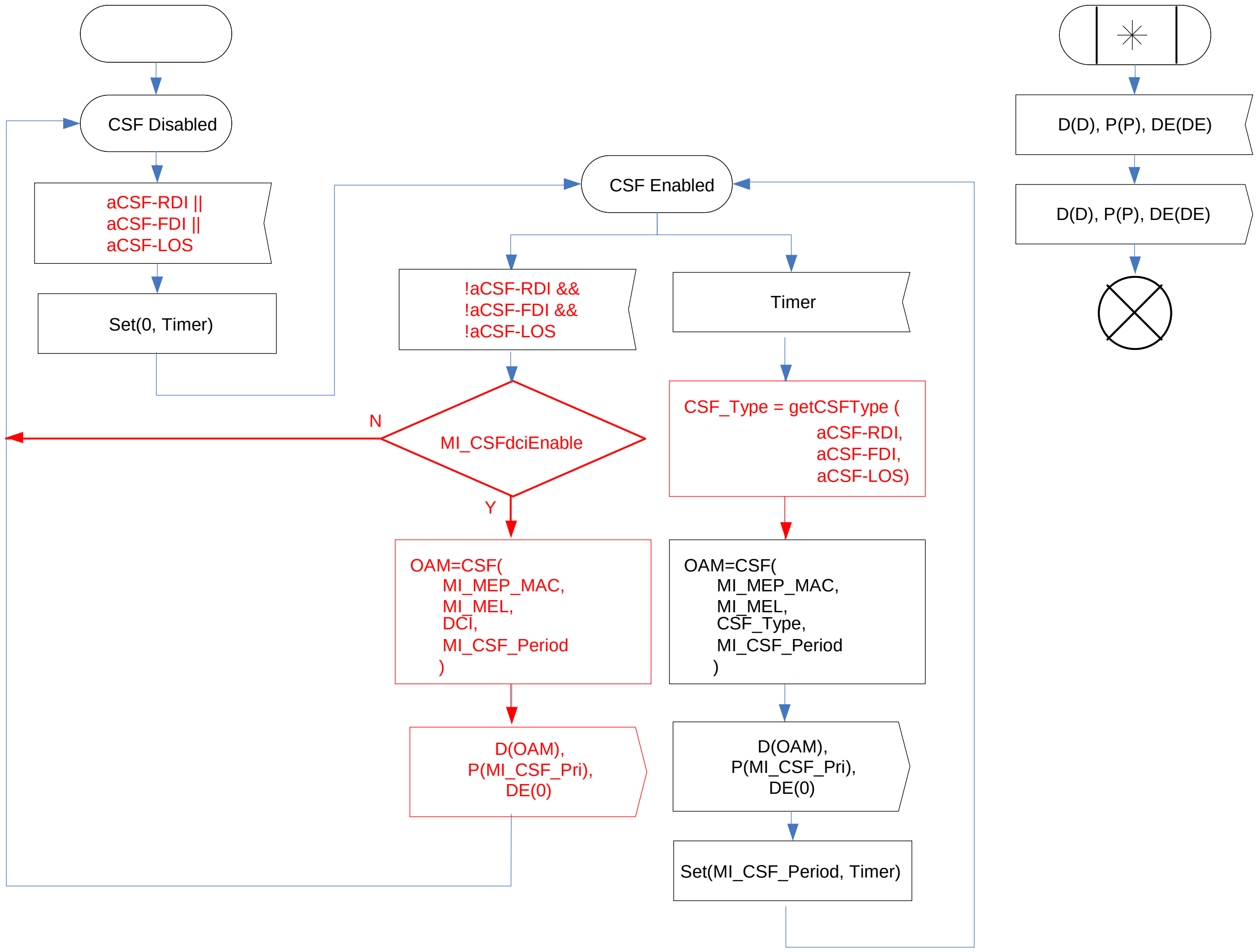

Figure 8-101 – CSF insert behaviour

If exactly one of aCSF-RDI, aCSF-FDI and aCSF-LOS are set, the getCSFType() function returns RDI, FDI or LOS as appropriate. The behaviour of getCSFType() when more than one of the conditions are set is for further study.

NOTE 2 – As described in [ITU-T Y.1731], triggering CSF is client- and application-specific. Ideally all clients and applications should ensure that at most one of the conditions is set at any given time.

7) Clause 9, Ethernet MAC layer (ETH) functions

7.1) Clause 9.2.1.1

Update clause 9.2.1.1, ETHx flow termination source function (ETHx_FT_So), as indicated below:

…

Interfaces

Table 9-2 – ETHx_FT_So interfaces

Inputs	Outputs
ETH_AP: ETH_AI_D ETH_AI_P ETH_AI_DE **ETH_RP:** ETH_RI_CC_RxFCl ETH_RI_CC_TxFCf ETH_RI_CC_RDI ETH_RI_CC_Blk ETH_RI_LMM(OAM,P,DE) ETH_RI_LMR(rSA,TxFCf,RxFCf,TxFCb,RxFCl) [1...M_{LM}] ETH_RI_DMM(OAM,P,DE) ETH_RI_DMR(rSA,TxTimeStampf, RxTimeStampf,TxTimeStampb,RxTimeb, rTestID)[1...M_{DM}] ETH_RI_SLM(OAM,P,DE,TxFCb) ETH_RI_SLR(rMEP_ID,rTest_ID, TxFCf, TxFCb) [1...M_{SL}] **ETH_TP**: ETHx_FT_So_TI_TimeStampl **ETHx_FT_So_MP:** ETHx_FT_So_MI_MEL ETHx_FT_So_MI_MEP_MAC ETHx_FT_So_MI_CC_Enable ETHx_FT_So_MI_LMC_Enable ETHx_FT_So_MI_MEG_ID ETHx_FT_So_MI_MEP_ID ETHx_FT_So_MI_CC_Period ETHx_FT_So_MI_CC_Pri ETHx_FT_So_MI_LML_Enable[1...M_{LM}] ETHx_FT_So_MI_LM_MAC_DA[1...M_{LM}] ETHx_FT_So_MI_LM_Period[1...M_{LM}] ETHx_FT_So_MI_LM_Pri[1...M_{LM}] ETHx_FT_So_MI_DM_Enable[1...M_{DM}] ETHx_FT_So_MI_DM_MAC_DA[1...M_{DM}] ETHx_FT_So_MI_DM_Test_ID[1...M_{DM}] ETHx_FT_So_MI_DM_Length[1...M_{DM}] ETHx_FT_So_MI_DM_Period[1...M_{DM}] ETHx_FT_So_MI_DM_Pri[1...M_{DM}] ETHx_FT_So_MI_1DM_Enable[1...M_{1DM}] ETHx_FT_So_MI_1DM_MAC_DA[1...M_{1DM}] ETHx_FT_So_MI_1DM_Test_ID[1...M_{1DM}] ETHx_FT_So_MI_1DM_Length[1...M_{1DM}] ETHx_FT_So_MI_1DM_Period[1...M_{1DM}] ETHx_FT_So_MI_1DM_Pri[1...M_{1DM}] ETHx_FT_So_MI_SL_Enable[1...M_{SL}] ETHx_FT_So_MI_SL_MAC_DA[1...M_{SL}] ETHx_FT_So_MI_SL_Test_ID[1...M_{SL}] ETHx_FT_So_MI_SL_Length[1...M_{SL}] ETHx_FT_So_MI_SL_Period[1...M_{SL}] ETHx_FT_So_MI_SL_Pri[1...M_{SL}] ETHx_FT_So_MI_1SL_Enable[1...M_{1SL}] ETHx_FT_So_MI_1SL_MAC_DA[1...M_{1SL}] ETHx_FT_So_MI_1SL_Test_ID[1...M_{1SL}] ETHx_FT_So_MI_1SL_Length[1...M_{1SL}] ETHx_FT_So_MI_1SL_Period[1...M_{1SL}] ETHx_FT_So_MI_1SL_Pri[1...M_{1SL}]	**ETH_FP:** ETH_CI_D ETH_CI_P ETH_CI_DE **ETH_RP:** ETH_RI_LM_Result(N_TF,N_LF,F_TF,F_LF) [1...M_{LM}] ETH_RI_DM_Result(B_FD,F_FD,N_FD) [1...M_{DM}] ETH_RI_SL_Result(N_TF,N_LF,F_TF,F_LF) [1...M_{SL}]

Processes

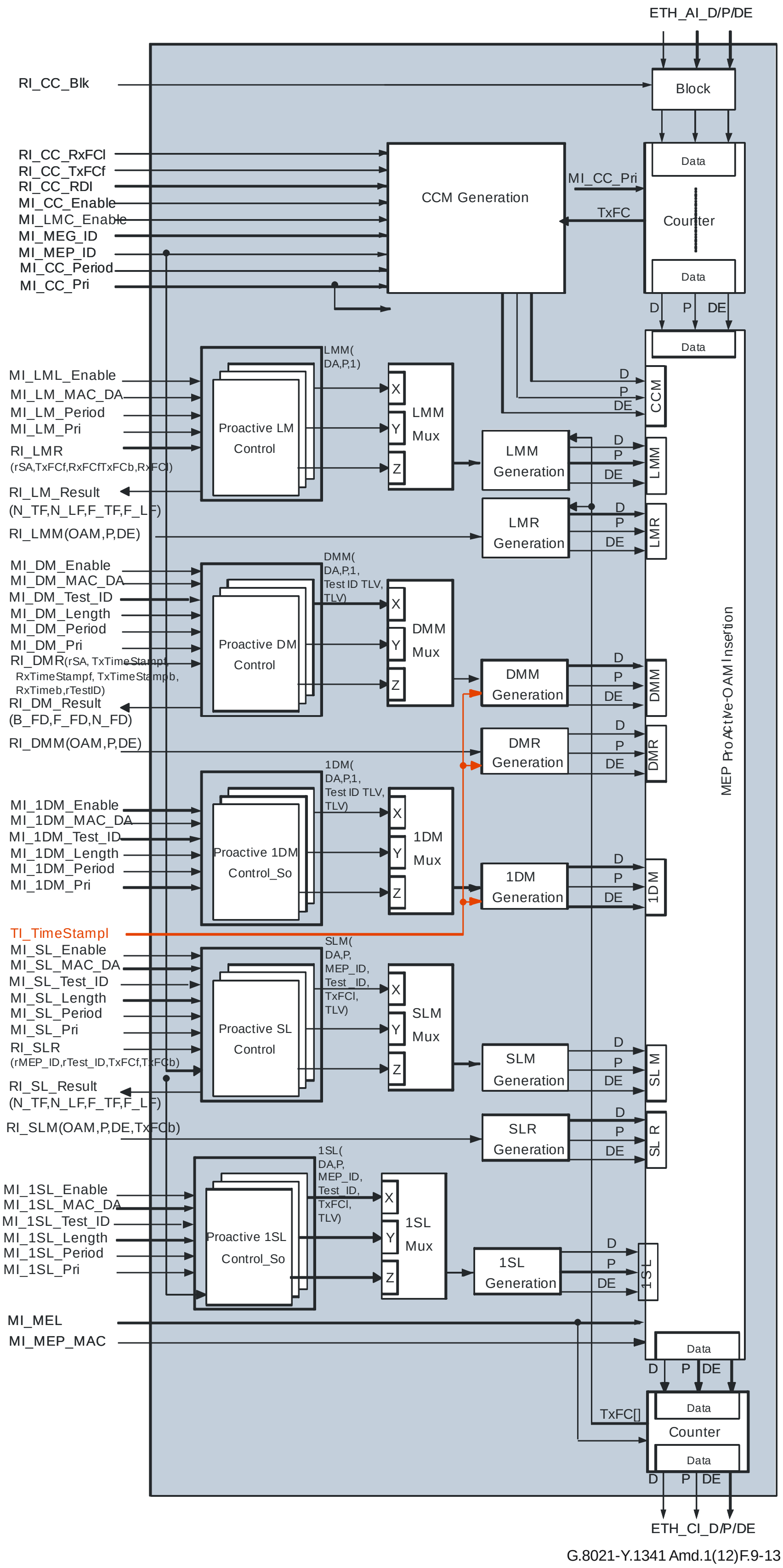

Figure 9-13 – ETHx_FT_So processes

7.2) Clause 9.2.1.2

Update clause 9.2.1.2, ETHx Flow Termination sink function (ETHx_FT_Sk), as indicated below:

…

Interfaces

Table 9-3 – ETHx_FT_Sk interfaces

Inputs	Outputs
ETH_FP: ETH_CI_D ETH_CI_P ETH_CI_DE ETH_CI_SSF **ETH_RP:** ETH_RI_LM_Result(N_TF,N_LF,F_TF,F_LF) [1...M_{LM}] ETH_RI_DM_Result(B_FD,F_FD,N_FD) [1...M_{DM}] ETH_RI_SL_Result(N_TF,N_LF,F_TF,F_LF) [1...M_{SL}] **ETH_TP:** ETHx_FT_Sk_TI_TimeStampl **ETHx_FT_Sk_MP:** ETHx_FT_Sk_MI_CC_Enable ETHx_FT_Sk_MI_LMC_Enable ETHx_FT_Sk_MI_1Second ETHx_FT_Sk_MI_LM_DEGM ETHx_FT_Sk_MI_LM_M ETHx_FT_Sk_MI_LM_DEGTHR ETHx_FT_Sk_MI_LM_TFMIN ETHx_FT_Sk_MI_MEL ETHx_FT_Sk_MI_MEG_ID ETHx_FT_Sk_MI_PeerMEP_ID[i] ETHx_FT_Sk_MI_CC_Period ETHx_FT_Sk_MI_CC_Pri ETHx_FT_Sk_MI_GetSvdCCM ETHx_FT_Sk_MI_1DM_Enable[1...M_{1DM}] ETHx_FT_Sk_MI_1DM_MAC_SA[1...M_{1DM}] ETHx_FT_Sk_MI_1DM_Test_ID[1...M_{1DM}] ETHx_FT_Sk_MI_1SL_Enable[1...M_{1SL}] ETHx_FT_Sk_MI_1SL_MAC_SA[1...M_{1SL}] ETHx_FT_Sk_MI_1SL_Test_ID[1...M_{1SL}]	**ETH_AP:** ETH_AI_D ETH_AI_P ETH_AI_DE ETH_AI_TSF ETH_AI_TSD ETH_AI_AIS **ETH_RP:** ETH_RI_CC_RxFCl ETH_RI_CC_TxFCf ETH_RI_CC_RDI ETH_RI_CC_Blk ETH_RI_LMM(OAM,P,DE) ETH_RI_LMR(rSA,TxFCf,RxFCf,TxFCb,RxFCl) [1...M_{LM}] ETH_RI_DMM(OAM,P,DE) ETH_RI_DMR(rSA,TxTimeStampf, RxTimeStampf,TxTimeStampb,RxTimeb, rTestID) [1...M_{DM}] ETH_RI_SLM(OAM,P,DE,TxFCb) ETH_RI_SLR(rMEP_ID,rTest_ID,TxFCf, TxFCb) [1...M_{SL}] **ETHx_FT_Sk_MP:** ETHx_FT_Sk_MI_cLOC[i] ETHx_FT_Sk_MI_cUNL ETHx_FT_Sk_MI_cMMG ETHx_FT_Sk_MI_cUNM ETHx_FT_Sk_MI_cDEG ETHx_FT_Sk_MI_cUNP ETHx_FT_Sk_MI_cUNPr ETHx_FT_Sk_MI_cRDI ETHx_FT_Sk_MI_cSSF ETHx_FT_Sk_MI_cLCK ETHx_FT_Sk_MI_pN_TF ETHx_FT_Sk_MI_pN_LF ETHx_FT_Sk_MI_pF_TF ETHx_FT_Sk_MI_pF_LF ETHx_FT_Sk_MI_pF_DS ETHx_FT_Sk_MI_pN_DS ETHx_FT_Sk_MI_pB_FD ETHx_FT_Sk_MI_pB_FDV ETHx_FT_Sk_MI_pF_FD ETHx_FT_Sk_MI_pF_FDV ETHx_FT_Sk_MI_pN_FD ETHx_FT_Sk_MI_pN_FDV ETHx_FT_Sk_MI_SvdCCM

NOTE – If the delay measurement message rate is smaller than one second, there will be more than one set of primitive values (i.e., pB_FD, pB_FDV, pF_FD, pF_FDV, pN_FD, pN_FDV) for some one-second periods. If the delay measurement message rate is larger than one second, there will be no set of primitive values for some one-second periods.

Processes

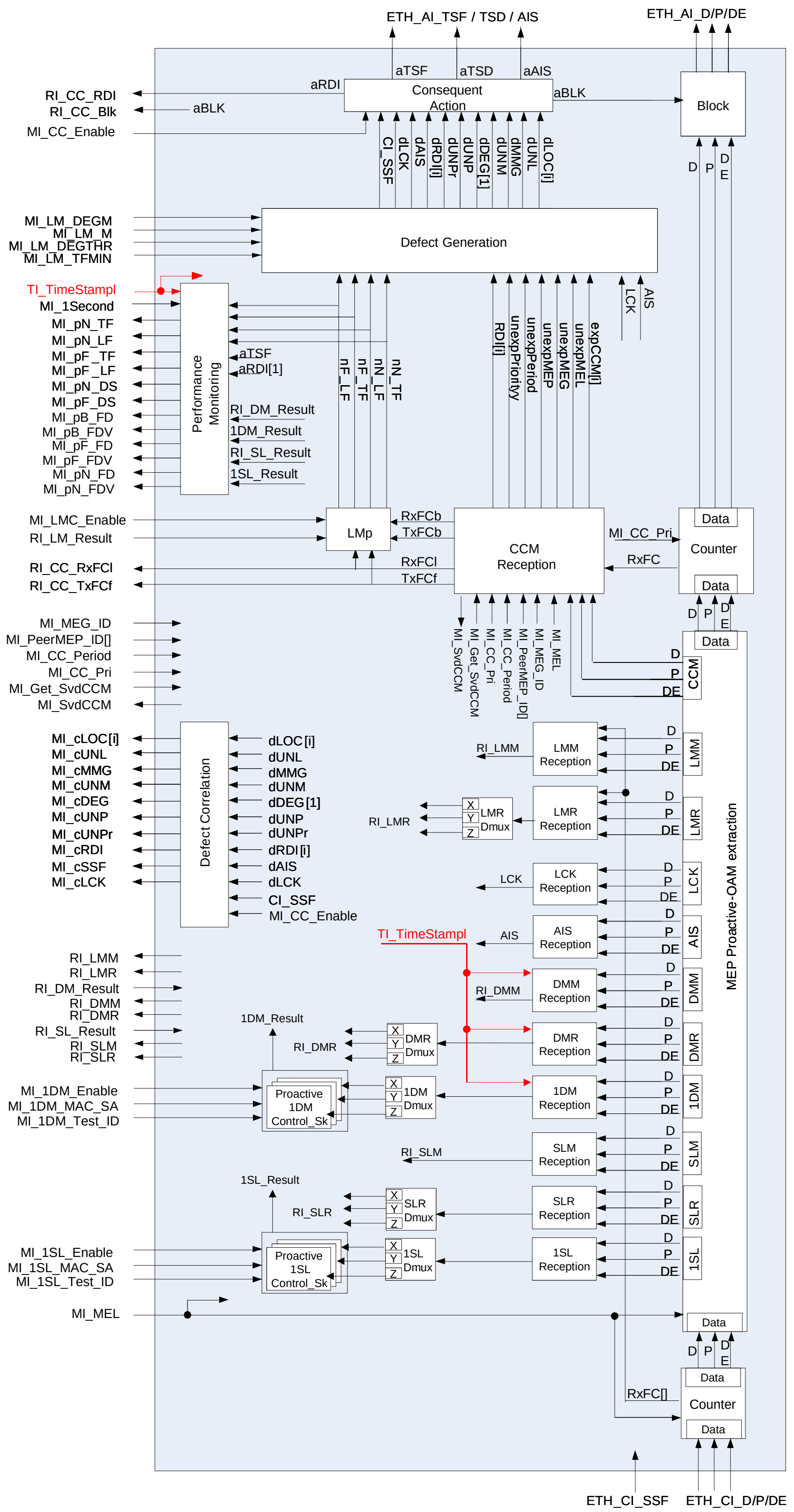

Figure 9-16 – ETHx_FT_Sk process

7.3) Clause 9.2.2.1

Update clause 9.2.2.1, ETH group flow termination source function (ETHG_FT_So), as indicated below:

…

Interfaces

Table 9-4 – ETHG_FT_So interfaces

Inputs	Outputs
ETH_AP: ETH_AI_D[1...M] ETH_AI_P[1...M] ETH_AI_DE[1...M] **ETH_RP:** ETH_RI_CC_RxFCl ETH_RI_CC_TxFCf ETH_RI_CC_RDI ETH_RI_CC_Blk ETH_RI_LMM(OAM,P,DE) ETH_RI_LMR(rSA,TxFCf,RxFCf,TxFCb,RxFCl) [1...M_{LM}] ETH_RI_DMM(OAM,P,DE) ETH_RI_DMR(rSA,TxTimeStampf, RxTimeStampf,TxTimeStampb,RxTimeb, rTestID) [1...M_{DM}] ETH_RI_SLM(OAM,P,DE,TxFCb) ETH_RI_SLR(rMEP_ID,rTest_ID, TxFCf, TxFCb) [1...M_{SL}] **ETH_TP:** ETHG_FT_So_TI_TimeStampl **ETHG_FT_So_MP:** ETHG_FT_So_MI_MEL ETHG_FT_So_MI_MEP_MAC ETHG_FT_So_MI_CC_Enable ETHG_FT_So_MI_LMC_Enable ETHG_FT_So_MI_MEG_ID ETHG_FT_So_MI_MEP_ID ETHG_FT_So_MI_CC_Period ETHG_FT_So_MI_CC_Pri ETHG_FT_So_MI_LML_Enable[1...M_{LM}] ETHG_FT_So_MI_LM_MAC_DA[1...M_{LM}] ETHG_FT_So_MI_LM_Period[1...M_{LM}] ETHG_FT_So_MI_LM_Pri [1...M_{LM}] ETHG_FT_So_MI_DM_Enable [1...M_{DM}] ETHG_FT_So_MI_DM_MAC_DA [1...M_{DM}] ETHG_FT_So_MI_DM_Test_ID [1...M_{DM}] ETHG_FT_So_MI_DM_Length [1...M_{DM}] ETHG_FT_So_MI_DM_Period [1...M_{DM}] ETHG_FT_So_MI_DM_Pri [1...M_{DM}] ETHG_FT_So_MI_1DM_Enable [1...M_{1DM}] ETHG_FT_So_MI_1DM_MAC_DA [1...M_{1DM}] ETHG_FT_So_MI_1DM_Test_ID [1...M_{1DM}] ETHG_FT_So_MI_1DM_Length [1...M_{1DM}] ETHG_FT_So_MI_1DM_Period [1...M_{1DM}] ETHG_FT_So_MI_1DM_Pri [1...M_{1DM}] ETHG_FT_So_MI_SL_Enable [1...M_{SL}] ETHG_FT_So_MI_SL_MAC_DA [1...M_{SL}] ETHG_FT_So_MI_SL_Test_ID [1...M_{SL}] ETHG_FT_So_MI_SL_Length [1...M_{SL}] ETHG_FT_So_MI_SL_Period [1...M_{SL}] ETHG_FT_So_MI_SL_Pri [1...M_{SL}] ETHG_FT_So_MI_1SL_Enable [1...M_{1SL}] ETHG_FT_So_MI_1SL_MAC_DA [1...M_{1SL}] ETHG_FT_So_MI_1SL_Test_ID [1...M_{1SL}] ETHG_FT_So_MI_1SL_Length [1...M_{1SL}] ETHG_FT_So_MI_1SL_Period [1...M_{1SL}] ETHG_FT_So_MI_1SL_Pri [1...M_{1SL}]	**ETH_FP:** ETH_CI_D[1...M] ETH_CI_P[1...M] ETH_CI_DE[1...M] **ETH_RP:** ETH_RI_LM_Result(N_TF,N_LF,F_TF,F_LF) [1...M_{LM}] ETH_RI_DM_Result(B_FD,F_FD,N_FD) [1...M_{DM}] ETH_RI_SL_Result(N_TF,N_LF,F_TF,F_LF) [1...M_{SL}]

Processes

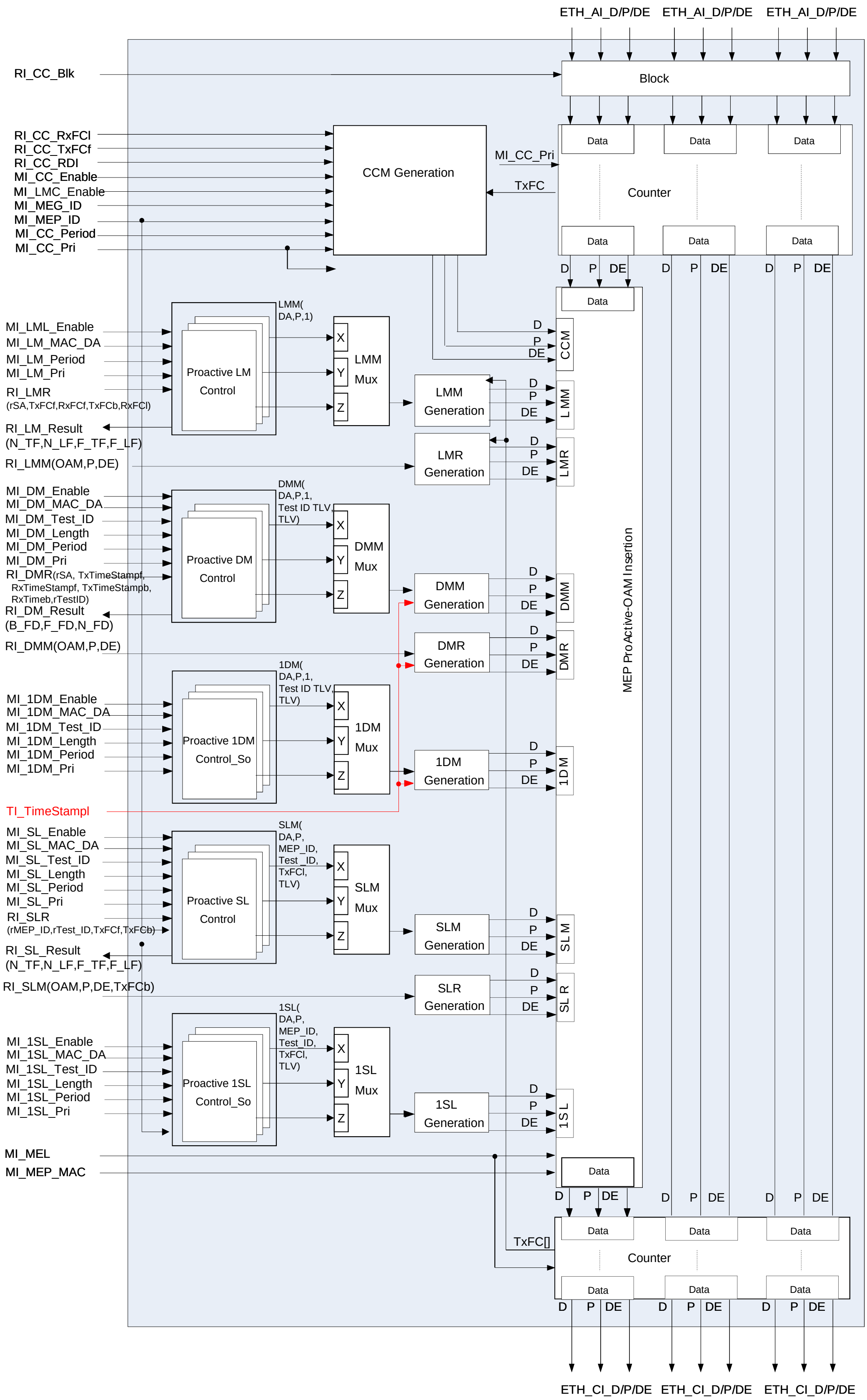

Figure 9-18 – ETHG_FT_So process

7.4) Clause 9.2.2.2

Update clause 9.2.2.2, ETH Group Flow Termination sink function (ETHG_FT_Sk), as indicated below:

…

Interfaces

Table 9-5 – ETHG_FT_Sk interfaces

Inputs	Outputs
ETH_FP: ETH_CI_D[1...M] ETH_CI_P[1...M] ETH_CI_DE[1...M] ETH_CI_SSF **ETH_RP:** ETH_RI_LM_Result(N_TF,N_LF,F_TF,F_LF) [1...M_{LM}] ETH_RI_DM_Result(B_FD,F_FD,N_FD) [1...M_{DM}] ETH_RI_SL_Result(N_TF,N_LF,F_TF,F_LF) [1...M_{SL}] **ETH_TP:** ETHG_FT_Sk_TI_TimeStampl **ETHG_FT_Sk_MP:** ETHG_FT_Sk_MI_CC_Enable ETHG_FT_Sk_MI_LMC_Enable ETHG_FT_Sk_MI_1Second ETHG_FT_Sk_MI_LM_DEGM ETHG_FT_Sk_MI_LM_M ETHG_FT_Sk_MI_LM_DEGTHR ETHG_FT_Sk_MI_LM_TFMIN ETHG_FT_Sk_MI_MEL ETHG_FT_Sk_MI_MEG_ID ETHG_FT_Sk_MI_PeerMEP_ID[i] ETHG_FT_Sk_MI_CC_Period ETHG_FT_Sk_MI_CC_Pri ETHG_FT_Sk_MI_GetSvdCCM ETHG_FT_Sk_MI_1DM_Enable [1...M_{1DM}] ETHG_FT_Sk_MI_1DM_MAC_SA [1...M_{1DM}] ETHG_FT_Sk_MI_1DM_Test_ID [1...M_{1DM}] ETHG_FT_Sk_MI_1SL_Enable [1...M_{1SL}] ETHG_FT_Sk_MI_1SL_MAC_SA [1...M_{1SL}] ETHG_FT_Sk_MI_1SL_Test_ID [1...M_{1SL}]	**ETH_AP:** ETH_AI_D[1...M] ETH_AI_P[1...M] ETH_AI_DE[1...M] ETH_AI_TSF ETH_AI_TSD ETH_AI_AIS **ETH_RP:** ETH_RI_CC_RxFCl ETH_RI_CC_TxFCf ETH_RI_CC_RDI ETH_RI_CC_Blk ETH_RI_LMM(OAM,P,DE) ETH_RI_LMR(rSA,TxFCf,RxFCf,TxFCb,RxFCl) [1...M_{LM}] ETH_RI_DMM(OAM,P,DE) ETH_RI_DMR(rSA,TxTimeStampf, RxTimeStampf,TxTimeStampb,RxTimeb, rTestID) [1...M_{DM}] ETH_RI_SLM(OAM,P,DE,TxFCb) ETH_RI_SLR(rMEP_ID,rTest_ID,TxFCf, TxFCb) [1...M_{SL}] **ETHG_FT_Sk_MP:** ETHG_FT_Sk_MI_cLOC[i] ETHG_FT_Sk_MI_cUNL ETHG_FT_Sk_MI_cMMG ETHG_FT_Sk_MI_cUNM ETHG_FT_Sk_MI_cDEG ETHG_FT_Sk_MI_cUNP ETHG_FT_Sk_MI_cUNPr ETHG_FT_Sk_MI_cRDI ETHG_FT_Sk_MI_cSSF ETHG_FT_Sk_MI_cLCK ETHG_FT_Sk_MI_pN_TF ETHG_FT_Sk_MI_pN_LF ETHG_FT_Sk_MI_pF_TF ETHG_FT_Sk_MI_pF_LF ETHG_FT_Sk_MI_pF_DS ETHG_FT_Sk_MI_pN_DS ETHG_FT_Sk_MI_pB_FD ETHG_FT_Sk_MI_pB_FDV ETHG_FT_Sk_MI_pF_FD ETHG_FT_Sk_MI_pF_FDV ETHG_FT_Sk_MI_pN_FD ETHG_FT_Sk_MI_pN_FDV ETHG_FT_Sk_MI_SvdCCM

NOTE – If the delay measurement message rate is smaller than one second, there will be more than one set of primitive values (i.e., pB_FD, pB_FDV, pF_FD, pF_FDV, pN_FD, pN_FDV) for some one-second periods. If the delay measurement message rate is larger than one second, there will be no set of primitive values for some one-second periods.

Processes

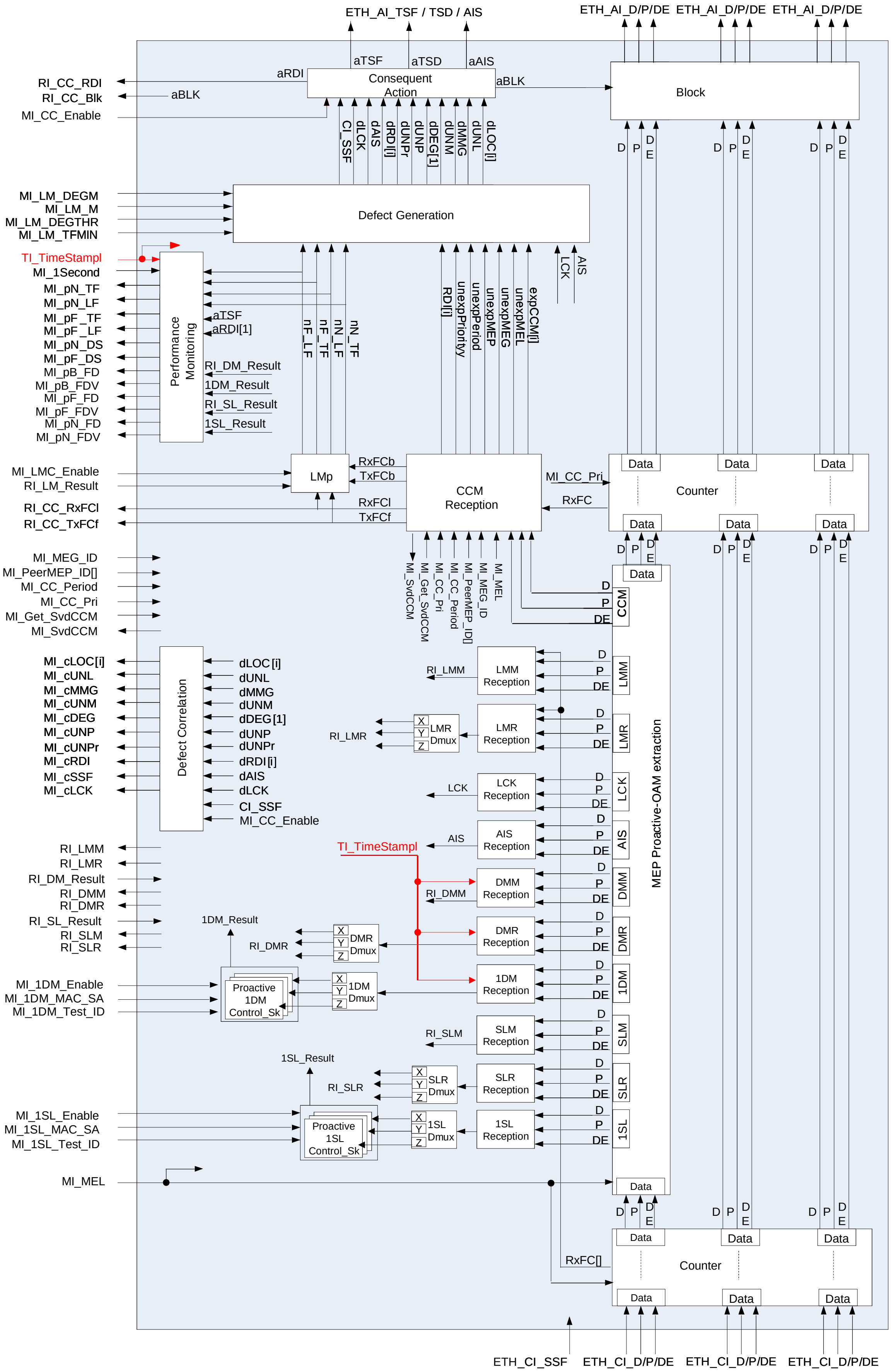

Figure 9-20 – ETHG_FT_Sk process

7.5) Clause 9.3.2.1

Update clause 9.3.2.1, ETH to ETH adaptation source function (ETHx/ETH_A_So), as indicated below:

…

Interfaces

Table 9-6 – ETHx/ETH_A_So interfaces

Inputs	Outputs
ETH_FP: ETH_CI_D ETH_CI_P ETH_CI_DE ETH_CI_APS ETH_CI_SSF ETH_CI_SSFrdi ETH_CI_SSFfdi **ETHx/ETH_A_So_MP:** ETHx/ETH_A_So_MI_Active ETHx/ETH_A_So_MI_MEP_MAC ETHx/ETH_A_So_MI_Client_MEL ETHx/ETH_A_So_MI_LCK_Period ETHx/ETH_A_So_MI_LCK_Pri ETHx/ETH_A_So_MI_Admin_State ETHx/ETH_A_So_MI_MEL ETHx/ETH_A_So_MI_APS_Pri ETHx/ETH_A_So_MI_CSF_Period ETHx/ETH_A_So_MI_CSF_Pri ETHx/ETH_A_So_MI_CSF_Enable ETHx/ETH_A_So_MI_CSFrdifdiEnable ETHx/ETH_A_So_MI_CSFdciEnable	**ETH_AP:** ETH_AI_D ETH_AI_P ETH_AI_DE

Processes

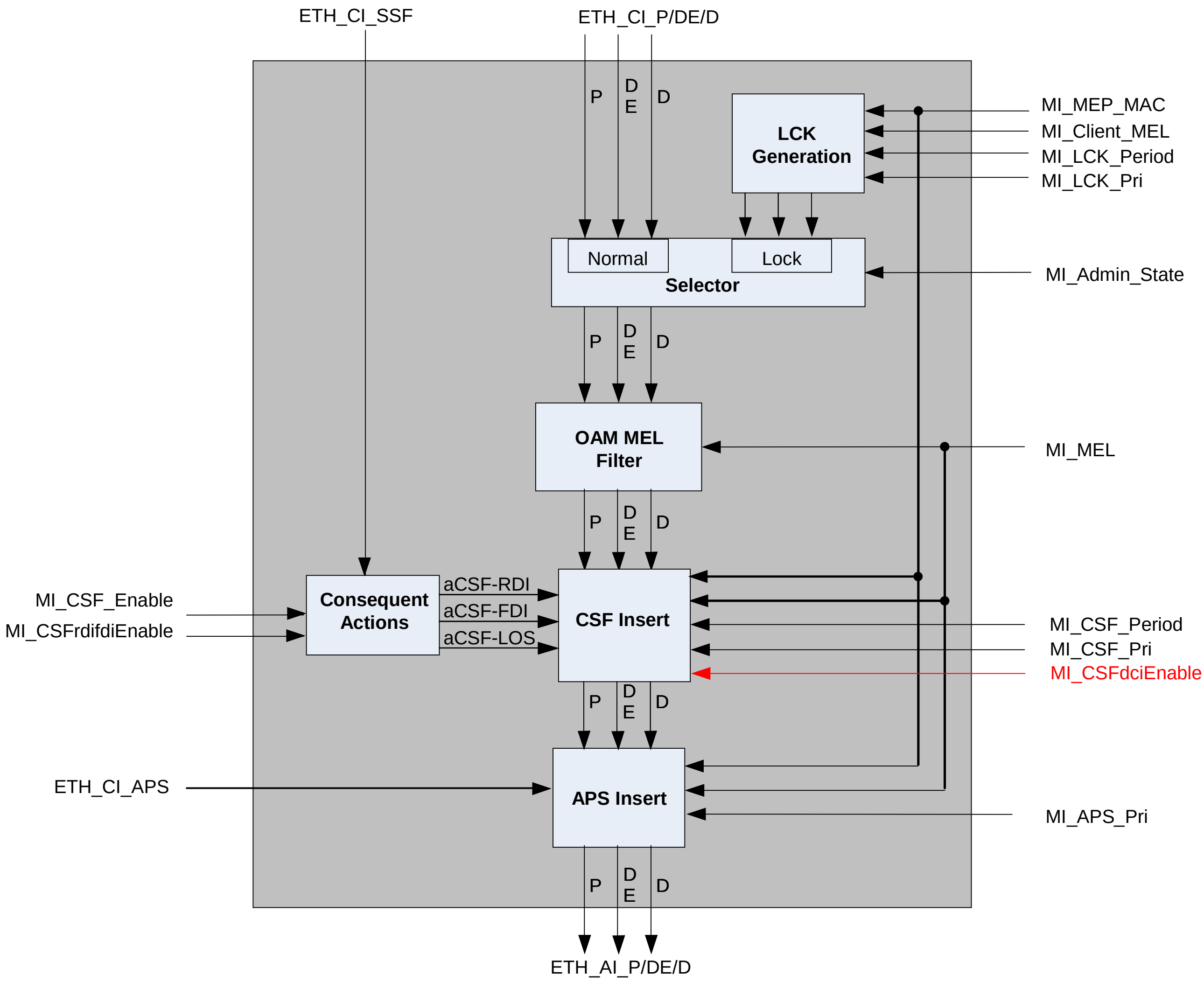

Figure 9-22 – ETHx/ETH_A_So process

7.6) Clause 9.3.3.1

Update clause 9.3.3.1, ETH to ETH multiplexing adaptation source function (ETHx/ETH-m_A_So), as indicated below:

…

Interfaces

Table 9-8 – ETHx/ETH-m_A_So interfaces

Inputs	Outputs
ETH_FP: ETH_CI_D[1...M] ETH_CI_P[1...M] ETH_CI_DE[1...M] ETH_CI_SSF[1] ETH_CI_SSFrdi[1] ETH_CI_SSFfdi[1] **ETH_TFP:** ETH_CI_D ETH_CI_P ETH_CI_DE **ETHx/ETH-m_A_So_MP:** ETHx/ETH-m_A_So_MI_Active ETHx/ETH-m_A_So_MI_MEP_MAC ETHx/ETH-m_A_So_MI_Client_MEL[1…M] ETHx/ETH-m_A_So_MI_LCK_Period[1…M] ETHx/ETH-m_A_So_MI_LCK_Pri[1…M] ETHx/ETH-m_A_So_MI_Admin_State ETHx/ETH-m_A_So_MI_VLAN_Config[1...M] ETHx/ETH-m_A_So_MI_Etype ETHx/ETH-m_A_So_MI_PCP_Config ETHx/ETH-m_A_So_MI_MEL ETHx/ETH-m_A_So_MI_CSF_Period ETHx/ETH-m_A_So_MI_CSF_Pri ETHx/ETH-m_A_So_MI_CSF_Enable ETHx/ETH-m_A_So_MI_CSFrdifdiEnable ETHx/ETH-m_A_So_MI_CSFdciEnable	**ETH_AP:** ETH_AI_D ETH_AI_P ETH_AI_DE **ETHF_PP:** ETH_PI_D ETH_PI_P ETH_PI_DE **ETHTF_PP:** ETH_PI_D ETH_PI_P ETH_PI_DE

Processes

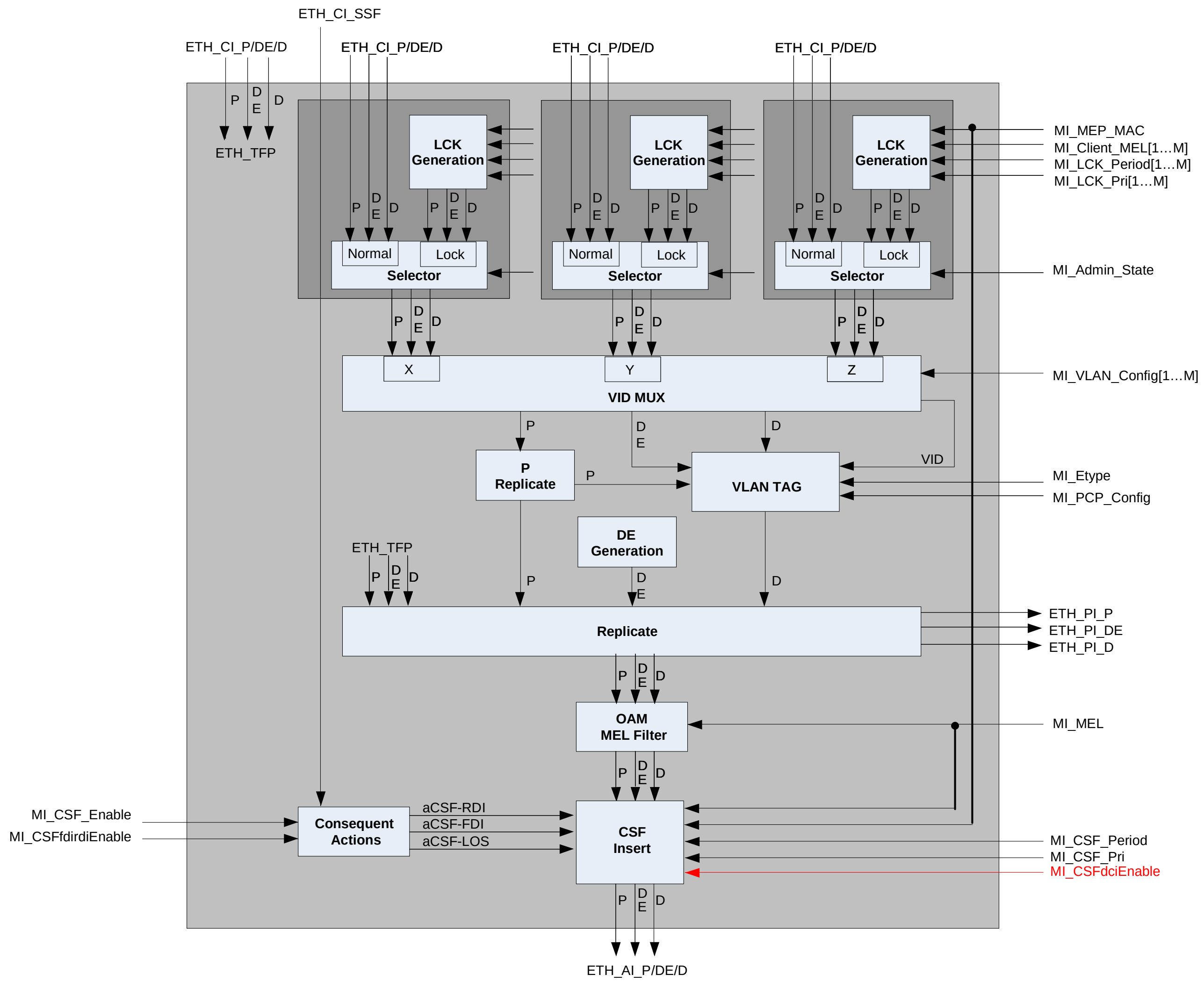

Figure 9-28 – ETHx/ETH-m_A_So process

7.7) Clause 9.3.4.1

Update clause 9.3.4.1, ETH Group to ETH adaptation source function (ETHG/ETH_A_So), as indicated below:

…

Interfaces

Table 9-10 – ETHG/ETH_A_So Interfaces

Inputs	Outputs
ETH_FP: ETH_CI_D[1…M] ETH_CI_P[1…M] ETH_CI_DE[1…M] ETH_CI_APS ETH_CI_SSF[1] ETH_CI_SSFrdi[1] ETH_CI_SSFfdi[1] **ETHG/ETH_A_So_MP:** ETHG/ETH_A_So_MI_Active ETHG/ETH_A_So_MI_MEP_MAC ETHG/ETH_A_So_MI_Client_MEL[1..M] ETHG/ETH_A_So_MI_LCK_Period[1…M] ETHG/ETH_A_So_MI_LCK_Pri[1…M] ETHG/ETH_A_So_MI_Admin_State ETHG/ETH_A_So_MI_MEL ETHG/ETH_A_So_MI_APS_Pri ETHG/ETH_A_So_MI_CSF_Period ETHG/ETH_A_So_MI_CSF_Pri ETHG/ETH_A_So_MI_CSF_Enable ETHG/ETH_A_So_MI_CSFrdifdiEnable ETHG/ETH_A_So_MI_CSFdciEnable	**ETH_AP:** ETH_AI_D[1...M] ETH_AI_P[1...M] ETH_AI_DE[1...M]

Processes

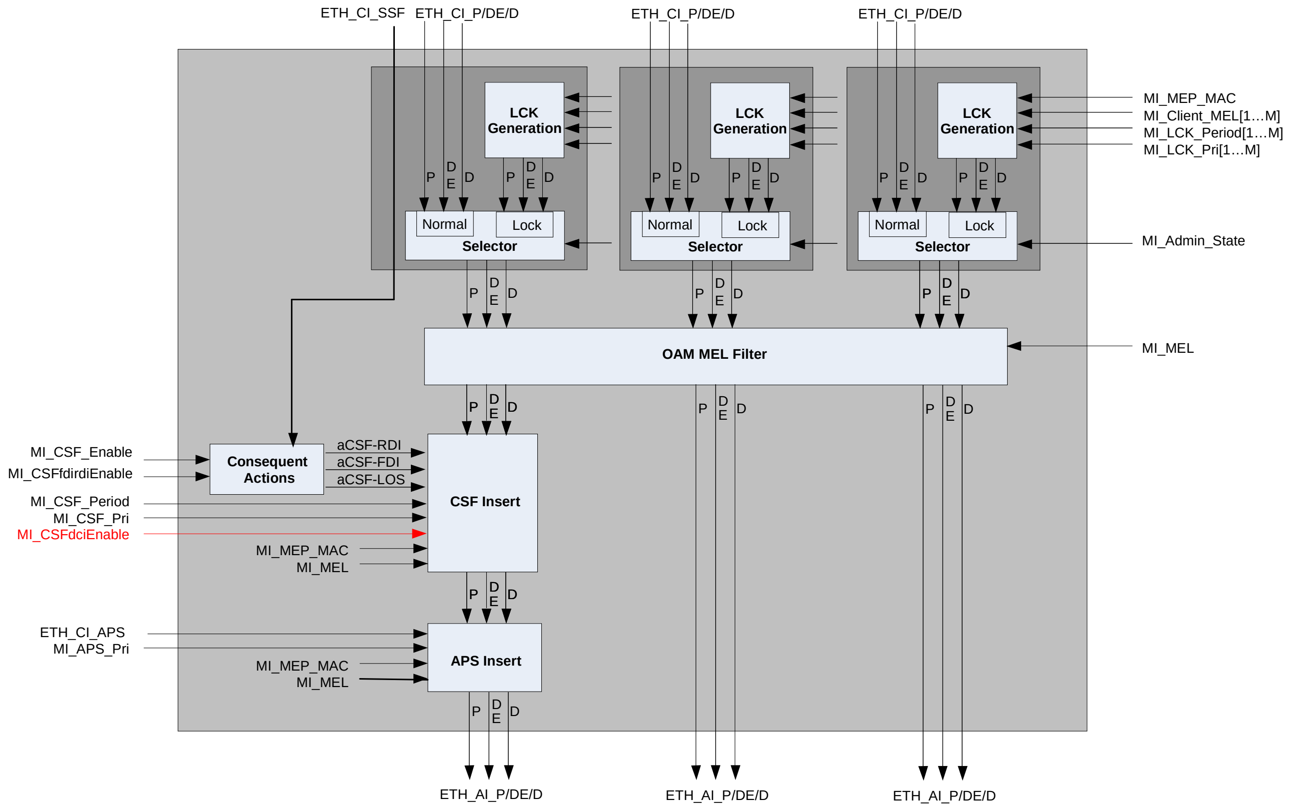

Figure 9-32 – ETHG/ETH_A_So process

7.8) Clause 9.4.1.1

Update clause 9.4.1.1, ETH diagnostic flow termination source function for MEPs (ETHDe_FT_So), as indicated below:

…

Interfaces

Table 9-14 – ETHDe_FT_So interfaces

Inputs	Outputs
ETH_AP: ETH_AI_D ETH_AI_P ETH_AI_DE **ETH_RP:** ETH_RI_LMM(D,P,DE) ETH_RI_LMR(rSA,TxFCf,RxFCf,TxFCb,RxFCl) ETH_RI_LBM(D,P,DE) ETH_RI_LBR(SA,rTLV,TID) ETH_RI_DMM(D,P,DE) ETH_RI_DMR(rSA,TxTimeStampf,RxTimeStampf, TxTimeStampb,RxTimeb,rTestID) ETH_RI_LTM(D,P,DE) ETH_RI_LTR(SA,TTL,TID,TLV) ETH_RI_SLM(OAM,P,DE,TxFCb) ETH_RI_SLR(rMEP_ID,rTest_ID,TxFCf,TxFCb) ETH_TP: ETHDe_FT_So_TI_TimeStampl **ETHDe_FT_So_MP:** ETHDe_FT_So_MI_LM_Start(DA,P,Period) ETHDe_FT_So_MI_LM_Intermediate_Request ETHDe_FT_So_MI_LM_Terminate ETHDe_FT_So_MI_LB_Discover(P) ETHDe_FT_So_MI_LB_Series(DA,DE,P,N, Length, Period) ETHDe_FT_So_MI_LB_Test (DA,DE,P,Pattern, Length, Period) ETHDe_FT_So_MI_LB_Test_Terminate ETHDe_FT_So_MI_DM_Start(DA,P,Test ID,Length,Period) ETHDe_FT_So_MI_DM_Intermediate_Request ETHDe_FT_So_MI_DM_Terminate ETHDe_FT_So_MI_1DM_Start(DA,P,Test ID,Length,Period) ETHDe_FT_So_MI_1DM_Terminate ETHDe_FT_So_MI_TST(DA,DE,P,Pattern, Length, Period) ETHDe_FT_So_MI_TST_Terminate ETHDe_FT_So_MI_LT(TA,TTL.P) ETHDe_FT_So_MI_MEP_MAC ETHDe_FT_So_MI_MEL ETHDe_FT_So_MI_MEP_ID ETHDe_FT_So_MI_SL_Start(DA,P,Test_ID,Length,Period) ETHDe_FT_So_MI_SL_Intermediate_Request ETHDe_FT_So_MI_SL_Terminate ETHDe_FT_So_MI_1SL_Start(DA,P,Test_ID,Length,Period) ETHDe_FT_So_MI_1SL_Terminate	**ETH_FP:** ETH_CI_D ETH_CI_P ETH_CI_DE **ETHDe_FT_So_MP:** ETHDe_FT_So_MI_LM_Result(N_TF, N_LF, F_TF, F_LF) ETHDe_FT_So_MI_LB_Discover_Result(MACs) ETHDe_FT_So_MI_LB_Series_Result(REC,ERR,OO) ETHDe_FT_So_MI_LB_Test_Result (Sent, REC, CRC, BER, OO) ETHDe_FT_So_MI_DM_Result(count,B_FD[],F_FD[],N_FD[]) ETHDe_FT_So_MI_TST_Result(Sent) ETHDe_FT_So_MI_LT_Results(Results) ETHDe_FT_So_MI_SL_Result(N_TF,N_LF,F_TF,F_LF)

Processes

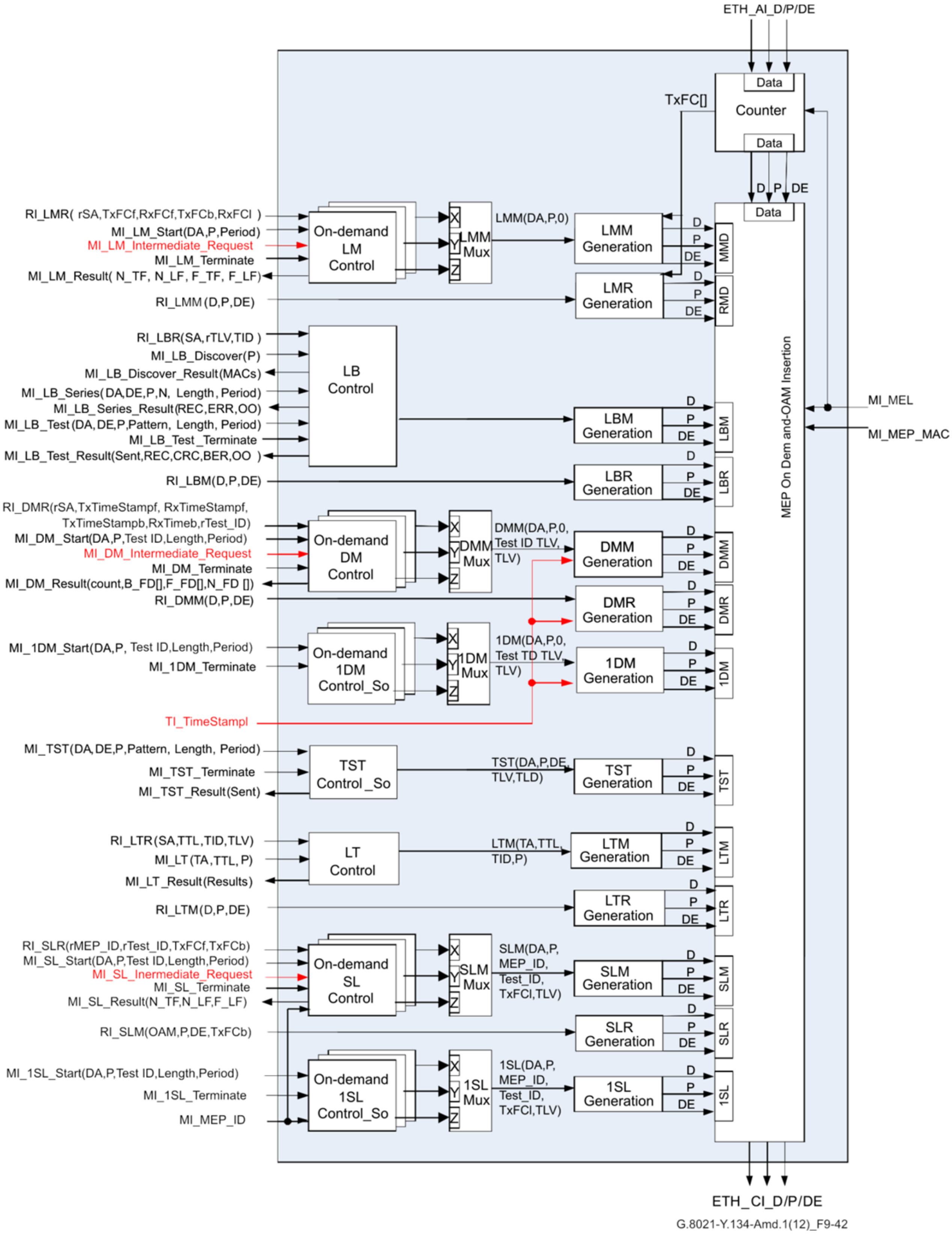

Figure 9-42 – ETHDe_FT_So process

7.9) Clause 9.4.1.2

Update clause 9.4.1.2, ETH diagnostic flow termination sink function for MEPs (ETHDe_FT_Sk), as indicated below:

…

Interfaces

Table 9-15 – ETHDe_FT_Sk interfaces

Inputs	Outputs
ETH_FP: ETH_CI_D ETH_CI_P ETH_CI_DE **ETHDe_FT_Sk_MP:** ETHDe_FT_Sk_MI_MEL ETHDe_FT_Sk_MI_MEP_MAC ETHDe_FT_Sk_MI_1DM_Start(SA,Test ID) ETHDe_FT_Sk_MI_1DM_Intermediate_Request ETHDe_FT_Sk_MI_1DM_Terminate ETHDe_FT_Sk_MI_TST_Start(SA,Pattern) ETHDe_FT_Sk_MI_1SL_Intermediate_Request ETHDe_FT_Sk_MI_TST_Terminate ETHDe_FT_Sk_MI_1SL_Start(SA,MEP ID, Test ID) ETHDe_FT_Sk_MI_1SL_Terminate ETH_TP: ETHDe_FT_Sk_TI_TimeStampl	**ETH_AP:** ETH_AI_D ETH_AI_P ETH_AI_DE **ETH _RP:** ETH_RI_LMM(D,P,DE) ETH_RI_LMR(TxFCf,RxFCb,Tx FCb,RxFCl) ETH_RI_LMR(rSA,TxFCf,RxFCf,TxFCb,RxFCl) ETH_RI_LBM(D,P,DE) ETH_RI_LBR(SA,rTLV,TID) ETH_RI_DMM(D,P,DE) ETH_RI_DMR(rSA,TxTimestampf,RxTimeStampf, TxTimeStampb,RxTimeb,rTest ID) ETH_RI_LTM(D,P,DE) ETH_RI_LTR(SA,TTL,TID,TLV) ETH_RI_SLM(OAM,P,DE,TxFCb) ETH_RI_SLR(rMEP_ID,rTest_ID,TxFCf,TxFCb) **ETHDe_FT_Sk_MP:** ETHDe_FT_Sk_MI_1DM_Result(count,N_FD[]) ETHDe_FT_Sk_MI_TST_Result(REC,CRC,BER,OO) ETHDe_FT_Sk_MI_1SL_Result(N_TF,N_LF)

Processes

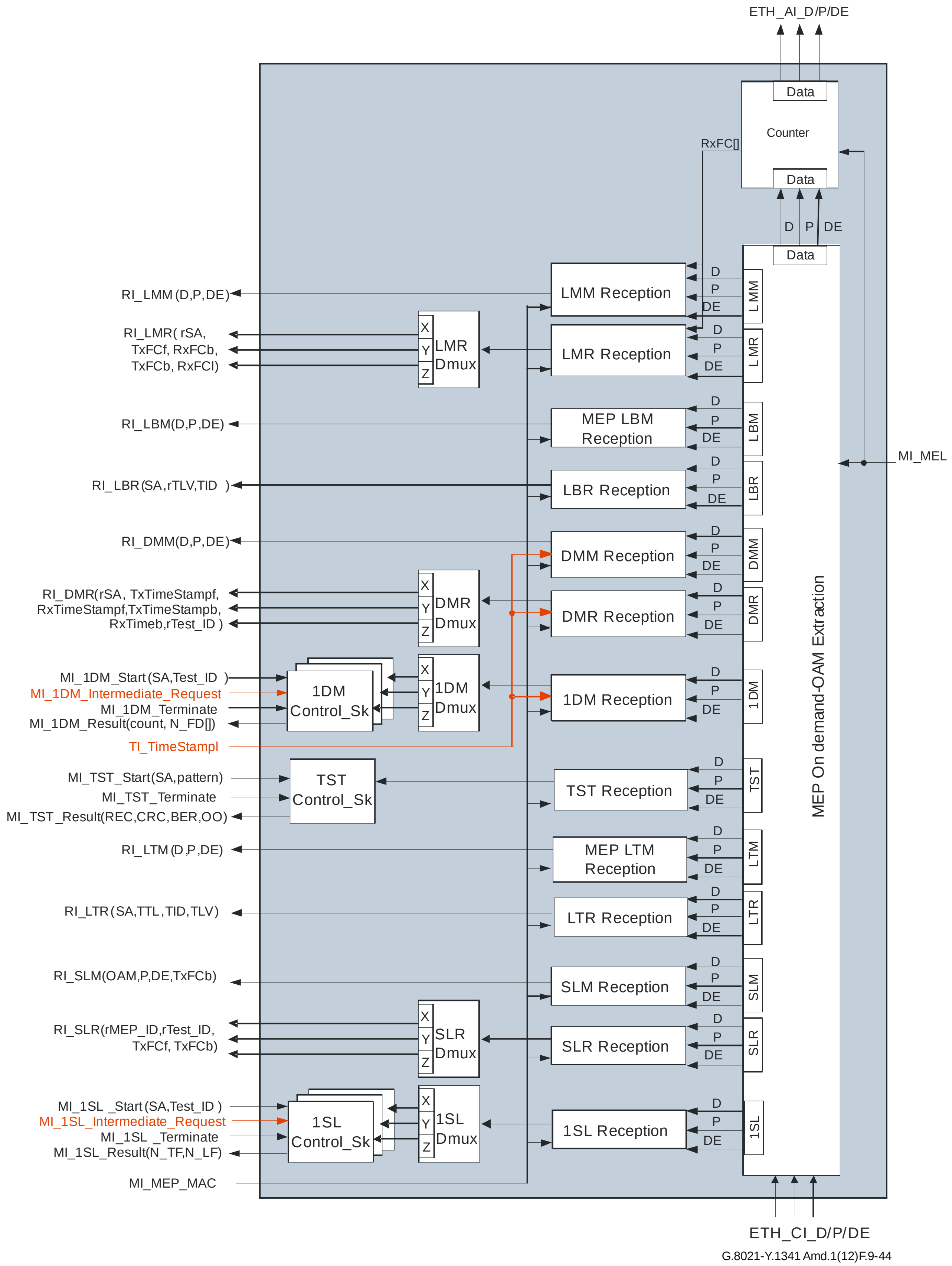

Figure 9-44 – ETHDe_FT_Sk processes

ITU-T Y-SERIES RECOMMENDATIONS

GLOBAL INFORMATION INFRASTRUCTURE, INTERNET PROTOCOL ASPECTS AND NEXT-GENERATION NETWORKS

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
Numbering, naming and addressing	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

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	General tariff principles
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	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	Terminals and subjective and objective assessment methods
Series Q	Switching and signalling
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 and next-generation networks**
Series Z	Languages and general software aspects for telecommunication systems

Printed in Switzerland
Geneva, 2013