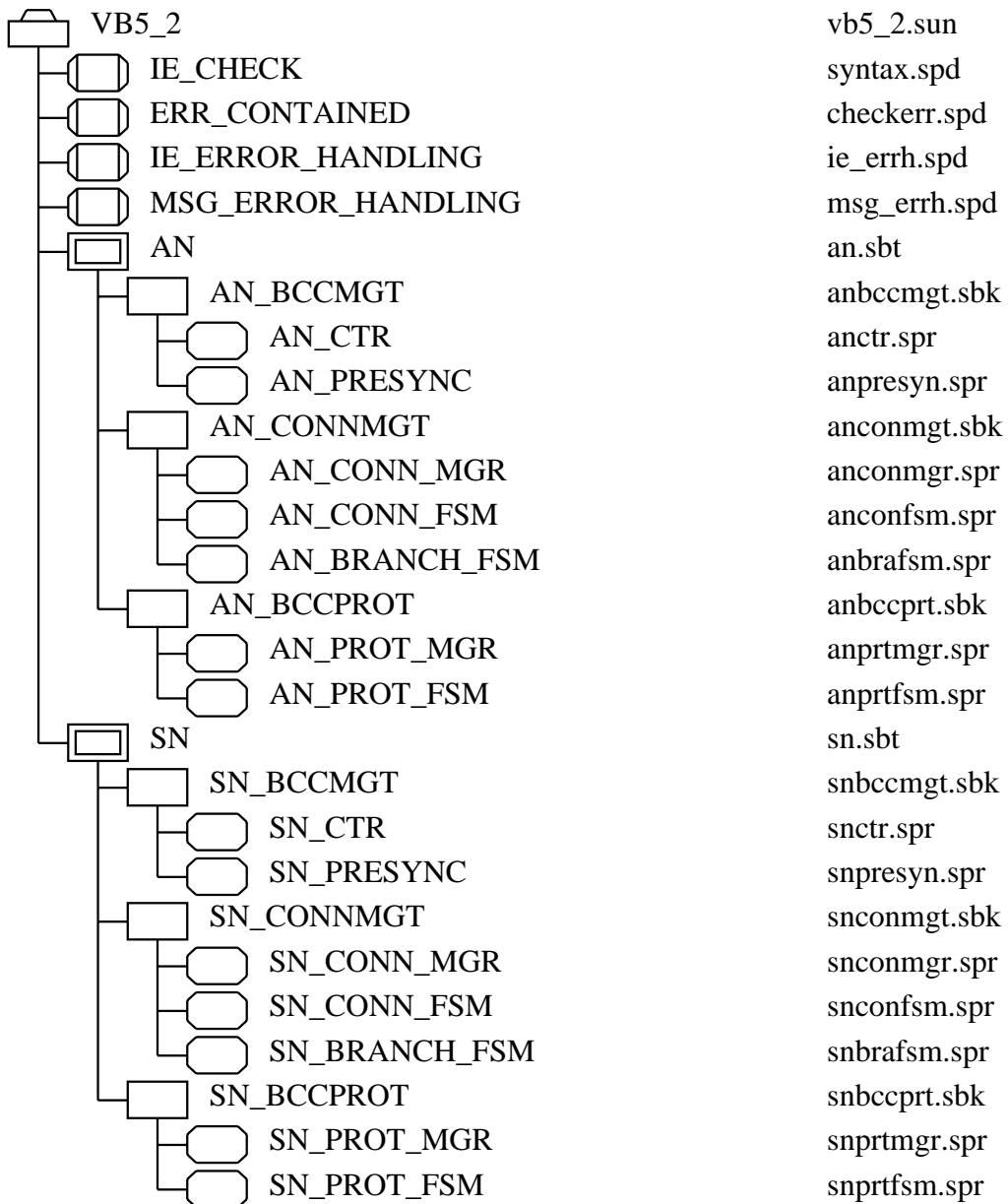


Table of Contents

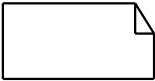
| | |
|------------------------------|-----|
| Organizer View | 1 |
| Package VB5_2 | 2 |
| Procedure IE_CHECK | 24 |
| Procedure ERR_CONTAINED | 25 |
| Procedure IE_ERROR_HANDLING | 26 |
| Procedure MSG_ERROR_HANDLING | 27 |
| Block Type AN | 28 |
| Block AN_BCCMGT | 29 |
| Process AN_CTR | 31 |
| Process AN_PRESYNC | 36 |
| Block AN_CONNMGT | 37 |
| Process AN_CONN_MGR | 40 |
| Process AN_CONN_FSM | 52 |
| Process AN_BRANCH_FSM | 64 |
| Block AN_BCCPROT | 68 |
| Process AN_PROT_MGR | 71 |
| Process AN_PROT_FSM | 93 |
| Block Type SN | 108 |
| Block SN_BCCMGT | 109 |
| Process SN_CTR | 111 |
| Process SN_PRESYNC | 119 |
| Block SN_CONNMGT | 121 |
| Process SN_CONN_MGR | 125 |
| Process SN_CONN_FSM | 138 |
| Process SN_BRANCH_FSM | 148 |
| Block SN_BCCPROT | 150 |
| Process SN_PROT_MGR | 153 |
| Process SN_PROT_FSM | 167 |

Diagram Structure



Systems





Package VB5_2

1(22)



/* The block types
AN & SN. */

/* Process types that are
used both in AN and SN. */

/* Procedures that are used
both in AN and SN. */

AN

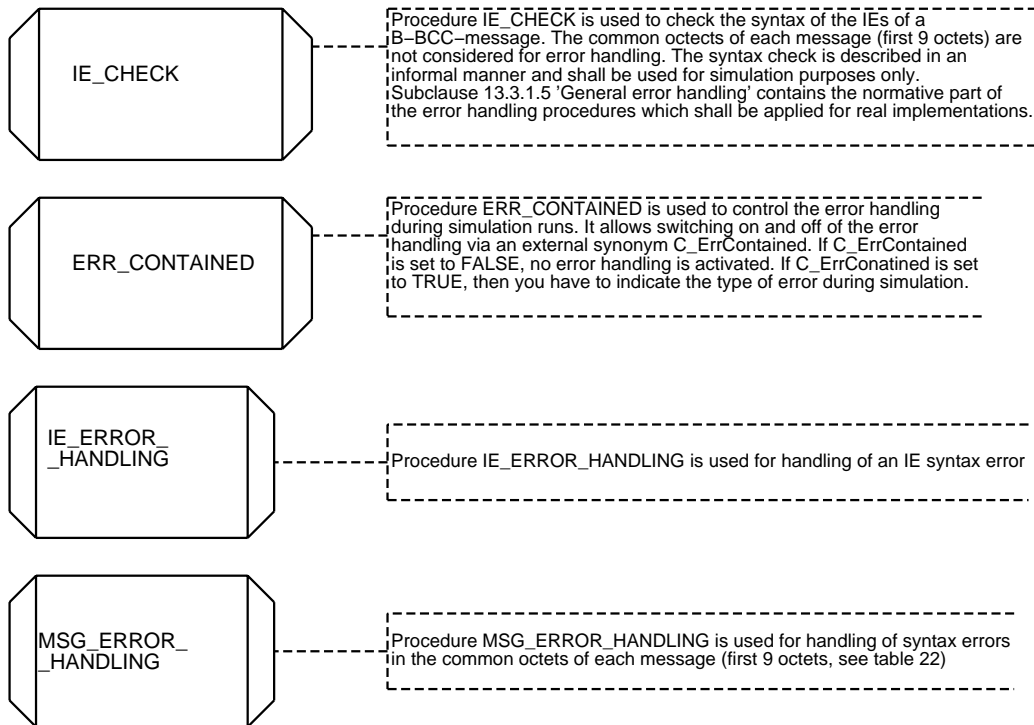
SN

Package VB5_2

2(22)

/*Procedures, data and comments for
enabling/facilitating
tool supported simulation*/

/* For handling of protocol syntax errors, two types of errors are distinguished (see subclauses 13.6.1.5 and 14.1 on general error handling procedures)
type 1: errors in the common message part, e.g. protocol discriminator, transaction id, message type
type 2: errors within message parameters (IE errors)
The checking for and handling of errors is shown in the SDLs in a different way.
For type 1 errors it is assumed that the common message part of RTMC messages is checked prior to processing in the SDL processes. These errors are represented in the SDLs by pseudo events. The SDLs show the subsequent error handling (MSG_ERROR_HANDLING).
For type 2 errors the SDLs show the possible results of IE syntax checking (procedure IE_CHECK) only and the subsequent error handling (procedure IE_ERROR_HANDLING)
*/



/* Shall error checking be done */
SYNONYM C_ErrContained BOOLEAN = EXTERNAL;



Package VB5_2

3(22)



/*Procedures, data and comments for
enabling/facilitating
tool supported simulation*/

/* Declaration of primitives for establishing the communication
between the protocol management processes at the AN and SN side. */

signal
simSnStarted; /* primitive to announce the PId of the SN_PROT_MGR process to the AN_PROT_MGR process */
simAnStarted; /* primitive to announce the PId of the AN_PROT_MGR process to the SN_PROT_MGR process */

/* Example for command file for the startup of the AN and SN VB5 SDL system. After
system startup further commands may be directed to the simulator
depending on the specific use case you want to perform.

----- begin of file
Go
Output-Via meeBbccStartupReq CH_SNVB5_MF
Proceed-Until +0.1
Output-Via aalBbccEstablishConf CH_SNVB5_SAAL
Output-Via aalBbccEstablishInd CH_ANVB5_SAAL
Proceed-Until +0.1
Proceed-Until +0.1
Output-Via ceeBbccResetAccRes (: (others: (. 0,0 .)) :) CH_ANVB5_CCF
go
----- end of file
*/

/*
Before you can start simulation, the values for the external synonym C_ErrContained must
be set. To avoid direct input during simulation start, you should use an input-file instead.
Example of an input file:
----- begin of input file
C_ErrContained FALSE;
----- end of input-file
*/

/* The VB5 messages DEALLOC, DROP_BRANCH and the corresponding responses and primitives may
contain a list of connection reference numbers or branch identifiers e.g. several connections may be
deallocated simultaneously. For simplification of the SDL specification, only one connection or
branch identifier is used but no list of parameters. Furthermore no segmentation of the messages is
shown in the SDLs. The segmentation of invoke messages and the collection of response messages
has to be taken into account in a real implementation. */



Package VB5_2

4(22)



/* Naming conventions */

```

/*Type definitions for signal parameters*/
/**/
/*In the following the SDL data types of signal parameters are defined. This is done in order
to allow for tool supported simulation and validation of the SDLs. This SDL data modelling
shall not supersede the message and parameter formats and codes as defined in clause 14.
Only those data are modelled which are needed for simulation and validation*/
/**/
/*Prefix Naming Conventions for Data Type Definitions
AT_xxx  ARRAY Data Type
IT_xxx  INTEGER Data Type
PT_xxx  PID Data Type
ST_xxx  STRUCT Data Type*/
/**/
/*Prefix Naming Conventions for Variables and Constants
A_xxx  ARRAY Variable
B_xxx  BOOLEAN Variable
C_xxx  Constant of any type
P_xxx  PID Variable
S_xxx  STRUCT Variable
V_xxx  INTEGER or NATURAL Variable
*/

```

```

/*Prefixes for primitives*/
/**/
/*B-BCC primitives represent in an abstract way the exchange of information and control
between functional entities. They do not specify or constrain implementation. The primitives
are classified as follows*/
/*
aal      primitives between the B-BCC management and the ATM adaptation layer
mdu      primitives between B-BCC management or connection management and B-BCC protocol
mee      primitives between B-BCC management or B-BCC protocol and the
          B-BCC management application functions
mie      primitives internal to a SDL block
mph      primitives between B-BCC management and connection management
cee      primitives between B-BCC connection management and the connection control
          application functions
*/

```

/* Drawing conventions */

```

/*In order to distinguish internal and external communication, two different types of input and output
symbols are used. External signals use input symbols coming from the left and output symbols
pointing to the left. Internal signals use input symbols coming from the right and output symbols
pointing to the right. External signals are signals coming from or going to the environment,
i.e. management entity, SAAL entity and remote side.
*/

```

Package VB5_2

5(22)

/*Data Type Definitions for
B-BCC messages*/

```

/* Information Elements */
/**/
/* Protocol error cause */
SYNTYPE IT_ProtErrCause = INTEGER
ENDSYNTYPE IT_ProtErrCause;
/*Possible Protocol error cause values*/
SYNONYM C_UnkMsgType           IT_ProtErrCause = 1;
SYNONYM C_ReplE                IT_ProtErrCause = 2;
SYNONYM C_MandIEMiss           IT_ProtErrCause = 3;
SYNONYM C_UnrecogIE            IT_ProtErrCause = 4;
SYNONYM C_IEContErr            IT_ProtErrCause = 5;
SYNONYM C_IENotAllowed          IT_ProtErrCause = 6;
/**/
/* Transaction identifier */
SYNTYPE IT_TransId = INTEGER
ENDSYNTYPE IT_TransId;
/**/
/* Connection reference number */
SYNTYPE IT_ConnRefNo = INTEGER
ENDSYNTYPE IT_ConnRefNo;
/**/
/* Branch identifier */
SYNTYPE IT_BranchId = INTEGER
ENDSYNTYPE IT_BranchId;
/* For simulation only, the value '0' is used to indicate a not present
branch identifier. Although this value may be used in a real
implementation as branch identifier. This SDL data modelling
does not supersede the message and parameter formats and codes
as defined in clause 14.*/
SYNONYM C_BranchIdNotPresent    IT_BranchId = 0;
/**/
/* Reject Cause */
SYNTYPE IT_RejCause = INTEGER
ENDSYNTYPE IT_RejCause;
SYNONYM C_Success                IT_RejCause = 255;
SYNONYM C_TransErr               IT_RejCause = 1;
SYNONYM C_IncompConnState        IT_RejCause = 2;
SYNONYM C_IncompBranchState      IT_RejCause = 3;
SYNONYM C_DeallocExpected        IT_RejCause = 4;
SYNONYM C_OpRejLoc               IT_RejCause = 5;
SYNONYM C_Saal                  IT_RejCause = 6;

```

/* Information Elements */
/* All other IEs are transported transparently through the vb5-system.
They are neither set nor evaluated by the vb5-system. */

Package VB5_2

6(22)

/*Data type definitions for
primitives */

```

/* Struct necessary to identify a single branch */
NEWTYPE ST_ConnBranchId STRUCT
  V_ConnRefNo  IT_ConnRefNo;
  V_BranchId  IT_BranchId;
ENDNEWTYPE;
SYNONYM C_EmptyElement ST_ConnBranchId = (. 0, 0 .);
/* Default value of an empty struct.
   For simulation only, the values '0' are used to indicate an empty element.
   Although these values may be used in a real implementation as connection
   reference number or branch identifier. This SDL data modelling does not
   supersede the message and parameter formats and codes as defined in
   clause 14.*/
/**/
/* List of connections, used in reset procedure */
NEWTYPE AT_ConnList ARRAY
  (INTEGER, ST_ConnBranchId)
ENDNEWTYPE;
/**/
/* Update needed in case of addBranch */
SYNTYPE IT_UpdateNeeded = INTEGER
  CONSTANTS 0:1
ENDSYNTYPE IT_UpdateNeeded;
/* Possible values for IT_UpdateNeeded */
SYNONYM B_UpdateNotNeeded IT_UpdateNeeded = 0;
SYNONYM B_UpdateNeeded IT_UpdateNeeded = 1;
/**/
/* Indicator for response primitives */
SYNTYPE IT_Response = INTEGER
  CONSTANTS 0:1
ENDSYNTYPE IT_Response;
/* Possible values for IT_Response */
SYNONYM C_NoResponse IT_Response = 0;
SYNONYM C_Response IT_Response = 1;

```

/*Data type definitions for
common used data types */

```

/* Data type used to indicate the result of the syntax check of the information elements */
SYNTYPE IT_SynResult = INTEGER
ENDSYNTYPE IT_SynResult;
/*Possible values for IT_SynResult */
SYNONYM C_Proceed INTEGER =1; /*possible result of syntax check: proceed with message processing*/
SYNONYM C_Error INTEGER =2; /*possible result of syntax check: error in message*/
/**/
/* Data type used to identify the own protocol process in a transaction */
SYNTYPE PT_ProId = PID
ENDSYNTYPE PT_ProId;

```


Package VB5_2

7(22)

/*Declaration of B-BCC messages*/

```

signal
ALLOC      (IT_TransId, IT_ConnRefNo, IT_BranchId),
ALLOC_ACC  (IT_TransId, IT_BranchId),
ALLOC_REJ  (IT_TransId, IT_RejCause),
ALLOC_COMP (IT_TransId, IT_ConnRefNo),
ALLOC_COMP_ACC  (IT_TransId),
ALLOC_COMP_REJ  (IT_TransId, IT_RejCause),
AN_FAULT      (IT_TransId, IT_ConnRefNo, IT_BranchId),
AN_FAULT_ACC  (IT_TransId),
BBCC_RESET    (IT_TransId),
BBCC_RESET_ACC (IT_TransId),
BBCC_RESET_REJ (IT_TransId, IT_RejCause),
PRESYNC      (IT_TransId),
PRESYNC_ACC  (IT_TransId),
PRESYNC_REJ  (IT_TransId),
DEALLOC      (IT_TransId, IT_ConnRefNo),
DEALLOC_ACC  (IT_TransId),
MODIFY      (IT_TransId, IT_ConnRefNo),
MODIFY_ACC  (IT_TransId),
MODIFY_REJ  (IT_TransId, IT_RejCause),
MODIFY_COMP (IT_TransId, IT_ConnRefNo),
MODIFY_COMP_ACC  (IT_TransId),
MODIFY_COMP_REJ  (IT_TransId, IT_RejCause),
MODIFY_ABORT  (IT_TransId, IT_ConnRefNo),
MODIFY_ABORT_ACC (IT_TransId),
MODIFY_ABORT_REJ (IT_TransId, IT_RejCause),
ADD_BRANCH    (IT_TransId, IT_ConnRefNo, IT_BranchId),
ADD_BRANCH_ACC (IT_TransId),
ADD_BRANCH_REJ (IT_TransId, IT_RejCause),
UPDATE_BRANCH (IT_TransId, IT_ConnRefNo, IT_BranchId),
UPDATE_BRANCH_ACC (IT_TransId),
UPDATE_BRANCH_REJ (IT_TransId, IT_RejCause),
DROP_BRANCH   (IT_TransId, IT_ConnRefNo, IT_BranchId),
DROP_BRANCH_ACC (IT_TransId),
DROP_BRANCH_REJ (IT_TransId, IT_RejCause),
PROTOCOL_ERROR (IT_TransId, IT_ProtErrCause);

```

```

/*Primitives to the B-BCC protocol entity representing erroneous VB5 messages */
signal
unknown_message(IT_TransId),
/*Representation of a message containing an unspecified message type information element
*/
msg_too_short,
/*Representation of a message that is too short to contain a complete message length information element
*/
trans_id_err,
/*Representation of a message containing an invalid transaction identifier format or a transaction identifier procedural error
*/
unknown_protdisc;
/*Representation of a message containing an unspecified protocol discriminator information element*/

```



Package VB5_2

8(22)



/* Declaration of primitives
from/to the SAAL */

```
signal
aalBbccEstablishReq,
/*Request from VB5 system management to SAAL to establish SAAL connection to the peer side
*/
aalBbccEstablishConf,
/*Confirmation from SAAL to VB5 system management of successful SAAL establishment
*/
aalBbccEstablishInd,
/*Indication from SAAL to VB5 system management that peer side requested SAAL establishment
*/
aalBbccReleaseReq,
/*Request from VB5 system management to SAAL to release SAAL connection to the peer side
*/
aalBbccReleaseConf,
/*Confirmation from SAAL to VB5 system management of successful SAAL release
*/
aalBbccReleaseInd;
/*Indication from SAAL to VB5 system management that peer side requested SAAL release*/
```

Package VB5_2

9(22)

/*Declaration of common primitives
from/to the management application function*/

signal
meeBbccStartTrafficInd;
meeBbccStopTrafficReq;
meeBbccStopTrafficConf;
meeBbccStopTrafficInd;
meeBbccErrorInd;

/*Declaration of AN specific primitives
from/to the connection and control application function*/

signal
ceeAllocInd (IT_ConnRefNo, IT_BranchId),
ceeAllocAccRes (IT_ConnRefNo, IT_BranchId),
ceeAllocRejRes (IT_ConnRefNo, IT_BranchId, IT_RejCause),
ceeAllocCompInd (IT_ConnRefNo),
ceeAllocCompAccRes (IT_ConnRefNo),
ceeAllocCompRejRes (IT_ConnRefNo, IT_RejCause),
ceeDeallocInd (IT_ConnRefNo),
ceeDeallocAccRes (IT_ConnRefNo),
ceeModifyInd (IT_ConnRefNo),
ceeModifyAccRes (IT_ConnRefNo),
ceeModifyRejRes (IT_ConnRefNo, IT_RejCause),
ceeModifyCompInd (IT_ConnRefNo),
ceeModifyCompAccRes (IT_ConnRefNo),
ceeModifyCompRejRes (IT_ConnRefNo, IT_RejCause),
ceeModifyAbortInd (IT_ConnRefNo),
ceeModifyAbortAccRes (IT_ConnRefNo),
ceeModifyAbortRejRes (IT_ConnRefNo, IT_RejCause),
ceeAddBranchInd (IT_ConnRefNo, IT_BranchId),
ceeAddBranchAccRes (IT_ConnRefNo, IT_BranchId, IT_UpdateNeeded),
ceeAddBranchRejRes (IT_ConnRefNo, IT_BranchId, IT_RejCause),
ceeUpdateBranchInd (IT_ConnRefNo, IT_BranchId),
ceeUpdateBranchAccRes (IT_ConnRefNo, IT_BranchId),
ceeUpdateBranchRejRes (IT_ConnRefNo, IT_BranchId, IT_RejCause),
ceeDropBranchInd (IT_ConnRefNo, IT_BranchId),
ceeDropBranchAccRes (IT_ConnRefNo, IT_BranchId),
ceeBbccResetInd ,
ceeBbccResetAccRes (AT_ConnList),
ceeBbccResetRejRes (IT_RejCause),
ceeAnFaultReq (IT_ConnRefNo, IT_BranchId),
ceeAnFaultAccConf (IT_ConnRefNo, IT_BranchId, IT_RejCause);

/*
It is assumed that the ceeAnFaultReq primitive
at the management interface always contains at
least the connection reference number and
optionally a branch identifier as parameters.
This does not mean that they are included as
parameters in the AN_FAULT message at the
VB5.2 reference point. They may be omitted if the
user port connection identifier or the service port
connection identifier are present.
*/

/*Declaration of AN specific primitives
from/to the management application function*/

signal
meeBbccStartTrafficReq;
meeBbccStartTrafficConf;
meeBbccPresyncInd ,
meeBbccPresyncAccRes,
meeBbccPresyncRejRes;

Package VB5_2

10(22)

/*Declaration of SN specific primitives
from/to the connection and control application function*/

```

signal
ceeAllocReq      (IT_ConnRefNo, IT_BranchId),
ceeAllocAccConf  (IT_ConnRefNo, IT_BranchId),
ceeAllocRejConf  (IT_ConnRefNo, IT_BranchId, IT_RejCause),
ceeAllocCompReq  (IT_ConnRefNo),
ceeAllocCompAccConf  (IT_ConnRefNo),
ceeAllocCompRejConf  (IT_ConnRefNo, IT_RejCause),
ceeDeallocReq    (IT_ConnRefNo),
ceeDeallocAccConf  (IT_ConnRefNo),
ceeDeallocRejConf  (IT_ConnRefNo, IT_RejCause),
ceeModifyReq     (IT_ConnRefNo),
ceeModifyAccConf  (IT_ConnRefNo),
ceeModifyRejConf  (IT_ConnRefNo, IT_RejCause),
ceeModifyCompReq  (IT_ConnRefNo),
ceeModifyCompAccConf  (IT_ConnRefNo),
ceeModifyCompRejConf  (IT_ConnRefNo, IT_RejCause),
ceeModifyAbortReq  (IT_ConnRefNo),
ceeModifyAbortAccConf  (IT_ConnRefNo),
ceeModifyAbortRejConf  (IT_ConnRefNo, IT_RejCause),
ceeAddBranchReq   (IT_ConnRefNo, IT_BranchId),
ceeAddBranchAccConf  (IT_ConnRefNo, IT_BranchId),
ceeAddBranchRejConf  (IT_ConnRefNo, IT_BranchId, IT_RejCause),
ceeUpdateBranchReq  (IT_ConnRefNo, IT_BranchId),
ceeUpdateBranchAccConf  (IT_ConnRefNo, IT_BranchId),
ceeUpdateBranchRejConf  (IT_ConnRefNo, IT_BranchId, IT_RejCause),
ceeDropBranchReq   (IT_ConnRefNo, IT_BranchId),
ceeDropBranchAccConf  (IT_ConnRefNo, IT_BranchId),
ceeDropBranchRejConf  (IT_ConnRefNo, IT_BranchId, IT_RejCause),
ceeBbccResetReq    (AT_ConnList),
ceeBbccResetAccConf  (IT_RejCause),
ceeBbccResetRejConf  (IT_RejCause),
ceeAnFaultInd      (IT_ConnRefNo, IT_BranchId);

```

/*Declaration of SN specific primitives
from/to the management application function*/

```

signal
meeBbccPresyncReq
meeBbccPresyncAccConf  (IT_RejCause),
meeBbccPresyncRejConf  (IT_RejCause),
meeBbccStartupReq      (IT_RejCause),
meeBbccStartupConf     (IT_RejCause),
meeBbccRestartReq      (IT_RejCause),
meeBbccRestartConf     (IT_RejCause),
meeBbccStartResetInd;

```

Package VB5_2

11(22)

/*Declaration of common primitives
from/to BCC protocol entity*/

signal
mduStartTraffic;
mduStopTraffic;
mduErrorInd;

/*Declaration of AN specific primitives
from/to BCC protocol entity*/

```
signal
mduAllocInd      (PT_ProtId, IT_ConnRefNo, IT_BranchId),
mduAllocAccRes   (PT_ProtId, IT_ConnRefNo, IT_BranchId),
mduAllocRejRes   (PT_ProtId, IT_ConnRefNo, IT_BranchId, IT_RejCause),
mduAllocCompInd  (PT_ProtId, IT_ConnRefNo),
mduAllocCompAccRes (PT_ProtId, IT_ConnRefNo),
mduAllocCompRejRes (PT_ProtId, IT_ConnRefNo, IT_RejCause),
mduDeallocInd    (PT_ProtId, IT_ConnRefNo),
mduDeallocAccRes (PT_ProtId, IT_ConnRefNo),
mduModifyInd     (PT_ProtId, IT_ConnRefNo),
mduModifyAccRes  (PT_ProtId, IT_ConnRefNo),
mduModifyRejRes  (PT_ProtId, IT_ConnRefNo, IT_RejCause),
mduModifyCompInd (PT_ProtId, IT_ConnRefNo),
mduModifyCompAccRes (PT_ProtId, IT_ConnRefNo),
mduModifyCompRejRes (PT_ProtId, IT_ConnRefNo, IT_RejCause),
mduModifyAbortInd (PT_ProtId, IT_ConnRefNo),
mduModifyAbortAccRes (PT_ProtId, IT_ConnRefNo),
mduModifyAbortRejRes (PT_ProtId, IT_ConnRefNo, IT_RejCause),
mduAddBranchInd  (PT_ProtId, IT_ConnRefNo, IT_BranchId),
mduAddBranchAccRes (PT_ProtId, IT_ConnRefNo, IT_BranchId),
mduAddBranchRejRes (PT_ProtId, IT_ConnRefNo, IT_BranchId, IT_RejCause),
mduUpdateBranchInd (PT_ProtId, IT_ConnRefNo, IT_BranchId),
mduUpdateBranchAccRes (PT_ProtId, IT_ConnRefNo, IT_BranchId),
mduUpdateBranchRejRes (PT_ProtId, IT_ConnRefNo, IT_BranchId, IT_RejCause),
mduDropBranchInd (PT_ProtId, IT_ConnRefNo, IT_BranchId),
mduDropBranchAccRes (PT_ProtId, IT_ConnRefNo, IT_BranchId),
mduDropBranchRejRes (PT_ProtId, IT_ConnRefNo, IT_BranchId, IT_RejCause),
mduAnFaultReq    (IT_ConnRefNo, IT_BranchId, IT_Response),
mduAnFaultAccConf (IT_ConnRefNo, IT_BranchId, IT_RejCause),
mduBbccResetInd  (PT_ProtId),
mduBbccResetAccRes (PT_ProtId),
mduBbccResetRejRes (PT_ProtId, IT_RejCause),
mduBbccPresyncInd (PT_ProtId),
mduBbccPresyncAccRes (PT_ProtId),
mduBbccPresyncRejRes (PT_ProtId);
```

Package VB5_2

12(22)

/*Declaration of SN specific primitives
from/to BCC protocol entity*/

```

signal
mduAllocReq      (IT_ConnRefNo, IT_BranchId),
mduAllocAccConf  (IT_ConnRefNo, IT_BranchId),
mduAllocRejConf  (IT_ConnRefNo, IT_BranchId, IT_RejCause),
mduAllocCompReq  (IT_ConnRefNo),
mduAllocCompAccConf (IT_ConnRefNo),
mduAllocCompRejConf (IT_ConnRefNo, IT_RejCause),
mduDeallocReq    (IT_ConnRefNo),
mduDeallocAccConf (IT_ConnRefNo),
mduDeallocRejConf (IT_ConnRefNo, IT_RejCause),
mduModifyReq     (IT_ConnRefNo),
mduModifyAccConf (IT_ConnRefNo),
mduModifyRejConf (IT_ConnRefNo, IT_RejCause),
mduModifyCompReq (IT_ConnRefNo),
mduModifyCompAccConf (IT_ConnRefNo),
mduModifyCompRejConf (IT_ConnRefNo, IT_RejCause),
mduModifyAbortReq (IT_ConnRefNo),
mduModifyAbortAccConf (IT_ConnRefNo),
mduModifyAbortRejConf (IT_ConnRefNo, IT_RejCause),
mduAddBranchReq  (IT_ConnRefNo, IT_BranchId),
mduAddBranchAccConf (IT_ConnRefNo, IT_BranchId),
mduAddBranchRejConf (IT_ConnRefNo, IT_BranchId, IT_RejCause),
mduUpdateBranchReq (IT_ConnRefNo, IT_BranchId),
mduUpdateBranchAccConf (IT_ConnRefNo, IT_BranchId),
mduUpdateBranchRejConf (IT_ConnRefNo, IT_BranchId, IT_RejCause),
mduDropBranchReq (IT_ConnRefNo, IT_BranchId),
mduDropBranchAccConf (IT_ConnRefNo, IT_BranchId),
mduDropBranchRejConf (IT_ConnRefNo, IT_BranchId, IT_RejCause),
mduAnFaultInd    (IT_ConnRefNo, IT_BranchId),
mduBbccResetReq  ,
mduBbccResetAccConf ,
mduBbccResetRejConf (IT_RejCause),
mduBbccPresyncReq ,
mduBbccPresyncAccConf ,
mduBbccPresyncRejConf (IT_RejCause);

```



Package VB5_2

13(22)



/*Declaration of common primitives
between BCC management and connection management */

signal
mphStopTraffic,
mphStartTraffic;

/*Declaration of SN specific primitives
between BCC management and connection management */

signal
mphBbccResetReq,
mphBbccResetAccConf,
mphBbccResetRejConf (IT_RejCause);



Package VB5_2

14(22)



/*Declaration of signallists
common for AN and SN,
from/to block BCCMG */

signallist SL_CTR_SAAL =
aalBbccEstablishReq,
aalBbccReleaseReq;

signallist SL_SAAL_CTR =
aalBbccEstablishConf,
aalBbccEstablishInd,
aalBbccReleaseConf,
aalBbccReleaseInd;

signallist SL_CTR_PROT =
mduStartTraffic,
mduStopTraffic;

signallist SL_PROT_CTR =
mduErrorInd;



Package VB5_2

15(22)



```
/*Declaration of signallists
from/to block AN_BCCMGT */
```

```
signallist SL_MF_ANCTR =
meeBbccStartTrafficReq,
meeBbccStopTrafficReq;
```

```
signallist SL_ANCTR_MF =
meeBbccStartTrafficConf,
meeBbccStartTrafficInd,
meeBbccStopTrafficConf,
meeBbccStopTrafficInd,
meeBbccErrorInd;
```

```
signallist SL_ANPRESYNC_PROT =
mduBbccPresyncAccRes,
mduBbccPresyncRejRes;
```

```
signallist SL_PROT_ANPRESYNC =
mduBbccPresyncInd;
```

```
signallist SL_ANPRESYNC_MF =
meeBbccPresyncInd;
```

```
signallist SL_MF_ANPRESYNC =
meeBbccPresyncAccRes,
meeBbccPresyncRejRes;
```

```
signallist SL_CTR_ANCONN =
mphStartTraffic,
mphStopTraffic;
```

```
signallist SL_ANMGT_PROT =
(SL_CTR_PROT),
(SL_ANPRESYNC_PROT);
```

```
signallist SL_PROT_ANMGT =
(SL_PROT_ANPRESYNC),
(SL_PROT_CTR);
```

```
signallist SL_ANMGT_CONN =
(SL_CTR_ANCONN);
```

Package VB5_2

16(22)

/*Declaration of signallists
from/to block AN_CONNMGMT */

signallist SL_ANCMGR_CCF =
ceeAllocInd,
ceeAllocCompInd,
ceeDeallocInd,
ceeModifyInd,
ceeModifyCompInd,
ceeModifyAbortInd,
ceeAddBranchInd,
ceeUpdateBranchInd,
ceeDropBranchInd,
ceeBbccResetInd,
ceeAnFaultAccConf;

signallist SL_CCF_ANCMGR =
ceeAllocAccRes,
ceeAllocRejRes,
ceeAllocCompAccRes,
ceeAllocCompRejRes,
ceeDeallocAccRes,
ceeModifyAccRes,
ceeModifyRejRes,
ceeModifyCompAccRes,
ceeModifyCompRejRes,
ceeModifyAbortAccRes,
ceeModifyAbortRejRes,
ceeAddBranchAccRes,
ceeAddBranchRejRes,
ceeUpdateBranchAccRes,
ceeUpdateBranchRejRes,
ceeDropBranchAccRes,
ceeBbccResetAccRes,
ceeBbccResetRejRes,
ceeAnFaultReq;

signallist SL_ANCONN_CCF =
(SL_ANCMGR_CCF);

signallist SL_CCF_ANCONN =
(SL_CCF_ANCMGR);

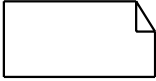
signallist SL_ANPROT_CMGR =
mduAllocInd,
mduAllocCompInd,
mduDeallocInd,
mduModifyInd,
mduModifyCompInd,
mduModifyAbortInd,
mduAddBranchInd,
mduUpdateBranchInd,
mduDropBranchInd,
mduBbccResetInd,
mduAnFaultAccConf;

signallist SL_CMGR_ANPROT =
mduAllocAccRes,
mduAllocRejRes,
mduAllocCompAccRes,
mduAllocCompRejRes,
mduDeallocAccRes,
mduModifyAccRes,
mduModifyRejRes,
mduModifyCompAccRes,
mduModifyCompRejRes,
mduModifyAbortAccRes,
mduModifyAbortRejRes,
mduAddBranchAccRes,
mduAddBranchRejRes,
mduUpdateBranchAccRes,
mduUpdateBranchRejRes,
mduDropBranchAccRes,
mduDropBranchRejRes,
mduBbccResetAccRes,
mduBbccResetRejRes,
mduAnFaultReq;

signallist SL_ANPROT_CONN =
(SL_ANPROT_CMGR);

signallist SL_CONN_ANPROT =
(SL_CMGR_ANPROT);

signallist SL_ANMGT_CMGR =
(SL_ANMGT_CONN);



Package VB5_2

17(22)



```
/*Declaration of signallists
from/to block AN_BCCPROT */
```

```
signallist SL_VB5_ANPROT =
ALLOC,
ALLOC_COMP,
AN_FAULT_ACC,
BBCC_RESET,
PRESYNC,
DEALLOC,
MODIFY,
MODIFY_COMP,
MODIFY_ABORT,
ADD_BRANCH,
UPDATE_BRANCH,
DROP_BRANCH,
PROTOCOL_ERROR,
unknown_protdisc,
unknown_message,
msg_too_short,
trans_id_err,
simSnStarted;
```

```
signallist SL_ANPROT_VB5 =
ALLOC_ACC,
ALLOC_REJ,
ALLOC_COMP_ACC,
ALLOC_COMP_REJ,
AN_FAULT,
BBCC_RESET_ACC,
BBCC_RESET_REJ,
PRESYNC_ACC,
PRESYNC_REJ,
DEALLOC_ACC,
MODIFY_ACC,
MODIFY_REJ,
MODIFY_COMP_ACC,
MODIFY_COMP_REJ,
MODIFY_ABORT_ACC,
MODIFY_ABORT_REJ,
ADD_BRANCH_ACC,
ADD_BRANCH_REJ,
UPDATE_BRANCH_ACC,
UPDATE_BRANCH_REJ,
DROP_BRANCH_ACC,
DROP_BRANCH_REJ,
PROTOCOL_ERROR,
unknown_protdisc,
unknown_message,
msg_too_short,
trans_id_err,
simAnStarted;
```

Package VB5_2

18(22)

/*Declaration of signallists
from/to block SN_BCCMGT */

signallist SL_MF_SNCTR =
meeBbccStopTrafficReq,
meeBbccStartupReq,
meeBbccRestartReq;

signallist SL_SNCTR_MF =
meeBbccStartTrafficInd,
meeBbccStopTrafficConf,
meeBbccStopTrafficInd,
meeBbccStartupConf,
meeBbccRestartConf,
meeBbccStartResetInd,
meeBbccErrorInd;

signallist SL_SNPRESYNC_PROT =
mduBbccPresyncReq;

signallist SL_PROT_SNPRESYNC =
mduBbccPresyncAccConf,
mduBbccPresyncRejConf;

signallist SL_SNPRESYNC_MF =
meeBbccPresyncAccConf,
meeBbccPresyncRejConf;

signallist SL_MF_SNPRESYNC =
meeBbccPresyncReq;

signallist SL_CTR_SNCONN =
mphStartTraffic,
mphStopTraffic,
mphBbccResetReq;

signallist SL_SNCONN_CTR =
mphBbccResetRejConf,
mphBbccResetAccConf;

signallist SL_SNMGT_PROT =
(SL_CTR_PROT),
(SL_SNPRESYNC_PROT);

signallist SL_PROT_SNMGT =
(SL_PROT_SNPRESYNC),
(SL_PROT_CTR);

signallist SL_SNMGT_CONN =
(SL_CTR_SNCONN);

signallist SL_CONN_SNMGT =
(SL_SNCONN_CTR);



Package VB5_2

19(22)



/*Declaration of signallists
from/to block SN_CONNMGMT */

signallist SL_SNCMGR_CCF =
ceeAllocAccConf,
ceeAllocRejConf,
ceeAllocCompAccConf,
ceeAllocCompRejConf,
ceeDeallocAccConf,
ceeDeallocRejConf,
ceeModifyAccConf,
ceeModifyRejConf,
ceeModifyCompAccConf,
ceeModifyCompRejConf,
ceeModifyAbortAccConf,
ceeModifyAbortRejConf,
ceeAddBranchAccConf,
ceeAddBranchRejConf,
ceeUpdateBranchAccConf,
ceeUpdateBranchRejConf,
ceeDropBranchAccConf,
ceeDropBranchRejConf,
ceeBbccResetAccConf,
ceeBbccResetRejConf,
ceeAnFaultInd;

signallist SL_CCF_SNCMGR =
ceeAllocReq,
ceeAllocCompReq,
ceeDeallocReq,
ceeModifyReq,
ceeModifyCompReq,
ceeModifyAbortReq,
ceeAddBranchReq,
ceeUpdateBranchReq,
ceeDropBranchReq,
ceeBbccResetReq;

signallist SL_SNCONN_CCF =
(SL_SNCMGR_CCF);

signallist SL_CCF_SNCONN =
(SL_CCF_SNCMGR);

signallist SL_CMGR_SNPROT =
mduAllocReq,
mduAllocCompReq,
mduDeallocReq,
mduModifyReq,
mduModifyCompReq,
mduModifyAbortReq,
mduAddBranchReq,
mduUpdateBranchReq,
mduDropBranchReq,
mduBbccResetReq;

signallist SL_SNPROT_CMGR =
mduAllocAccConf,
mduAllocRejConf,
mduAllocCompAccConf,
mduAllocCompRejConf,
mduDeallocAccConf,
mduDeallocRejConf,
mduModifyAccConf,
mduModifyRejConf,
mduModifyCompAccConf,
mduModifyCompRejConf,
mduModifyAbortAccConf,
mduModifyAbortRejConf,
mduAddBranchAccConf,
mduAddBranchRejConf,
mduUpdateBranchAccConf,
mduUpdateBranchRejConf,
mduDropBranchAccConf,
mduDropBranchRejConf,
mduBbccResetAccConf,
mduBbccResetRejConf,
mduAnFaultInd;

signallist SL_CONN_SNPROT =
(SL_CMGR_SNPROT);

signallist SL_SNPROT_CONN =
(SL_SNPROT_CMGR);

signallist SL_SNMGT_CMGR =
(SL_SNMGT_CONN);

signallist SL_CMGR_SNMGT =
(SL_CONN_SNMGT);



Package VB5_2

20(22)



```
/*Declaration of signallists
from/to block SN_BCCPROT */
```

```
signallist SL_SNPROT_VB5 =
ALLOC,
ALLOC_COMP,
AN_FAULT_ACC,
BBCC_RESET,
PRESYNC,
DEALLOC,
MODIFY,
MODIFY_COMP,
MODIFY_ABORT,
ADD_BRANCH,
UPDATE_BRANCH,
DROP_BRANCH,
PROTOCOL_ERROR,
unknown_protdisc,
unknown_message,
msg_too_short,
trans_id_err,
simSnStarted;
```

```
signallist SL_VB5_SNPROT =
ALLOC_ACC,
ALLOC_REJ,
ALLOC_COMP_ACC,
ALLOC_COMP_REJ,
AN_FAULT,
BBCC_RESET_ACC,
BBCC_RESET_REJ,
PRESYNC_ACC,
PRESYNC_REJ,
DEALLOC_ACC,
MODIFY_ACC,
MODIFY_REJ,
MODIFY_COMP_ACC,
MODIFY_COMP_REJ,
MODIFY_ABORT_ACC,
MODIFY_ABORT_REJ,
ADD_BRANCH_ACC,
ADD_BRANCH_REJ,
UPDATE_BRANCH_ACC,
UPDATE_BRANCH_REJ,
DROP_BRANCH_ACC,
DROP_BRANCH_REJ,
PROTOCOL_ERROR,
unknown_protdisc,
unknown_message,
msg_too_short,
trans_id_err,
simAnStarted;
```



Package VB5_2

21(22)



```
/*Declaration of signallists,
common for block type AN and SN */
```

```
signallist SL_SAAL_MGT =
(SL_SAAL_CTR);
```

```
signallist SL_SAAL_SYS =
(SL_SAAL_MGT);
```

```
signallist SL_MGT_SAAL =
(SL_CTR_SAAL);
```

```
signallist SL_SYS_SAAL =
(SL_MGT_SAAL);
```

```
/*Declaration of signallists,
for block type AN*/
```

```
signallist SL_MF_ANMGT =
(SL_MF_ANCTR,
SL_MF_ANPRESYNC);
```

```
signallist SL_MF_ANSYS =
(SL_MF_ANMGT);
```

```
signallist SL_VB5_ANSYS =
(SL_VB5_ANPROT);
```

```
signallist SL_ANMGT_MF =
(SL_ANCTR_MF,
SL_ANPRESYNC_MF);
```

```
signallist SL_ANSYS_MF =
(SL_ANMGT_MF);
```

```
signallist SL_ANSYS_VB5 =
(SL_ANPROT_VB5);
```

```
signallist SL_ANSYS_CCF =
(SL_ANCONN_CCF);
```

```
signallist SL_CCF_ANSYS =
(SL_CCF_ANCONN);
```

```
/*Declaration of signallists,
for block type SN*/
```

```
signallist SL_MF_SNMGT =
(SL_MF_SNCTR,
SL_MF_SNPRESYNC);
```

```
signallist SL_MF_SNSYS =
(SL_MF_SNMGT);
```

```
signallist SL_VB5_SNSYS =
(SL_VB5_SNPROT);
```

```
signallist SL_SNMGT_MF =
(SL_SNCTR_MF,
SL_SNPRESYNC_MF);
```

```
signallist SL_SNSYS_MF =
(SL_SNMGT_MF);
```

```
signallist SL_SNSYS_VB5 =
(SL_SNPROT_VB5);
```

```
signallist SL_SNSYS_CCF =
(SL_SNCONN_CCF);
```

```
signallist SL_CCF_SNSYS =
(SL_CCF_SNCONN);
```

Package VB5_2

22(22)

/* Declaration of common primitives internal
to AN_BCCPROT and SN_BCCPROT */

```

signal
bccAlloc      (IT_TransId, IT_ConnRefNo, IT_BranchId),
bccAllocAcc   (IT_TransId, IT_BranchId),
bccAllocRej   (IT_TransId, IT_RejCause),
bccAllocComp  (IT_TransId, IT_ConnRefNo),
bccAllocCompAcc (IT_TransId),
bccAllocCompRej (IT_TransId, IT_RejCause),
bccBbccReset  (IT_TransId),
bccBbccResetAcc (IT_TransId),
bccBbccResetRej (IT_TransId, IT_RejCause),
bccPresync    (IT_TransId),
bccPresyncAcc (IT_TransId),
bccPresyncRej (IT_TransId),
bccDealloc    (IT_TransId, IT_ConnRefNo),
bccDeallocAcc (IT_TransId),
bccModify     (IT_TransId, IT_ConnRefNo),
bccModifyAcc  (IT_TransId),
bccModifyRej  (IT_TransId, IT_RejCause),
bccModifyComp (IT_TransId, IT_ConnRefNo),
bccModifyCompAcc (IT_TransId),
bccModifyCompRej (IT_TransId, IT_RejCause),
bccModifyAbort (IT_TransId, IT_ConnRefNo),
bccModifyAbortAcc (IT_TransId),
bccModifyAbortRej (IT_TransId, IT_RejCause),
bccAddBranch  (IT_TransId, IT_ConnRefNo, IT_BranchId),
bccAddBranchAcc (IT_TransId),
bccAddBranchRej (IT_TransId, IT_RejCause),
bccUpdateBranch (IT_TransId, IT_ConnRefNo, IT_BranchId),
bccUpdateBranchAcc (IT_TransId),
bccUpdateBranchRej (IT_TransId, IT_RejCause),
bccDropBranch (IT_TransId, IT_ConnRefNo, IT_BranchId),
bccDropBranchAcc (IT_TransId),
bccDropBranchRej (IT_TransId, IT_RejCause),
bccAnFault    (IT_TransId, IT_ConnRefNo, IT_BranchId),
bccAnFaultAcc (IT_TransId),
bccProtocolError (IT_TransId, IT_ProtErrCause),
bccUnknownMessage (IT_TransId),
bccUnknownProtdisc,
bccMsgTooShort,
bccTransIdErr,
mieProtocolError (IT_TransId, IT_ProtErrCause),
mieAbortReq,
mieErrorInd;

```

/* Declaration of common primitives internal
to AN_BCCMG and SN_BCCMG */

```

signal
mieStartTraffic,
mieStopTraffic;

```


Procedure IE_CHECK

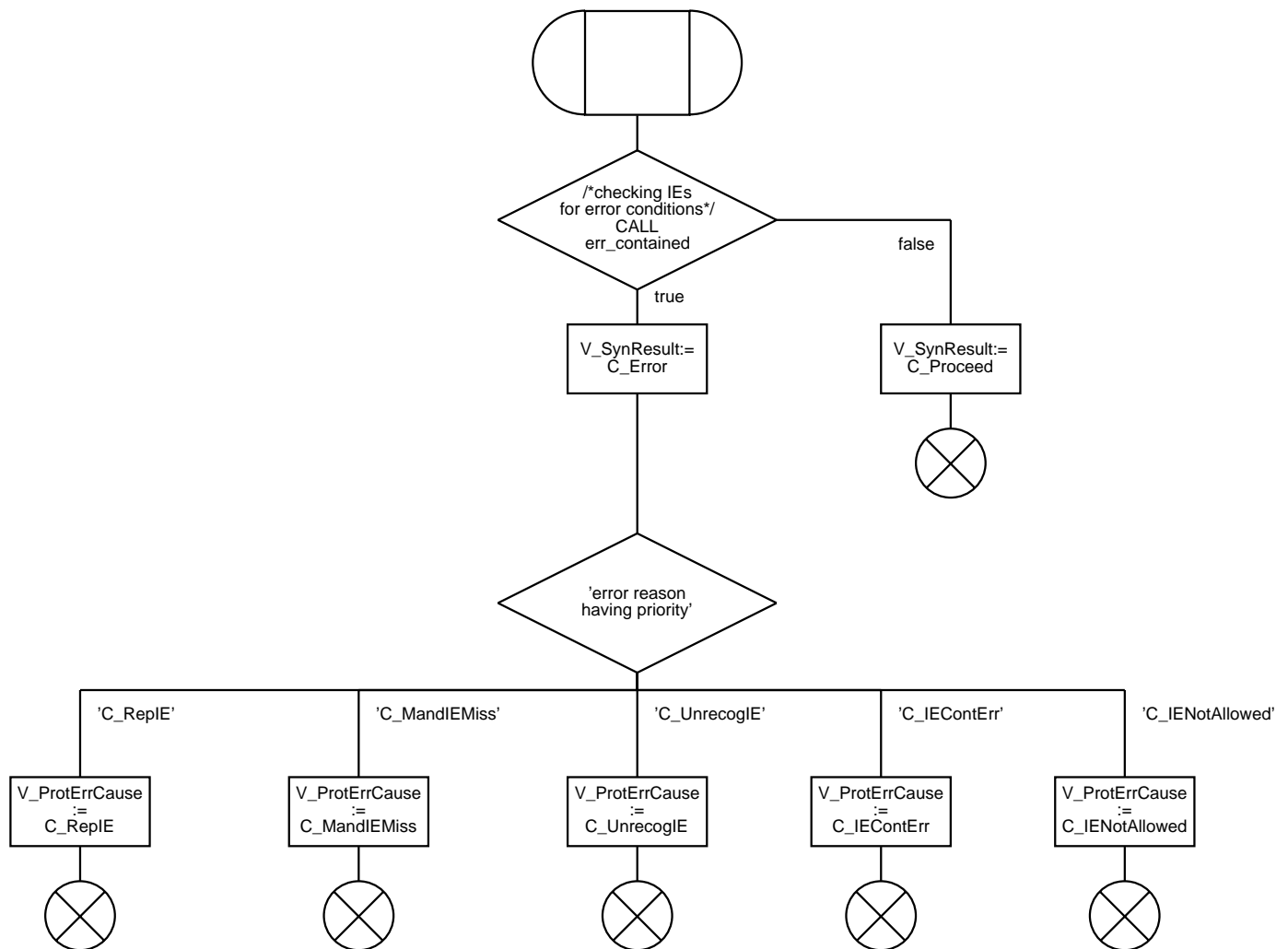
1(1)

```

FPAR
IN/OUT V_ProtErrCause IT_ProtErrCause;
RETURNS V_SynResult INTEGER;

```

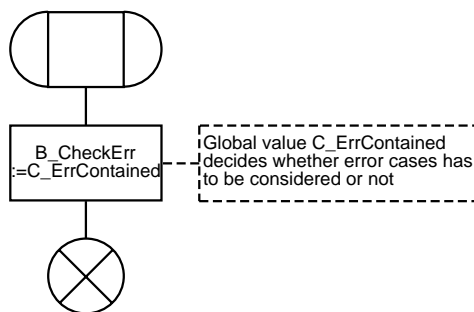
/* Procedure used only for simulation purposes. Supports the syntax error checking of IEs. The procedure checks first whether error checking is activated at all via the external synonym C_ErrContained. If error checking is activated, then the possible protocol error cause (see subclause 14.3.7.10) can be set by direct input during simulation. If error checking is de-activated, the normal procedures apply. */



Procedure ERR_CONTAINED

1(1)

```
RETURNS  
B_CheckErr BOOLEAN;  
/*Procedure only for  
simulation use. Decides whether  
error cases have to be considered  
or not*/
```



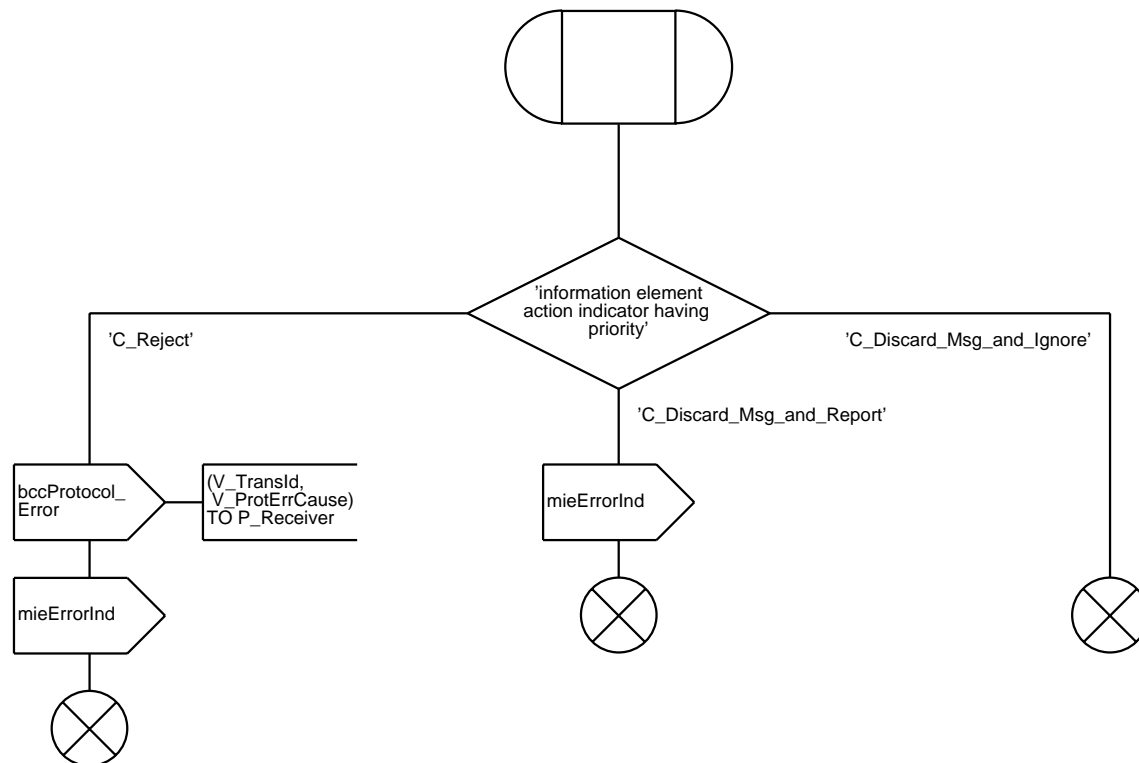
Procedure IE_ERROR_HANDLING

1(1)

FPAR
 V_TransId IT_TransId,
 V_ProtErrCause IT_ProtErrCause,
 P_Receiver PID;

/* Procedure used only for simulation purposes. The possible actions for handling of an IE syntax error (see table 36) can be set by direct input during simulation. */

/* The possible actions 'Discard IE and proceed' and 'Discard IE, proceed and report' are not included in the simulation for simplification reasons but shall be included in an implementation */



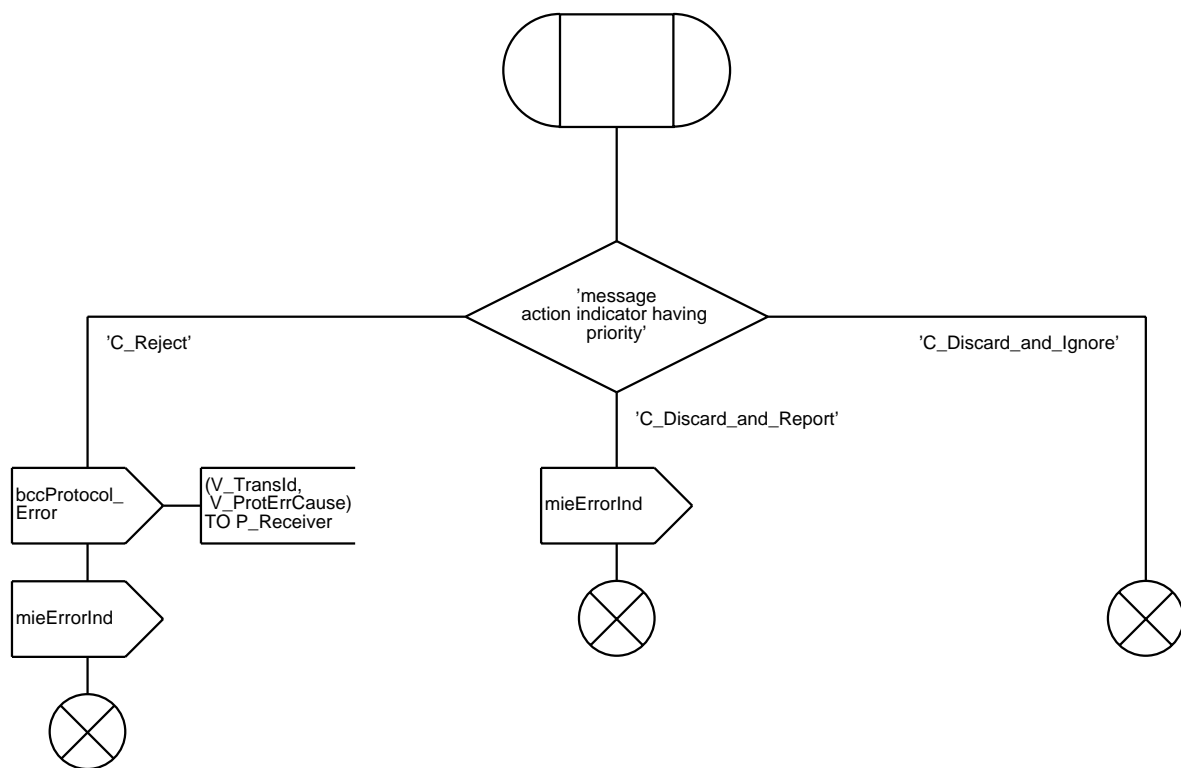
Procedure MSG_ERROR_HANDLING

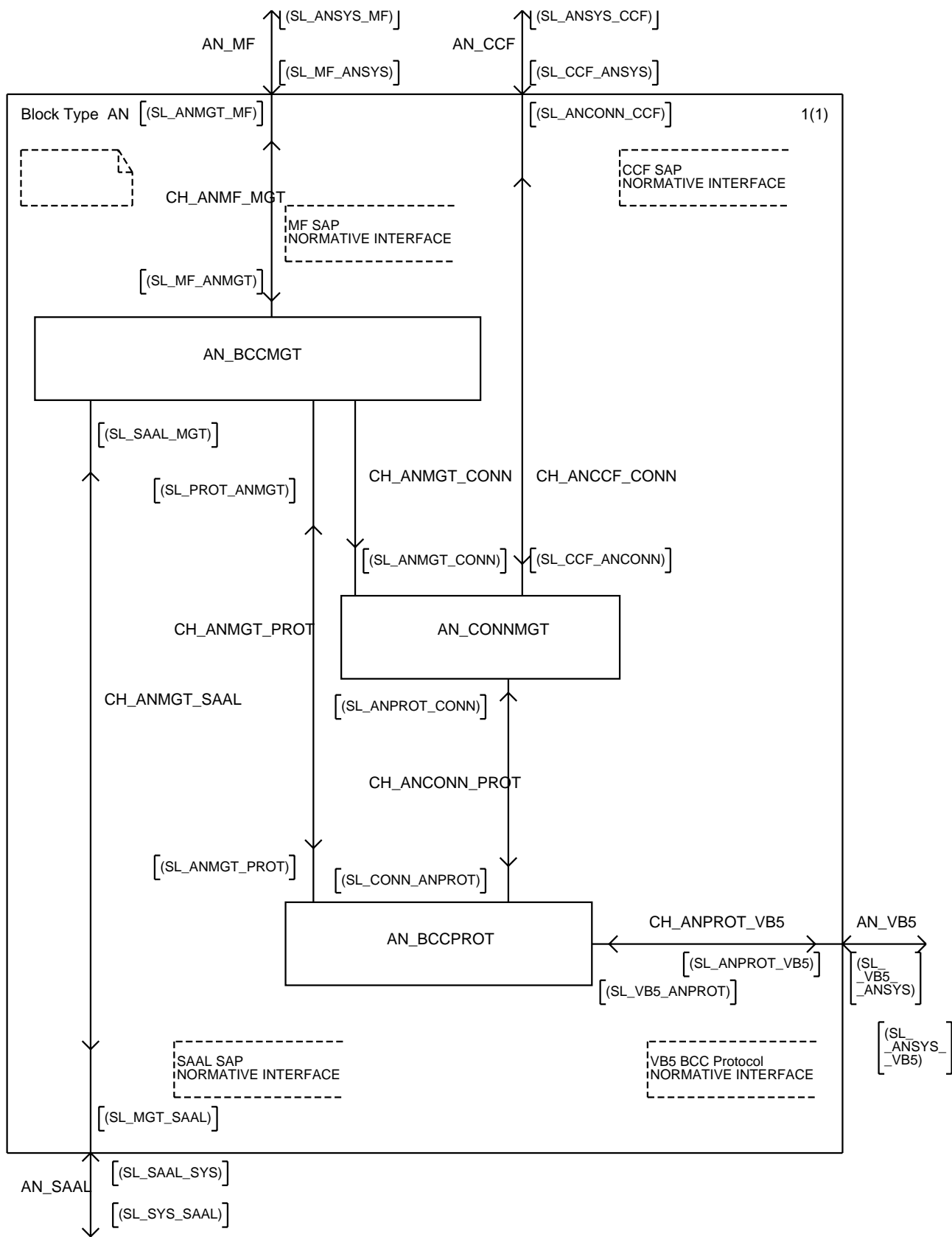
1(1)

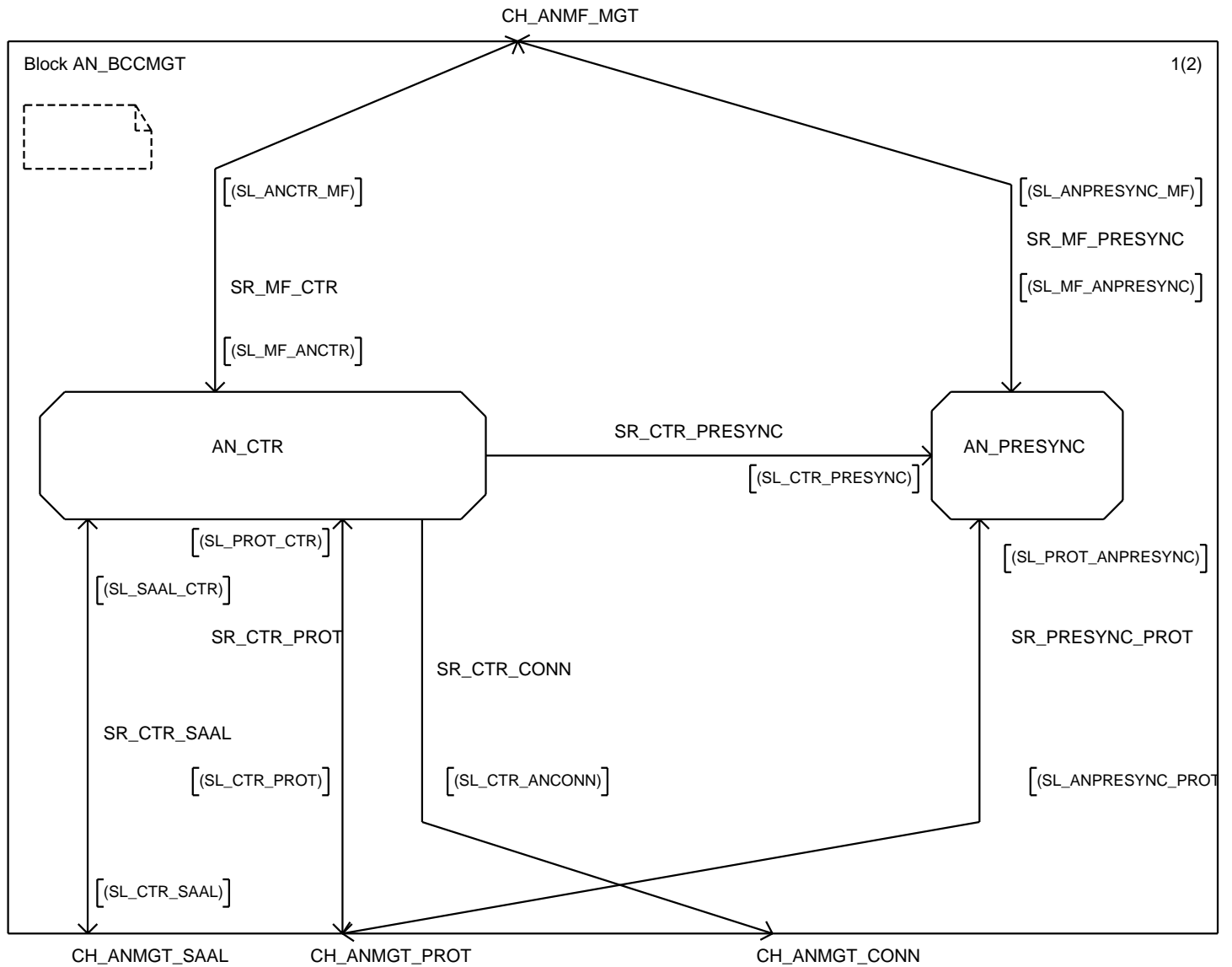
```

FPAR
V_TransId      IT_TransId,
V_ProtErrCause IT_ProtErrCause,
P_Receiver     PID;
  
```

/* Procedure used only for simulation purposes. The possible actions for handling of syntax errors in the common message information fields (see table 35) can be set by direct input during simulation. */







Block AN_BCCMGT

2(2)



```
/* Declaration of signallists  
internal to block SN_CONNMGT */
```

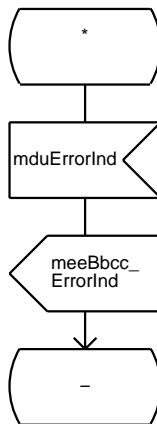
```
signallist SL_CTR_PRESYNC =  
mieStartTraffic,  
mieStopTraffic;
```

Process AN_CTR

1(5)

/* VB5.2 BCC management entity
Process CTR */

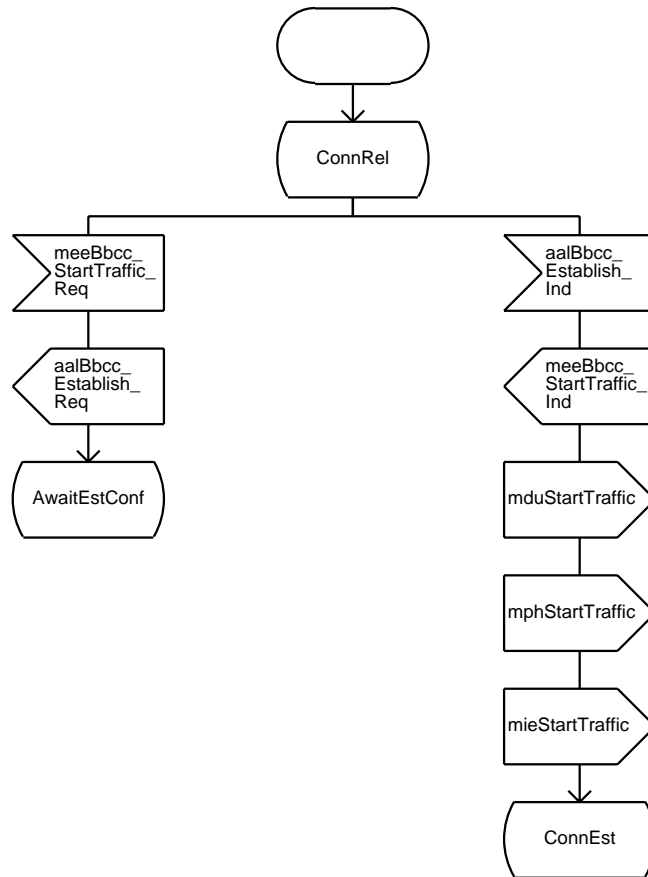
/* State descriptions
ConnRel: initial state, no SAAL connection available.
AwaitEstConf: process is waiting for establish confirmation of the peer entity.
AwaitRelConf: process is waiting for release confirmation of the peer entity.
ConnEst: SAAL in operation. */



Process AN_CTR

2(5)

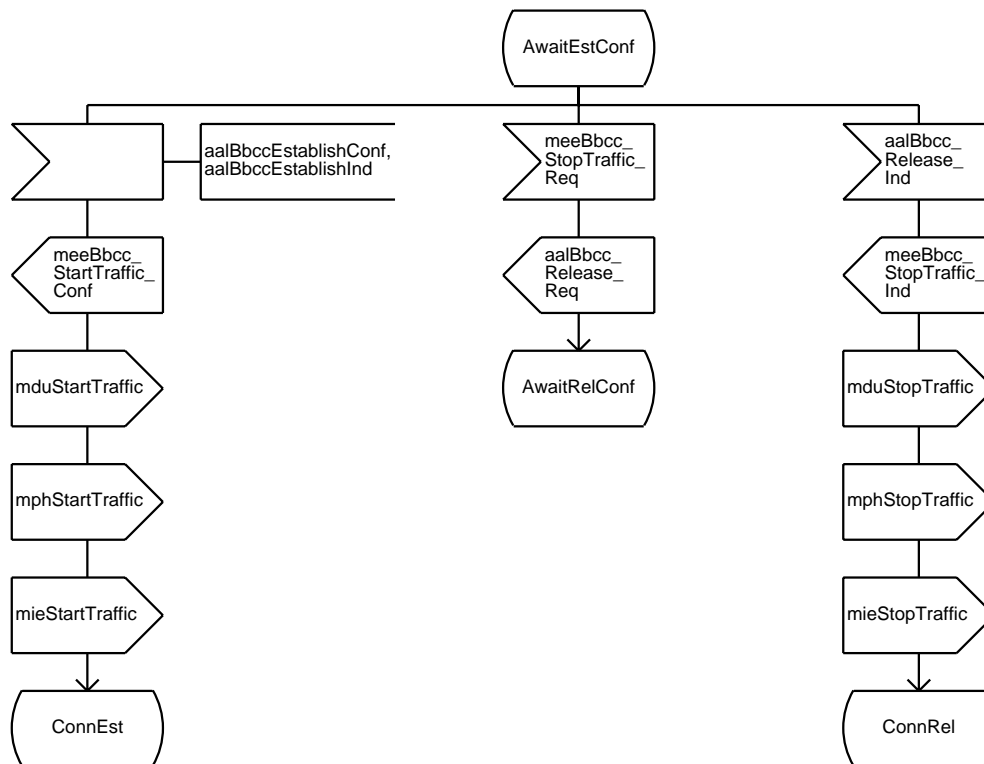
/* VB5.2 BCC management entity
Process CTR */



Process AN_CTR

3(5)

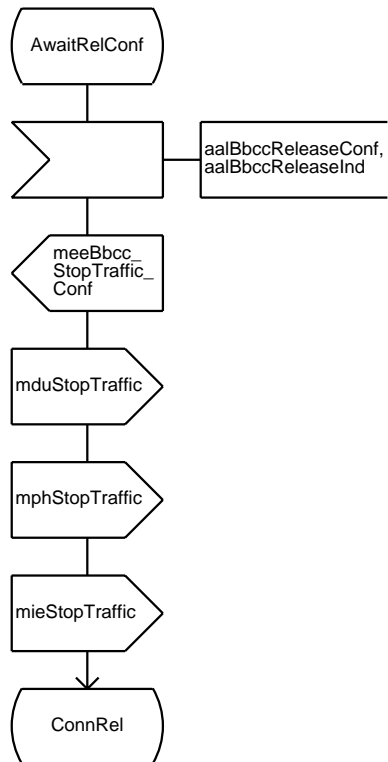
/* VB5.2 BCC management entity
Process CTR */



Process AN_CTR

4(5)

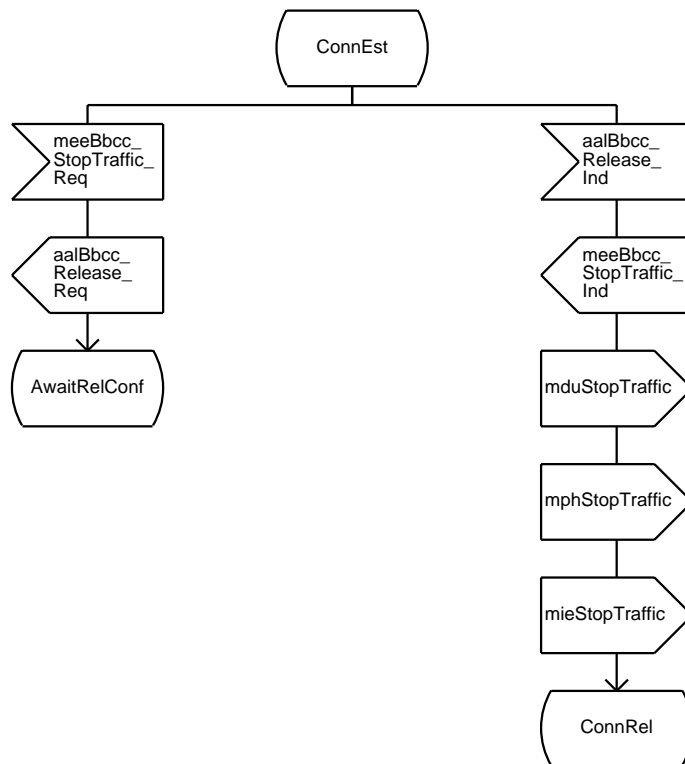
/* VB5.2 BCC management entity
Process CTR */



Process AN_CTR

5(5)

/* VB5.2 BCC management entity
Process CTR */



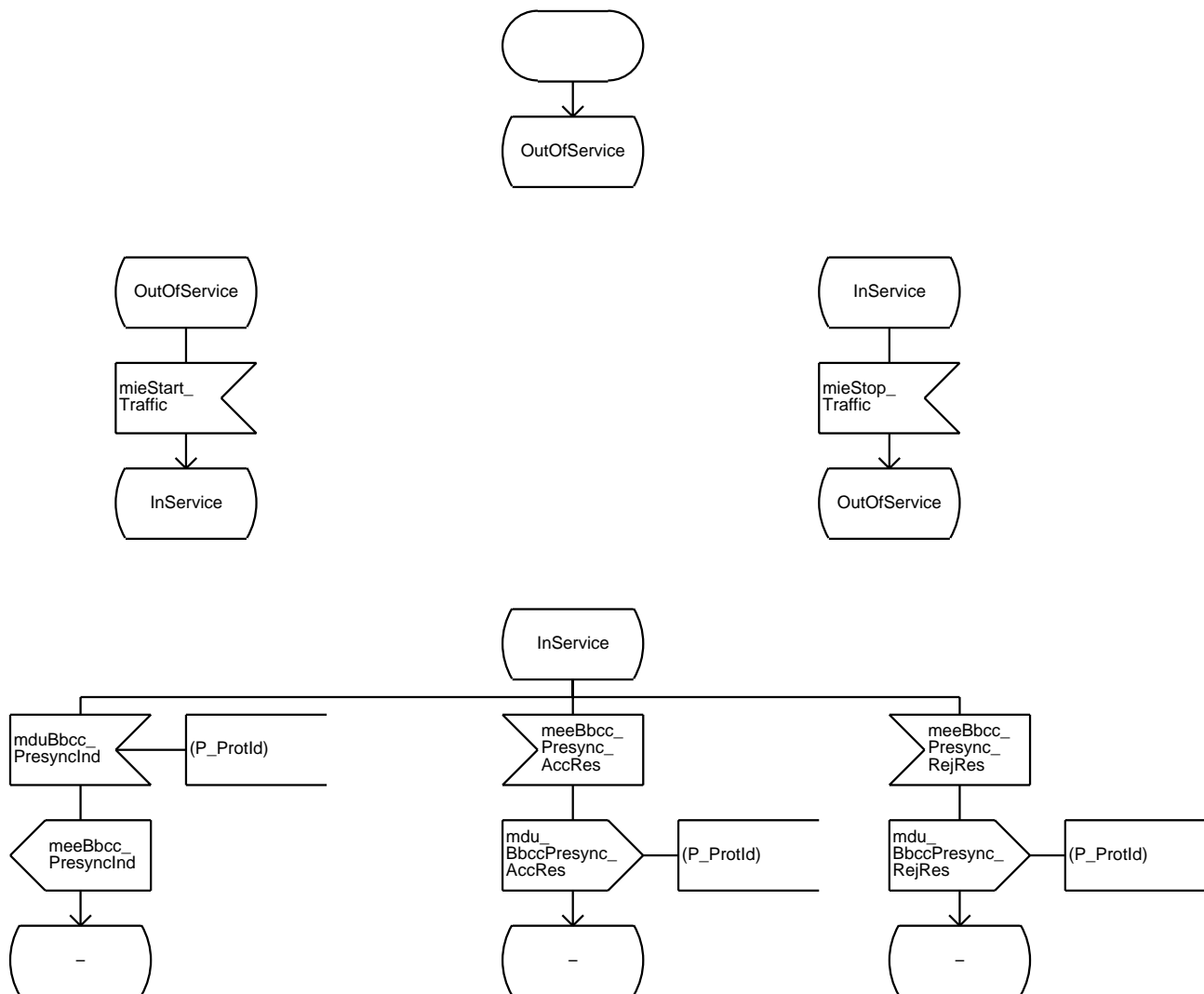
Process AN_PRESYNC

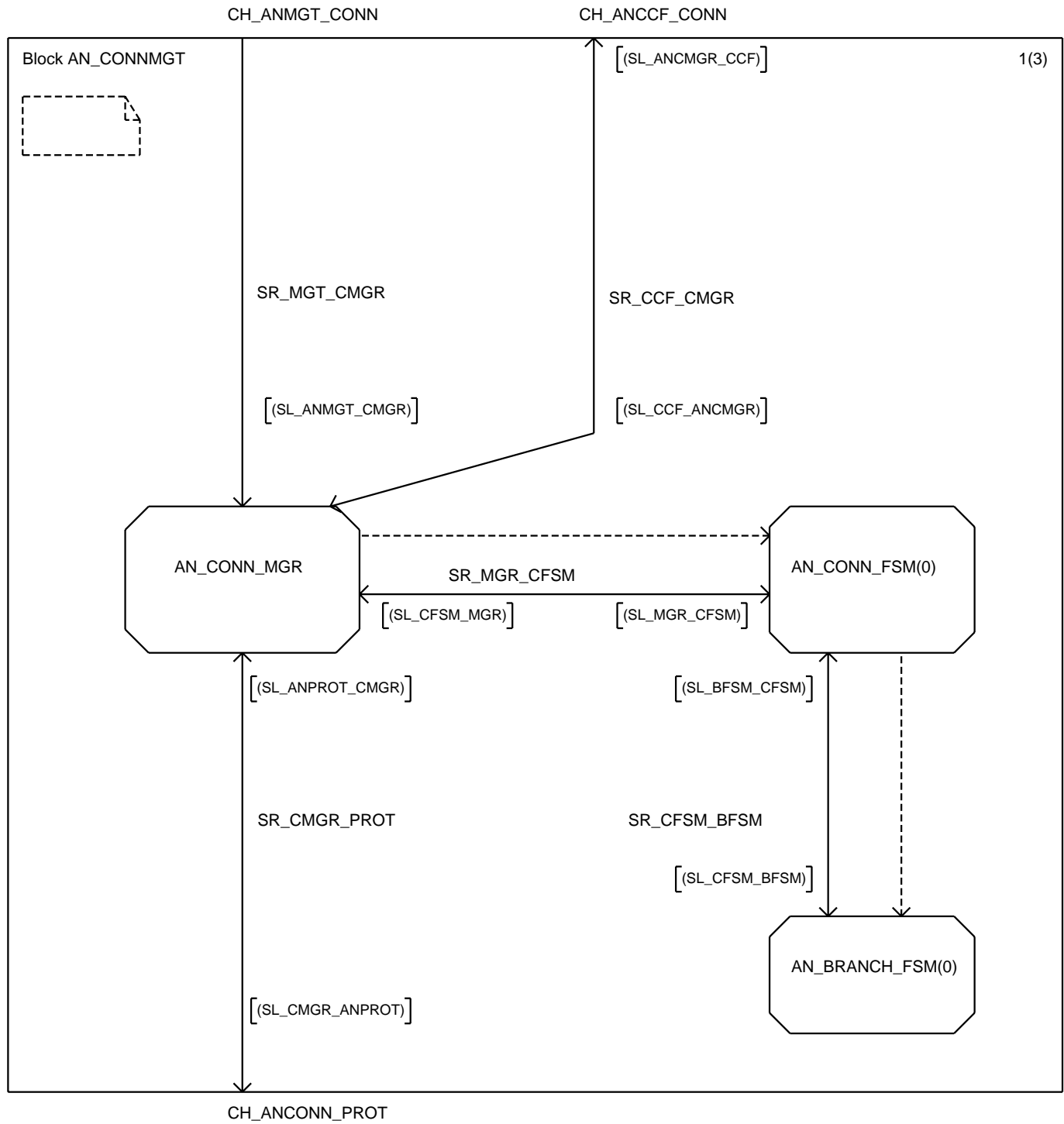
1(1)

```
/* Data type definitions internal to the process */
/**/
```

```
/* Primitive Data Declarations */
/**/
/* Protocol process identifier */
DCL
P_ProtId PT_ProtId;
```

```
/* Process Data Declarations */
/**/
```





Block AN_CONNMG

2(3)



/* Declaration of primitives internal
to block AN_CONNMG, between AN_CONN_MGR
and AN_CONN_FSM*/

```
signal
mieAllocCompInd (IT_ConnRefNo,
mieAllocCompRes (IT_ConnRefNo, IT_RejCause),
mieModifyInd (IT_ConnRefNo),
mieModifyRes (IT_ConnRefNo, IT_RejCause),
mieModifyCompInd (IT_ConnRefNo),
mieModifyCompRes (IT_ConnRefNo, IT_RejCause),
mieModifyAbortInd (IT_ConnRefNo),
mieModifyAbortRes (IT_ConnRefNo, IT_RejCause),
mieAddBranchInd (IT_ConnRefNo, IT_BranchId),
mieAddBranchRes (IT_ConnRefNo, IT_BranchId, IT_RejCause),
mieUpdateBranchInd (IT_ConnRefNo, IT_BranchId),
mieUpdateBranchRes (IT_ConnRefNo, IT_BranchId, IT_RejCause),
mieDropBranchInd (IT_ConnRefNo, IT_BranchId),
mieDropBranchRes (IT_ConnRefNo, IT_BranchId, IT_RejCause),
mieDeallocInd (IT_ConnRefNo),
mieDeallocRes (IT_ConnRefNo),
mieAnFaultReq (IT_ConnRefNo, IT_BranchId);
```

```
signal
mieAllocAccInd (IT_ConnRefNo, IT_BranchId),
mieAllocRejInd (IT_BranchId),
mieAllocCompAccInd,
mieAllocCompRejInd,
mieDeallocAccInd (IT_ConnRefNo),
mieDeallocAccRes (IT_ConnRefNo),
mieModifyAccInd,
mieModifyRejInd,
mieModifyCompAccInd,
mieModifyCompRejInd,
mieModifyAbortAccInd,
mieModifyAbortRejInd,
mieAddBranchAccInd (IT_BranchId, IT_UpdateNeeded),
mieAddBranchRejInd (IT_BranchId),
mieUpdateBranchAccInd (IT_BranchId),
mieUpdateBranchRejInd (IT_BranchId),
mieDropBranchAccInd (IT_ConnRefNo, IT_BranchId),
mieDropBranchAccRes (IT_ConnRefNo, IT_BranchId),
mieResetInd (IT_BranchId),
mieAnFaultInd (IT_BranchId);
```

/* Declaration of primitives internal
to block AN_CONNMG, between AN_CONN_FSM
and AN_BRANCH_FSM */

```
signal
mieUpdateBranchFSMInd (IT_BranchId),
mieUpdateBranchFSMRes (IT_BranchId, IT_RejCause),
mieDropBranchFSMInd (IT_BranchId),
mieDropBranchFSMRes (IT_BranchId, IT_RejCause),
mieAddBranchFSMAccInd (IT_BranchId, IT_UpdateNeeded),
mieAddBranchFSMRejInd,
mieUpdateBranchFSMAccInd,
mieUpdateBranchFSMRejInd,
mieDropBranchFSMAccInd (IT_BranchId),
mieDropBranchFSMAccRes (IT_BranchId),
mieDelBranchFSMInd,
mieAnFaultFSMInd;
```

Block AN_CONNMGT

3(3)



```
/* Declaration of signallists
   internal to block AN_CONNMGT */
```

```
signallist SL_MGR_CFSM =
mieAllocCompInd,
mieModifyInd,
mieModifyCompInd,
mieModifyAbortInd,
mieAddBranchInd,
mieUpdateBranchInd,
mieDropBranchInd,
mieDeallocInd,
mieAllocAccInd,
mieAllocRejInd,
mieAllocCompAccInd,
mieAllocCompRejInd,
mieDeallocAccInd,
mieModifyAccInd,
mieModifyRejInd,
mieModifyCompAccInd,
mieModifyCompRejInd,
mieModifyAbortAccInd,
mieModifyAbortRejInd,
mieAddBranchAccInd,
mieAddBranchRejInd,
mieUpdateBranchAccInd,
mieUpdateBranchRejInd,
mieDropBranchAccInd,
mieResetInd,
mieAnFaultInd;
```

```
signallist SL_CFSM_MGR =
mieAllocCompRes,
mieModifyRes,
mieModifyCompRes,
mieModifyAbortRes,
mieAddBranchRes,
mieUpdateBranchRes,
mieDropBranchRes,
mieDropBranchAccRes,
mieDeallocRes,
mieDeallocAccRes,
mieAnFaultReq;
```

```
signallist SL_CFSM_BFSM =
mieUpdateBranchFSMInd,
mieDropBranchFSMInd,
mieAddBranchFSMAccInd,
mieAddBranchFSMRejInd,
mieUpdateBranchFSMAccInd,
mieUpdateBranchFSMRejInd,
mieDropBranchFSMAccInd,
mieDelBranchFSMInd,
mieAnFaultFSMInd;
```

```
signallist SL_BFSM_CFSM =
mieUpdateBranchFSMRes,
mieDropBranchFSMRes,
mieDropBranchFSMAccRes;
```


Process AN_CONN_MGR

1(12)



```

/* Data type definitions internal to the process */
/**/
/* This data type is used for translation of the connection reference number to the process identifier (PID)
  of the connection FSM */
NEWTTYPE AT_ConnPIdList ARRAY
(IT_ConnRefNo,PID)
ENDNEWTTYPE AT_ConnPIdList;

```

```

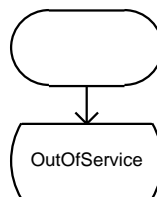
/*Primitive Data Declarations*/
/**/
/* Protocol process identifier */
DCL
P_ProtId PT_ProtId;
/**/
/* List of connections */
DCL
A_ConnList AT_ConnList;
/**/
/* Connection reference number */
DCL
V_ConnRefNo IT_ConnRefNo;
/**/
/* Branch identifier */
DCL
V_BranchId IT_BranchId;
/**/
/* Reject Cause */
DCL
V_RejCause IT_RejCause;
/**/
/* Update needed for addBranch */
DCL
V_UpdateNeeded IT_UpdateNeeded;

```

```

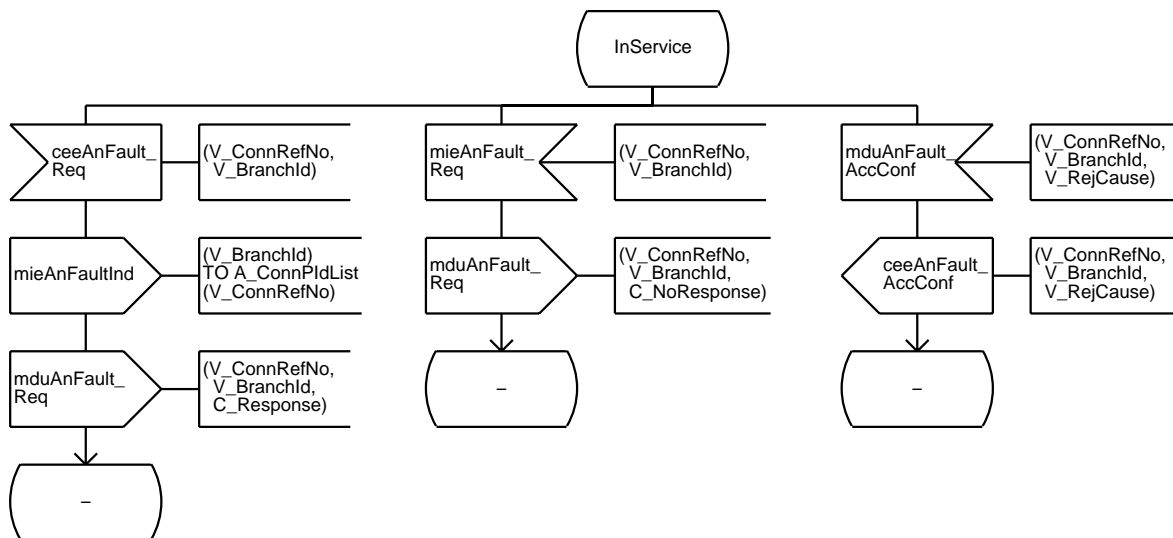
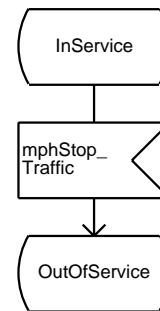
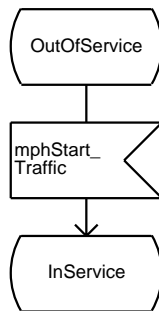
/* Process Data Declarations */
/**/
/* List of connection FSMs */
DCL
A_ConnPIdList AT_ConnPIdList;
/**/
/* Index of A_ConnList */
DCL
V_Element INTEGER;

```



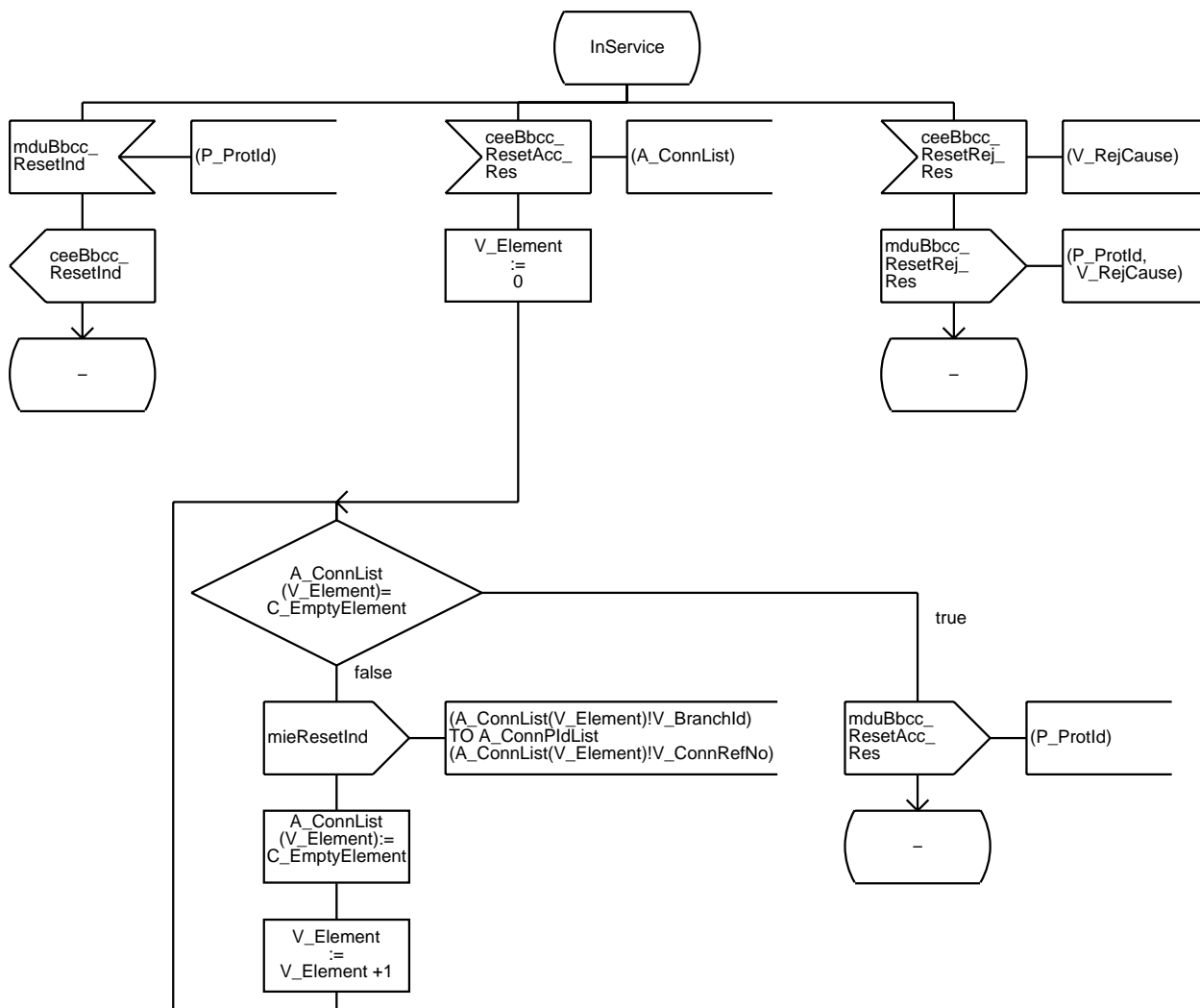
Process AN_CONN_MGR

2(12)



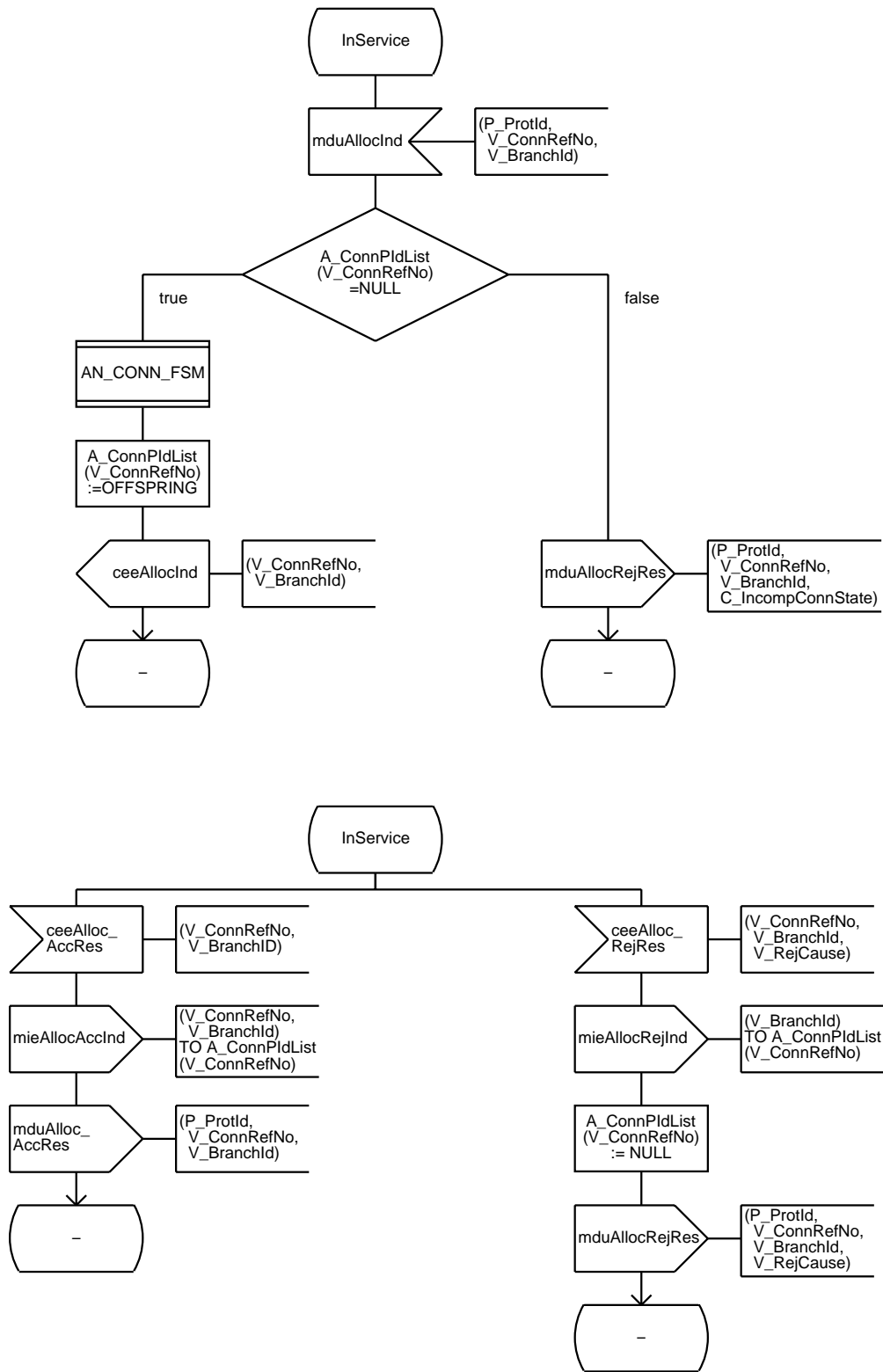
Process AN_CONN_MGR

3(12)



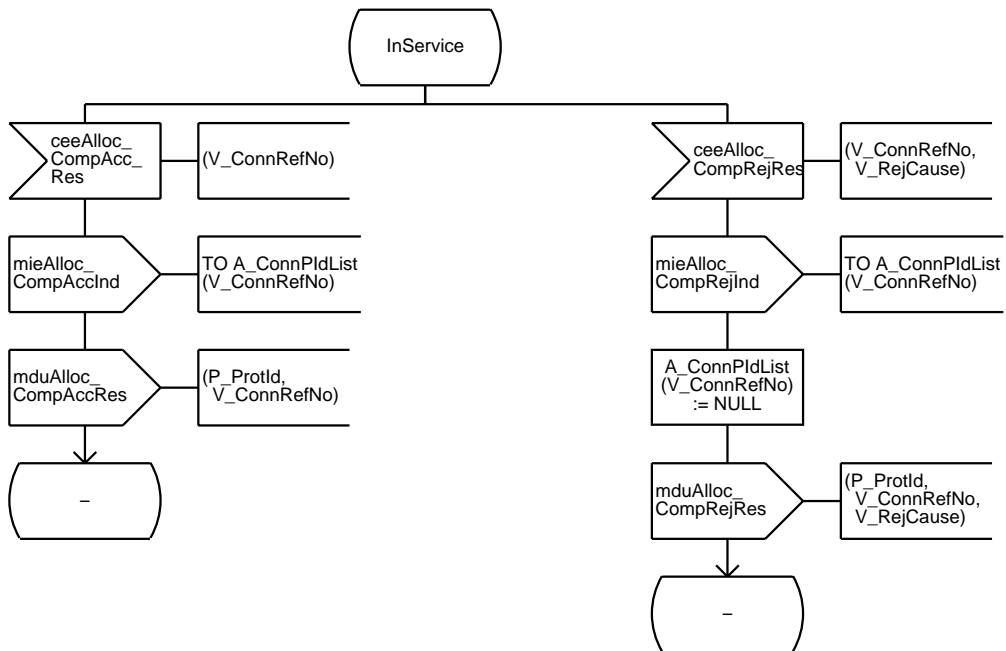
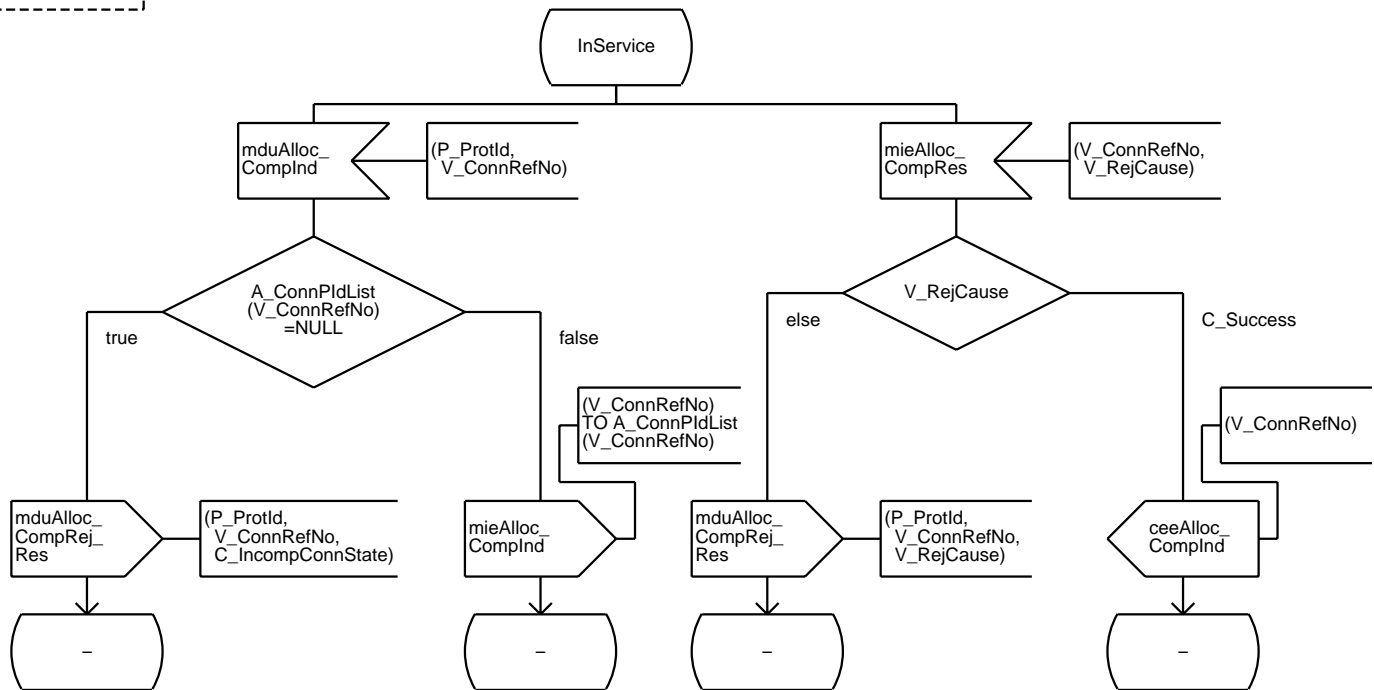
Process AN_CONN_MGR

4(12)



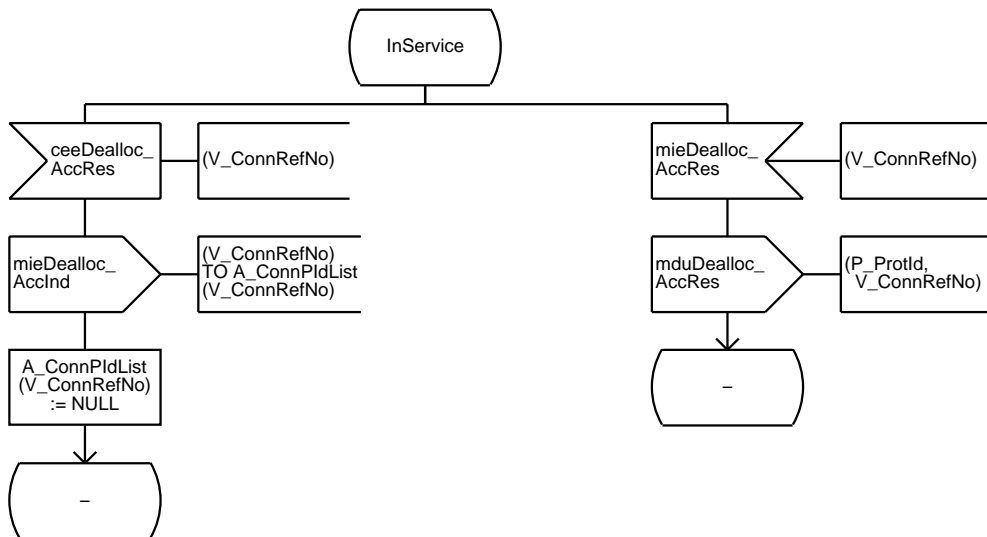
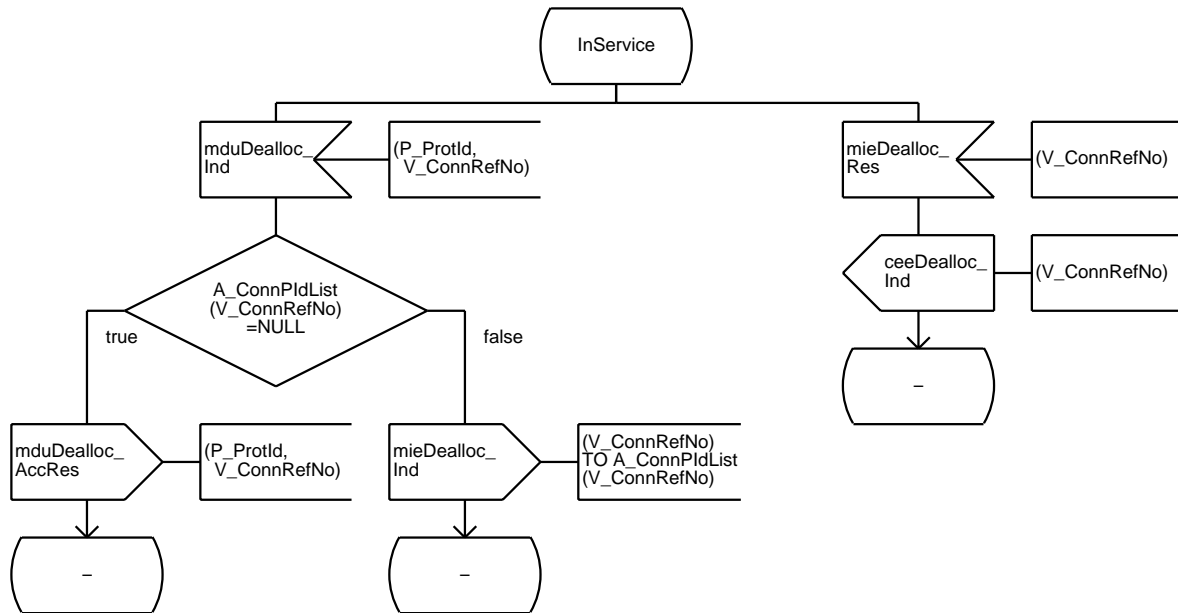
Process AN_CONN_MGR

5(12)



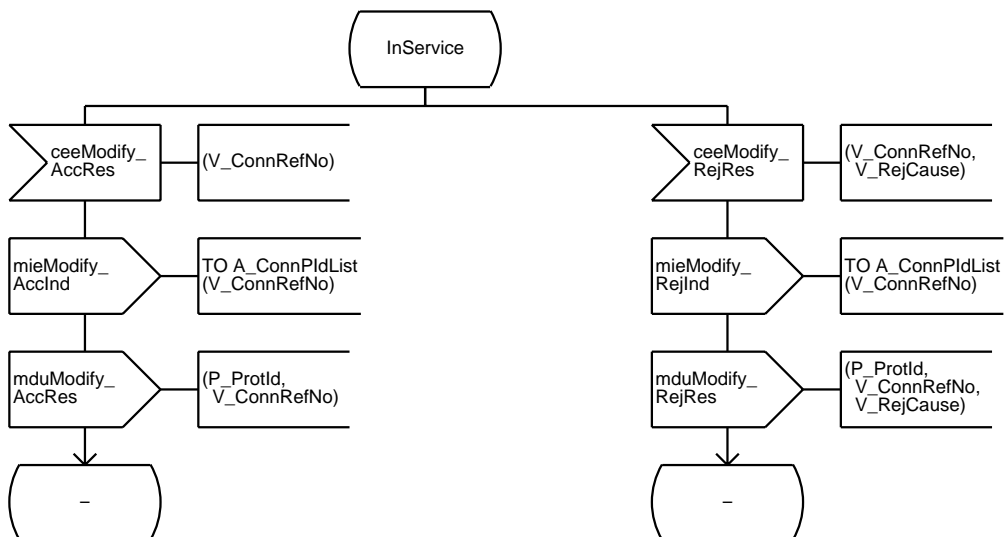
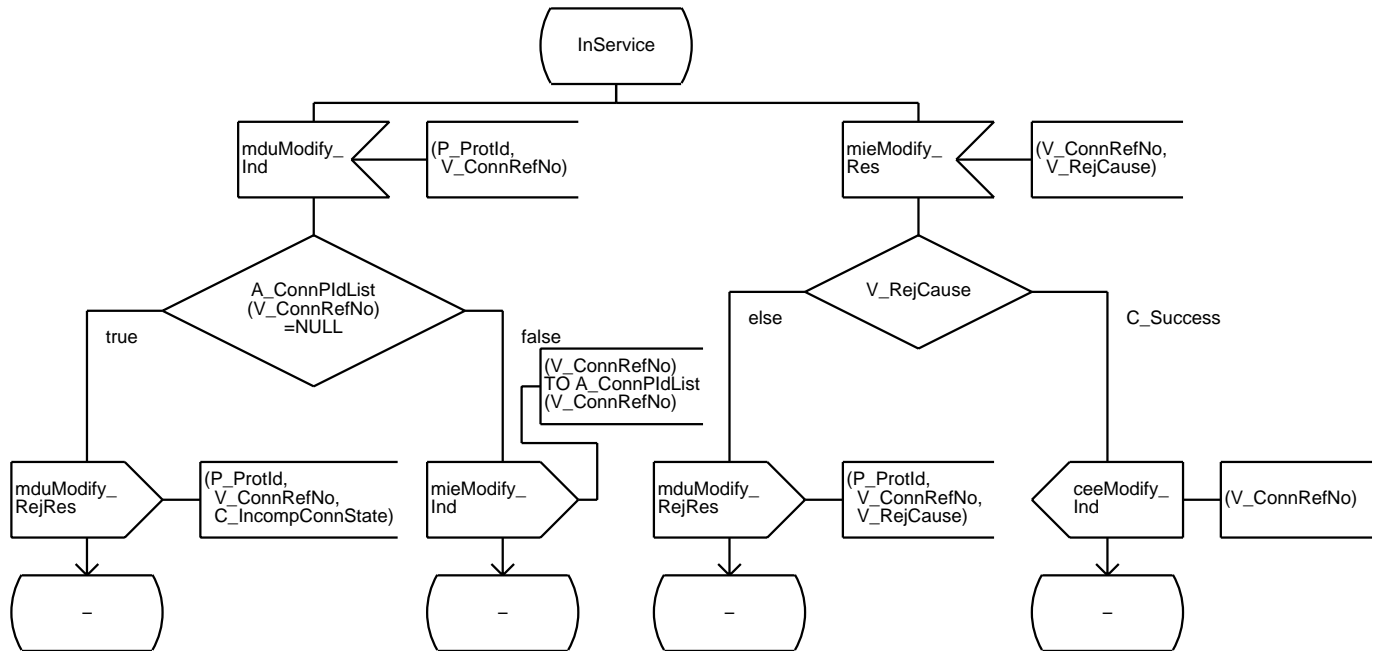
Process AN_CONN_MGR

6(12)



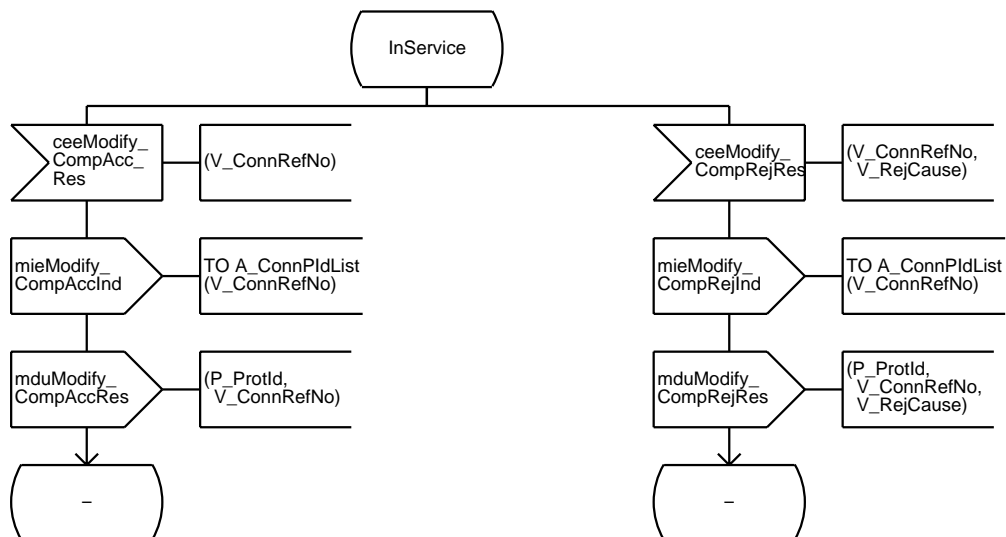
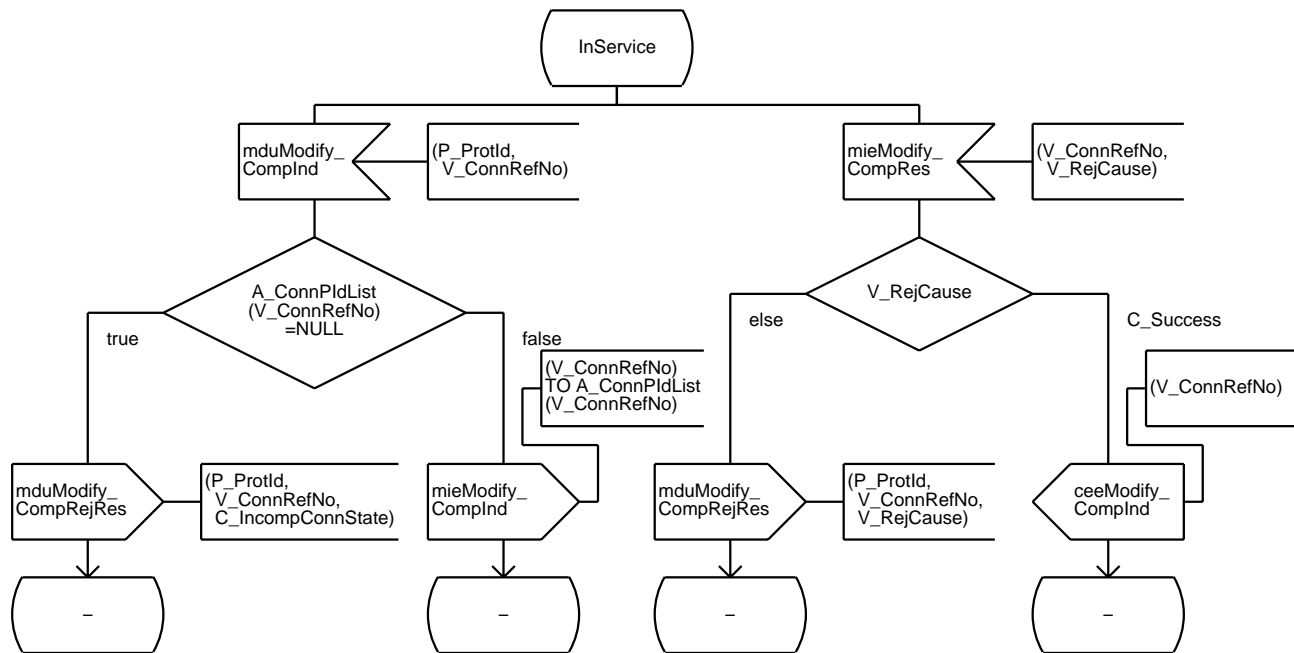
Process AN_CONN_MGR

7(12)



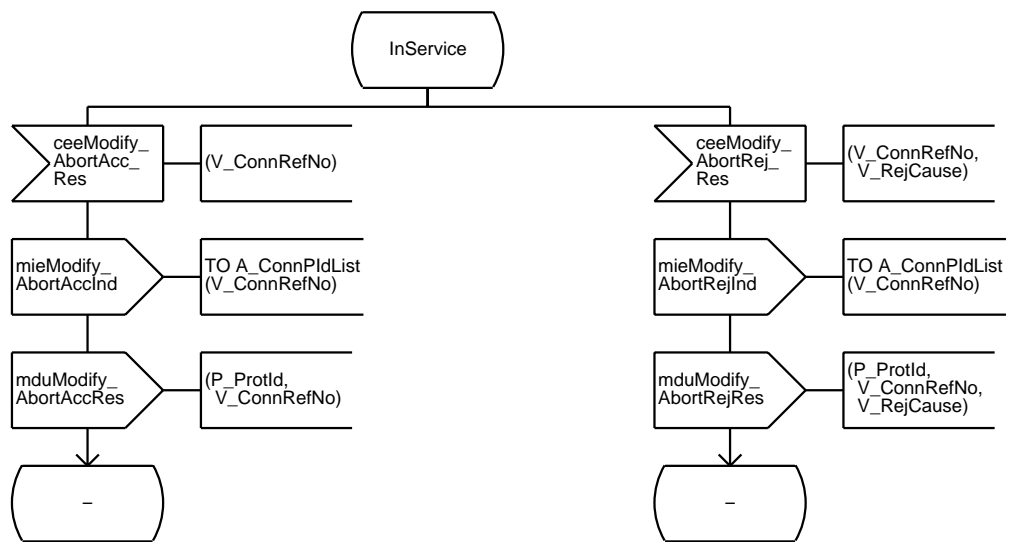
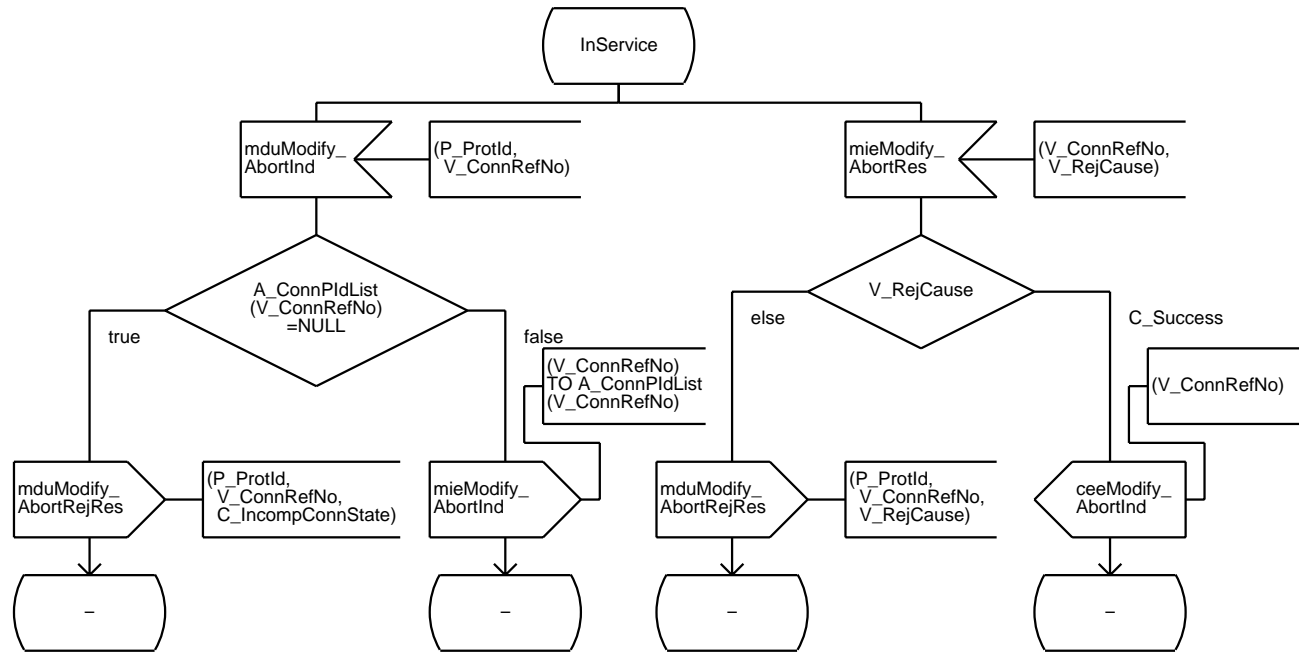
Process AN_CONN_MGR

8(12)



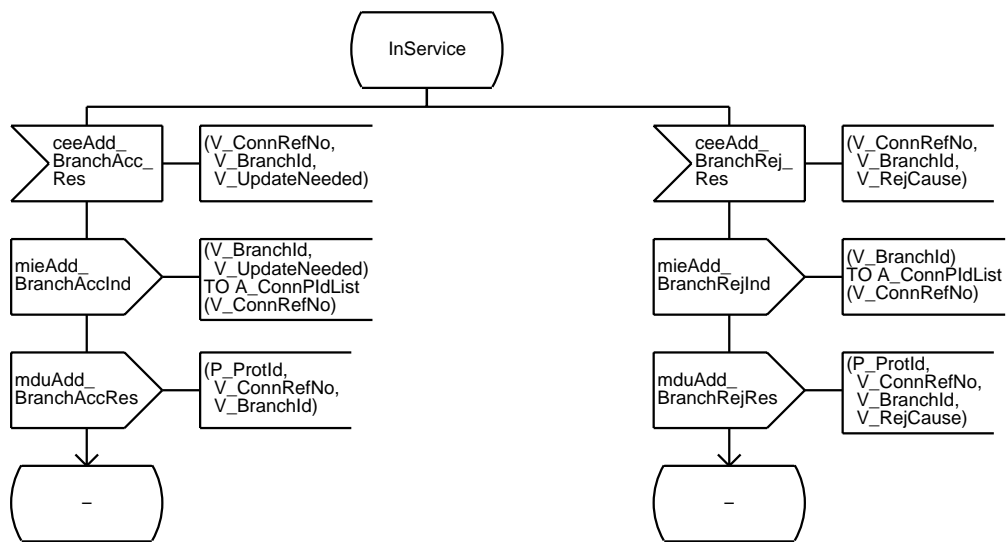
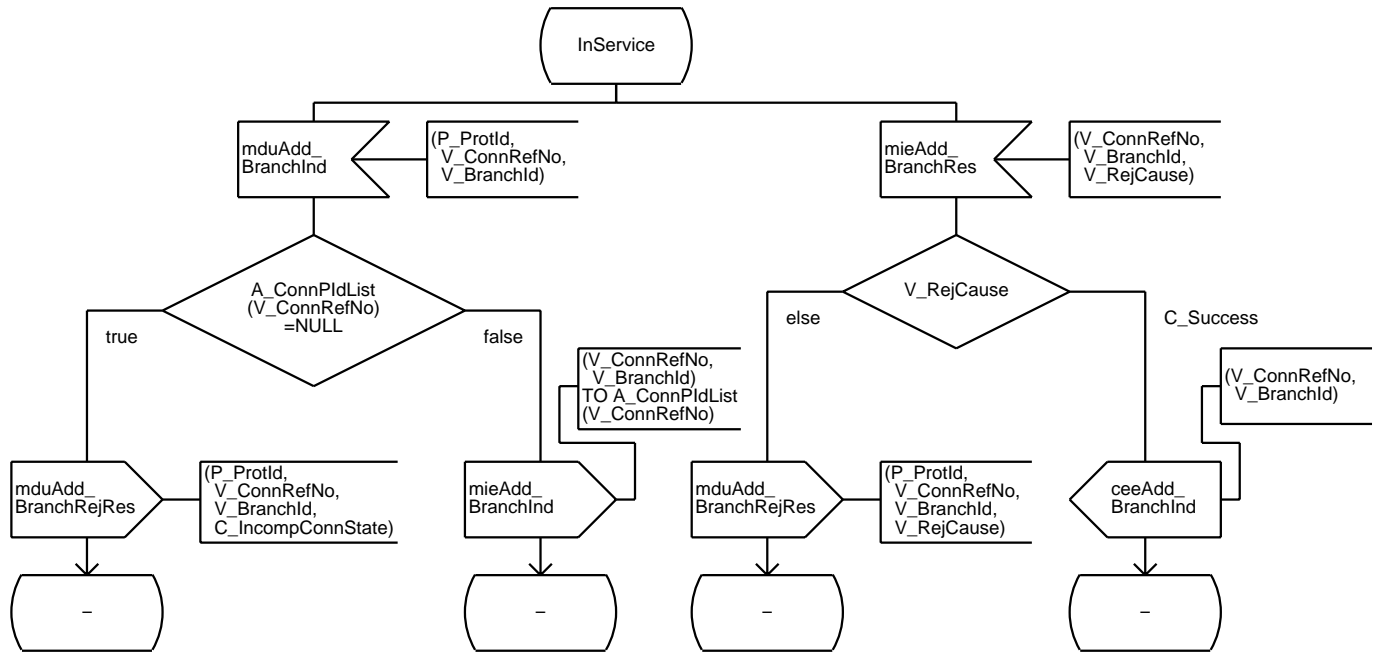
Process AN_CONN_MGR

9(12)



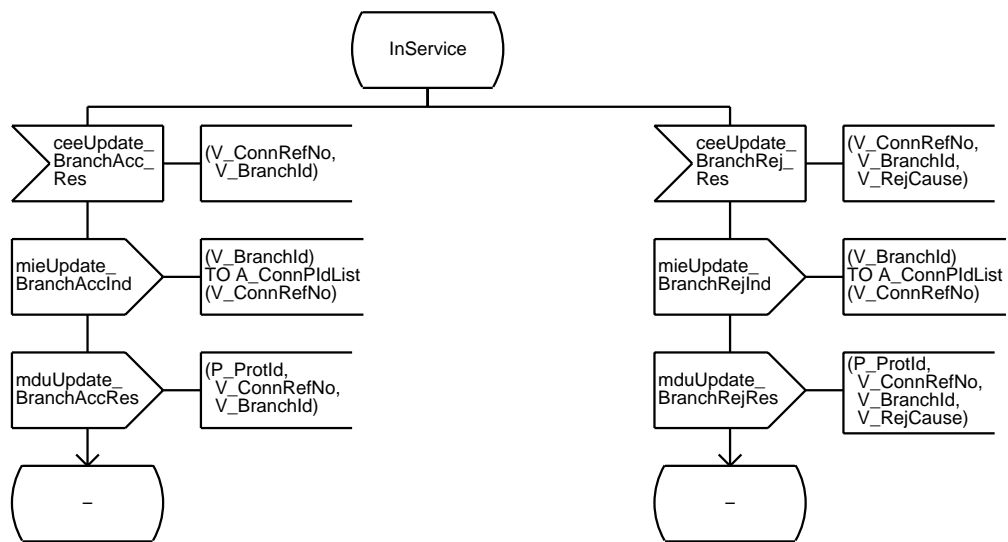
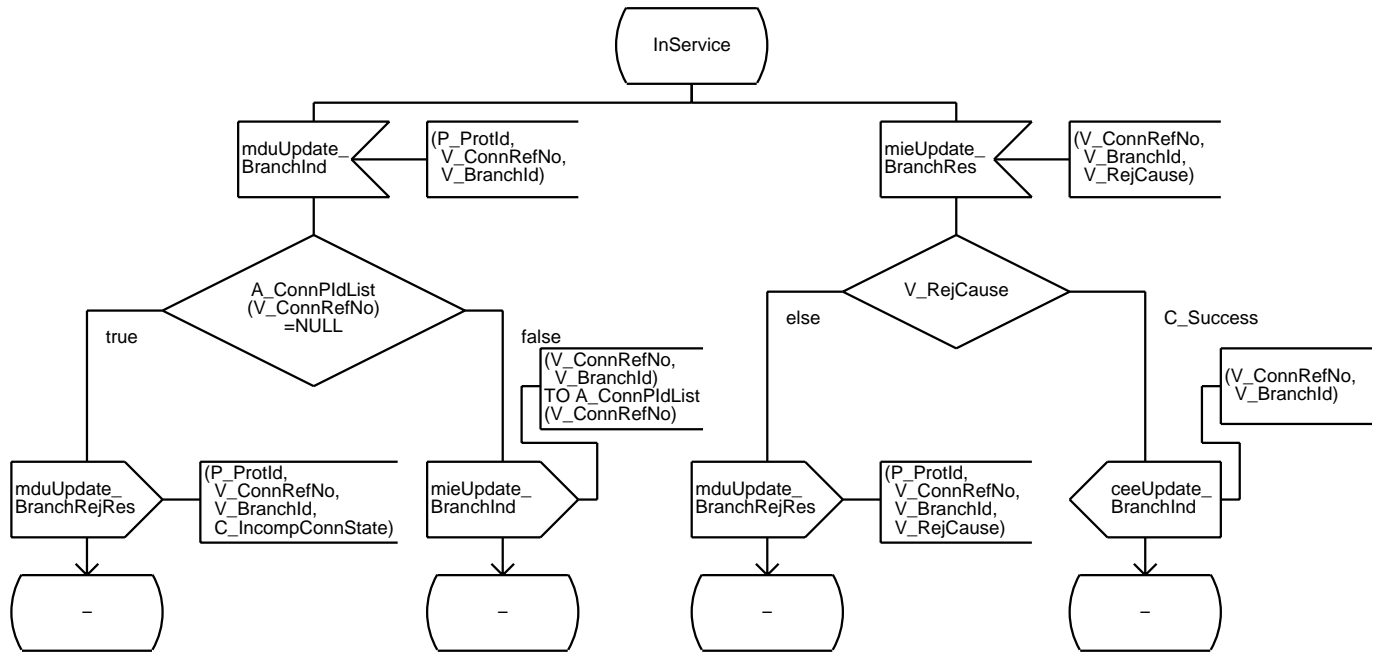
Process AN_CONN_MGR

10(12)



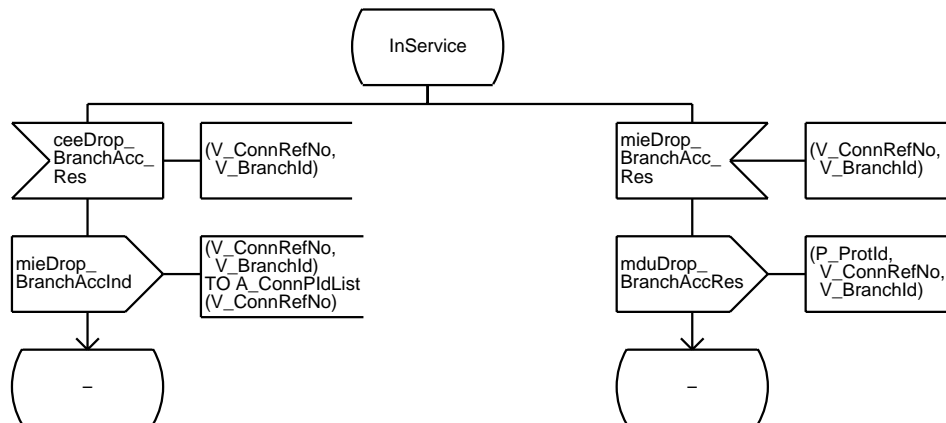
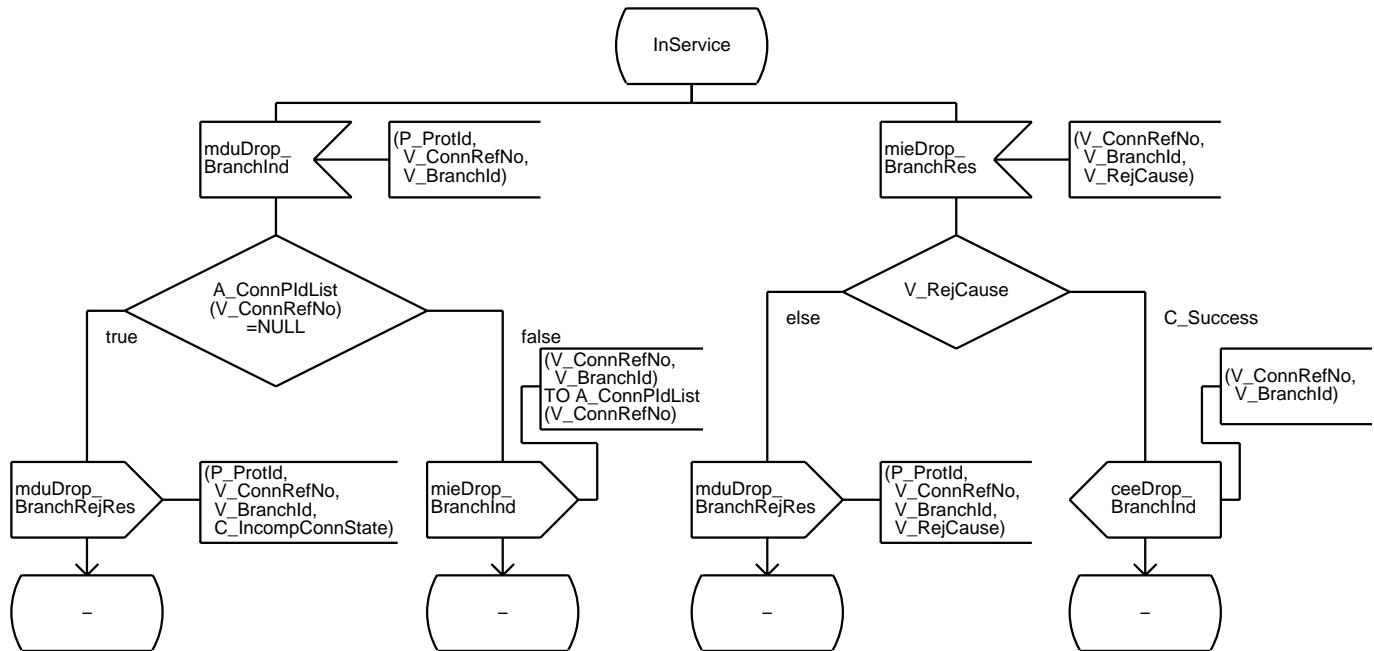
Process AN_CONN_MGR

11(12)



Process AN_CONN_MGR

12(12)



Process AN_CONN_FSM

1(12)



```

/* State descriptions
AllocEval:    An AllocReq is received but not evaluated yet
ptpAlloc:     Resources for a ptp connection are reserved (allocated) but the connection is not established yet
ptmAlloc:     Resources for a ptm connection are reserved (allocated) but the connection is not established yet
ptpAllocCompEval: An AllocCompReq for a ptp connection is received but not evaluated yet
ptmAllocCompEval: An AllocCompReq for a ptm connection is received but not evaluated yet
ptpInService: A ptp connection is established
ptmInService: A ptm connection is established
ModifyEval:   A ModifyReq for a ptp connection is received but not evaluated yet
Modify:       A ptp connection is prepared for modification, but the modification is not completed yet
ModifyAbortEval: A ModifyAbortReq for a ptp connection is received but not evaluated yet
ModifyCompEval: A ModifyCompReq for a ptp connection is received but not evaluated yet
DeallocEval:  A DeallocReq for a connection is received but not evaluated yet
DeallocEvalNoRes: A deallocation of the connection is requested, but no response shall be given in case of indication
ConnFail:    Failure state, only dealloc or reset is accepted
*/

```

```

/* Timerdefinitions for supervision of applications */
TIMER
T_Modification := 180; /* Default value for modification supervision timer is 180 sec, Tolerance +/- 10% */
T_ConnEst      := 180; /* Default value for allocation supervision timer is 180 sec, Tolerance +/- 10% */

```

```

/* Data type definitions internal to the process */
/**
/* This data type is used for translation of the branch identifier to the process identifier (PID)
of the branch FSM */
NEWTTYPE AT_BranchList ARRAY
(IT_BranchId,PID)
ENDNEWTTYPE AT_BranchList;

```

```

/*Primitive Data Declarations*/
/**
/* Connection identifier */
DCL
V_ConnRefNo IT_ConnRefNo;
/**
/* Branch identifier */
DCL
V_BranchId IT_BranchId;
/**
/* Reject Cause */
DCL
V_RejCause IT_RejCause;
/**
/* Update needed for addBranch */
DCL
V_UpdateNeeded IT_UpdateNeeded;

```

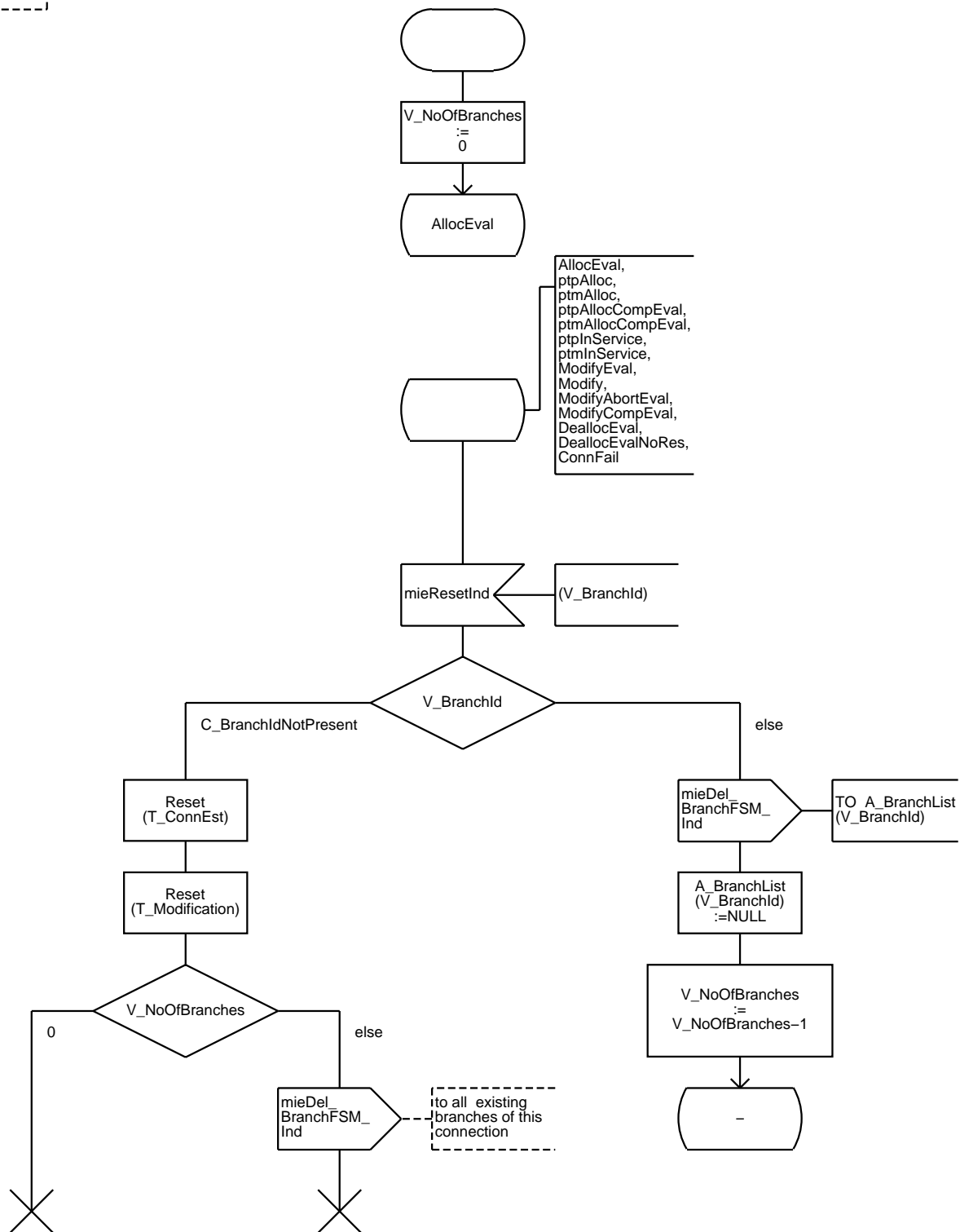
```

/* Process Data Declarations */
/**
/* List of connection FSMs */
DCL
A_BranchList AT_BranchList;
/**
/* Number of branches */
DCL
V_NoOfBranches INTEGER;
/**
/* First branch */
DCL
V_FirstBranchId IT_BranchId;

```

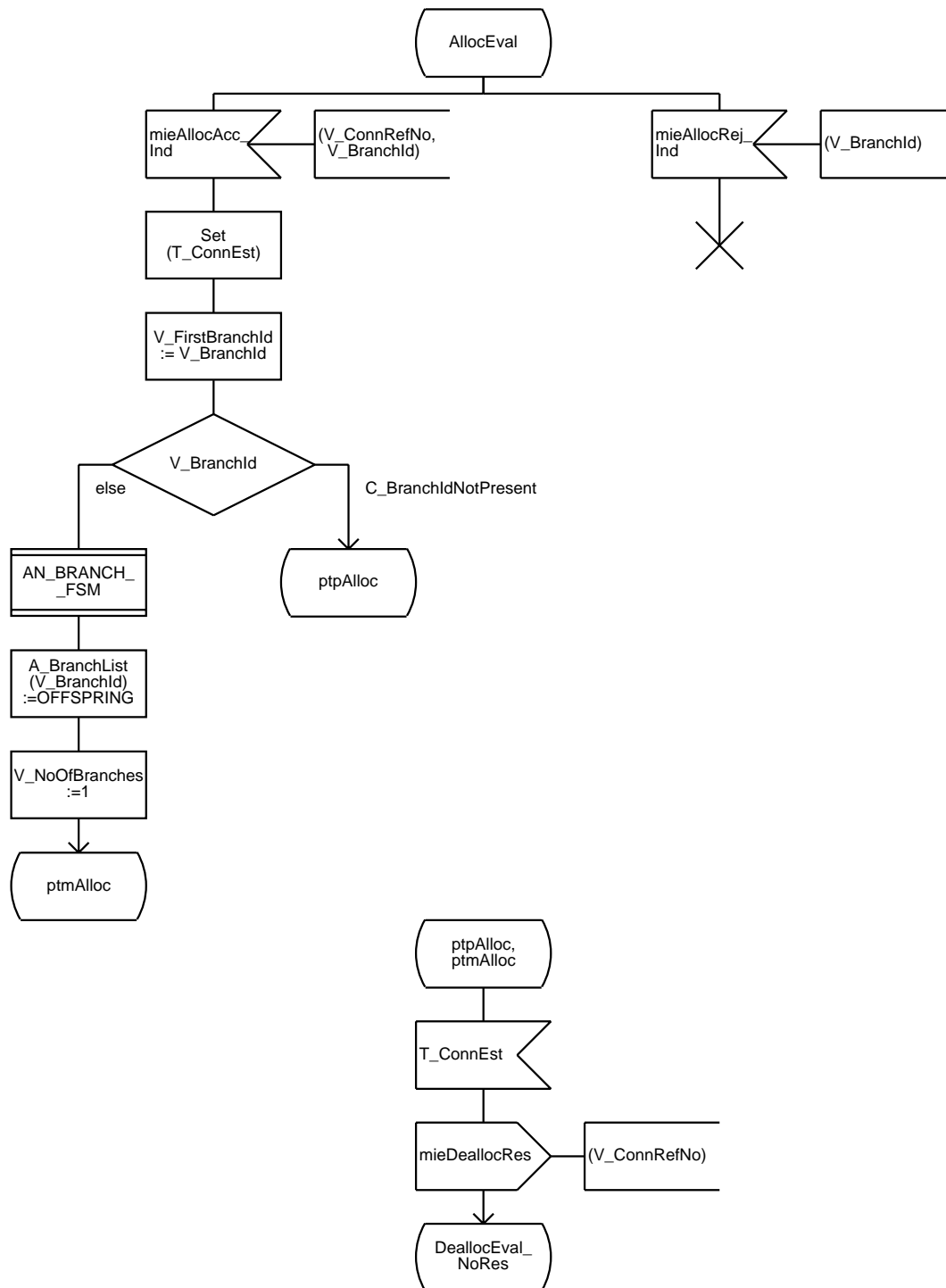
Process AN_CONN_FSM

2(12)



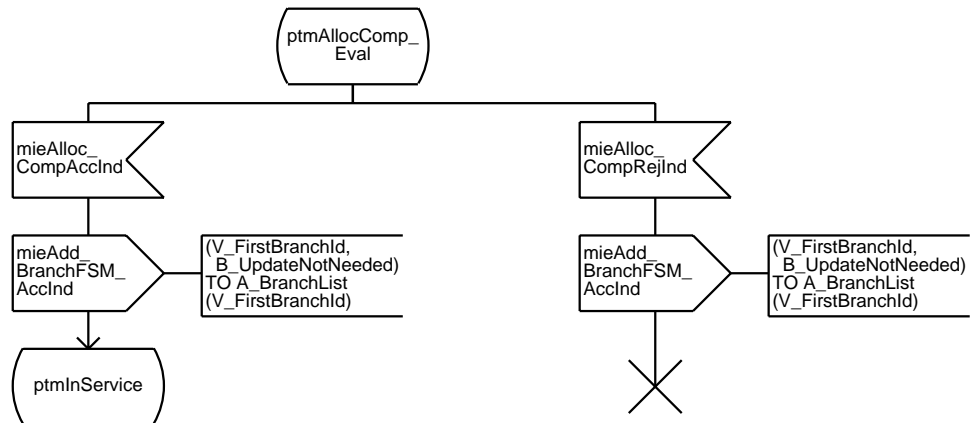
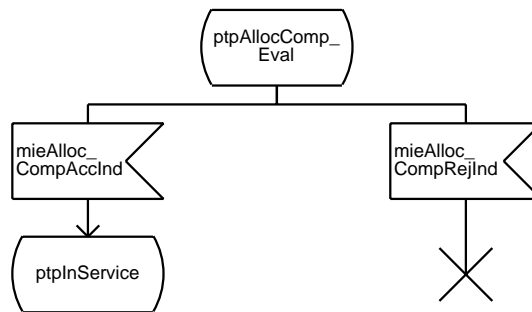
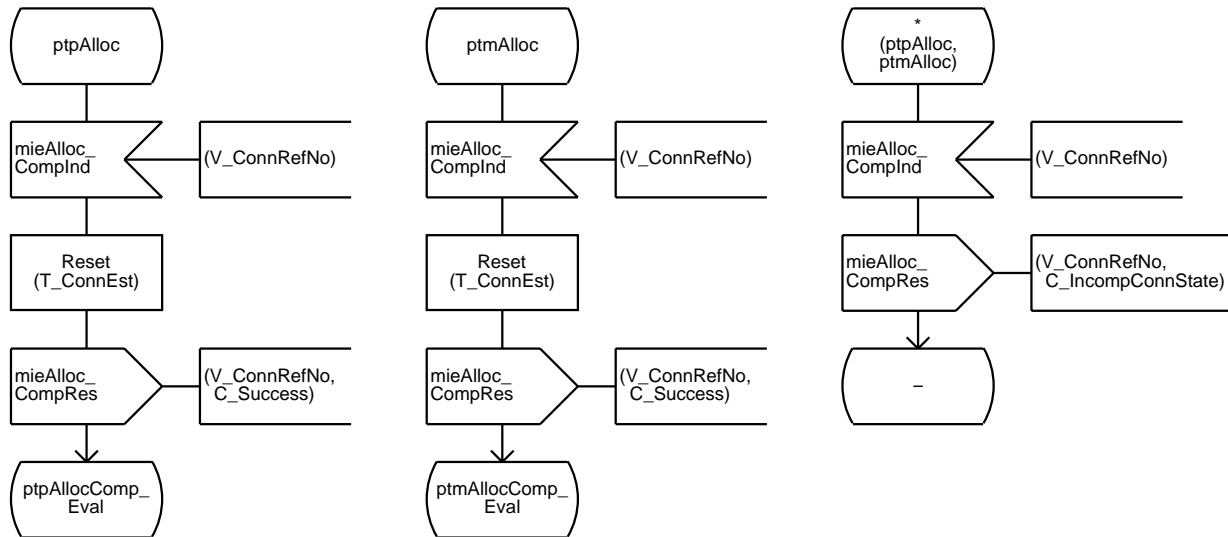
Process AN_CONN_FSM

3(12)



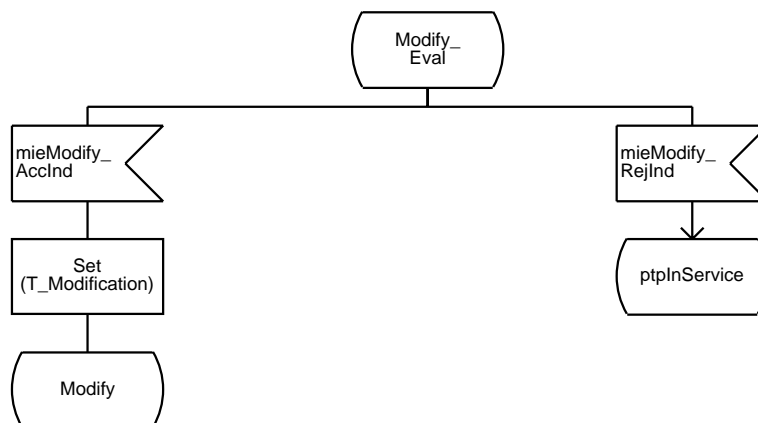
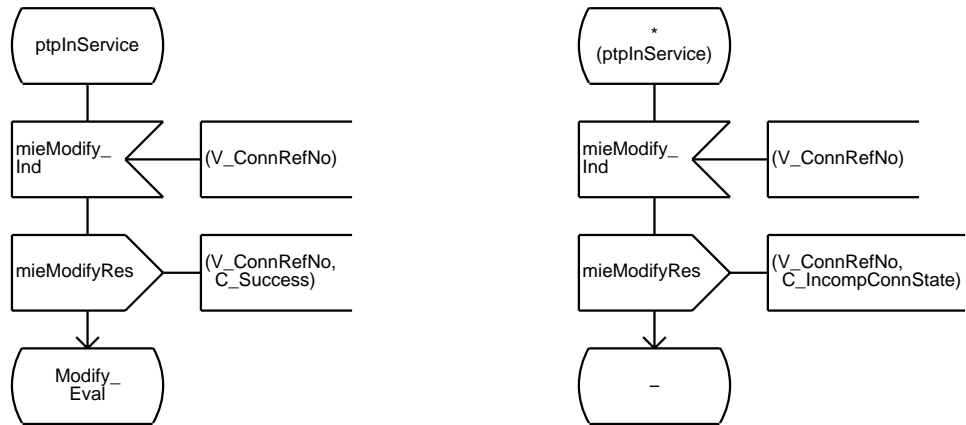
Process AN_CONN_FSM

4(12)



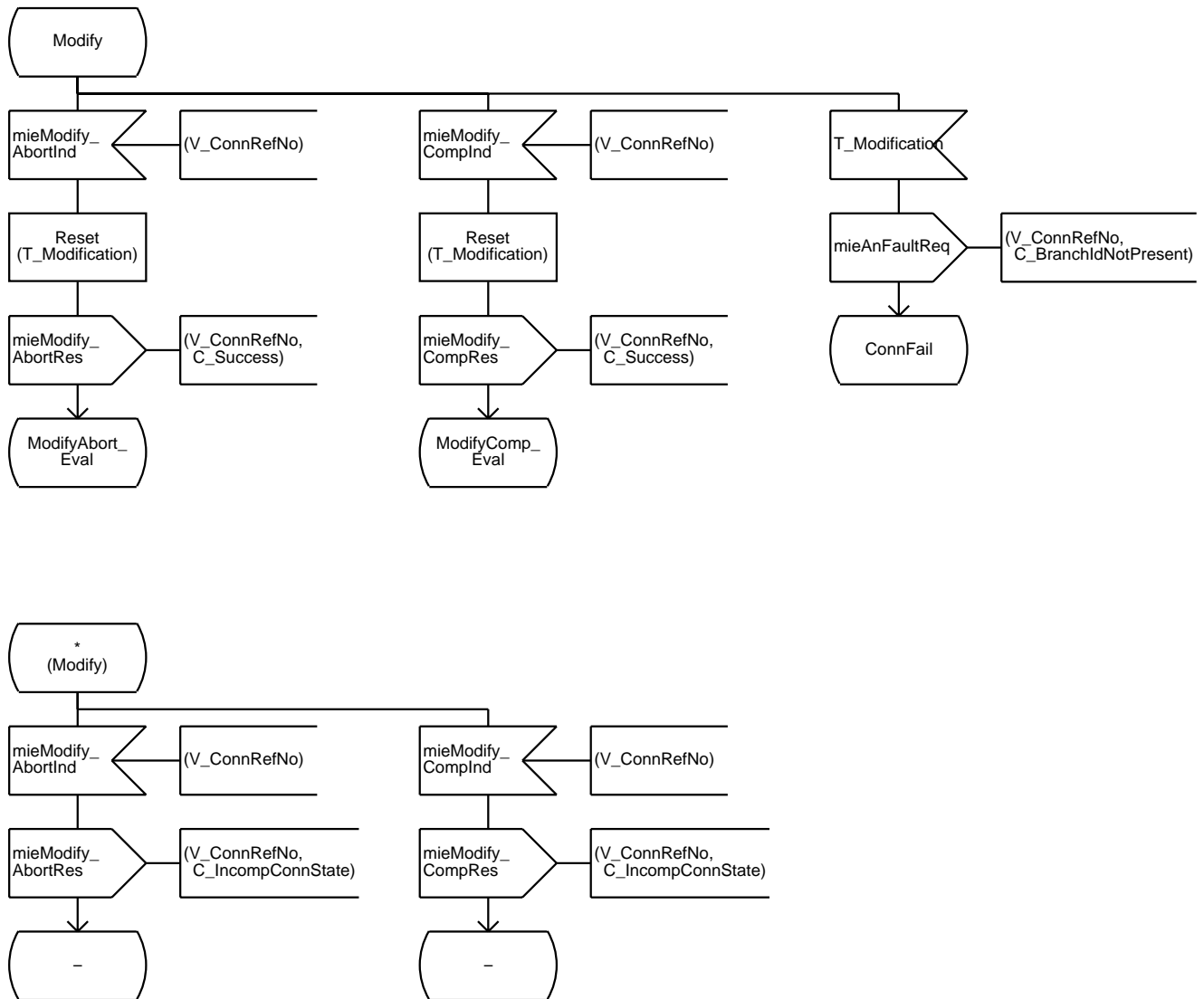
Process AN_CONN_FSM

5(12)



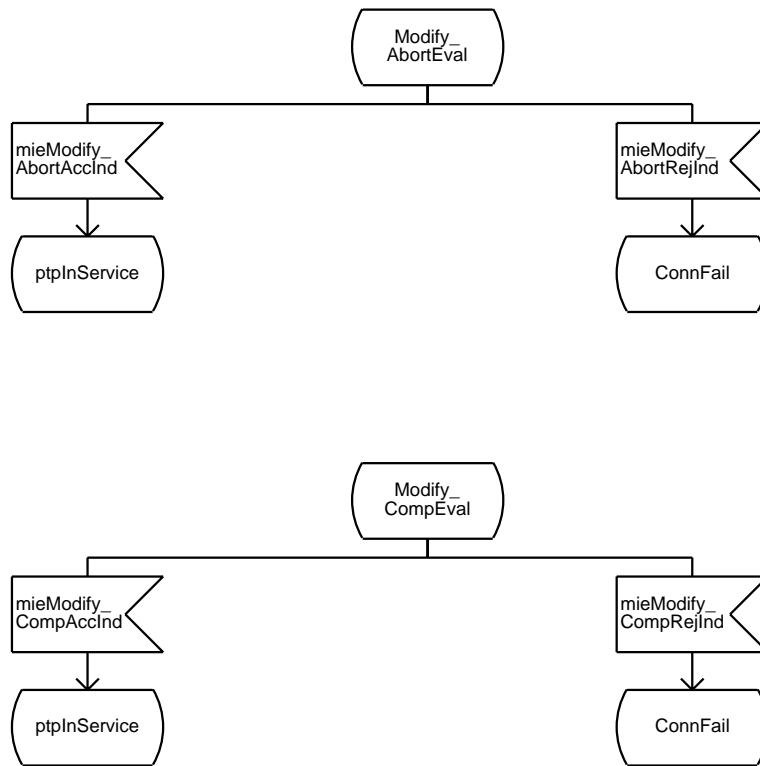
Process AN_CONN_FSM

6(12)



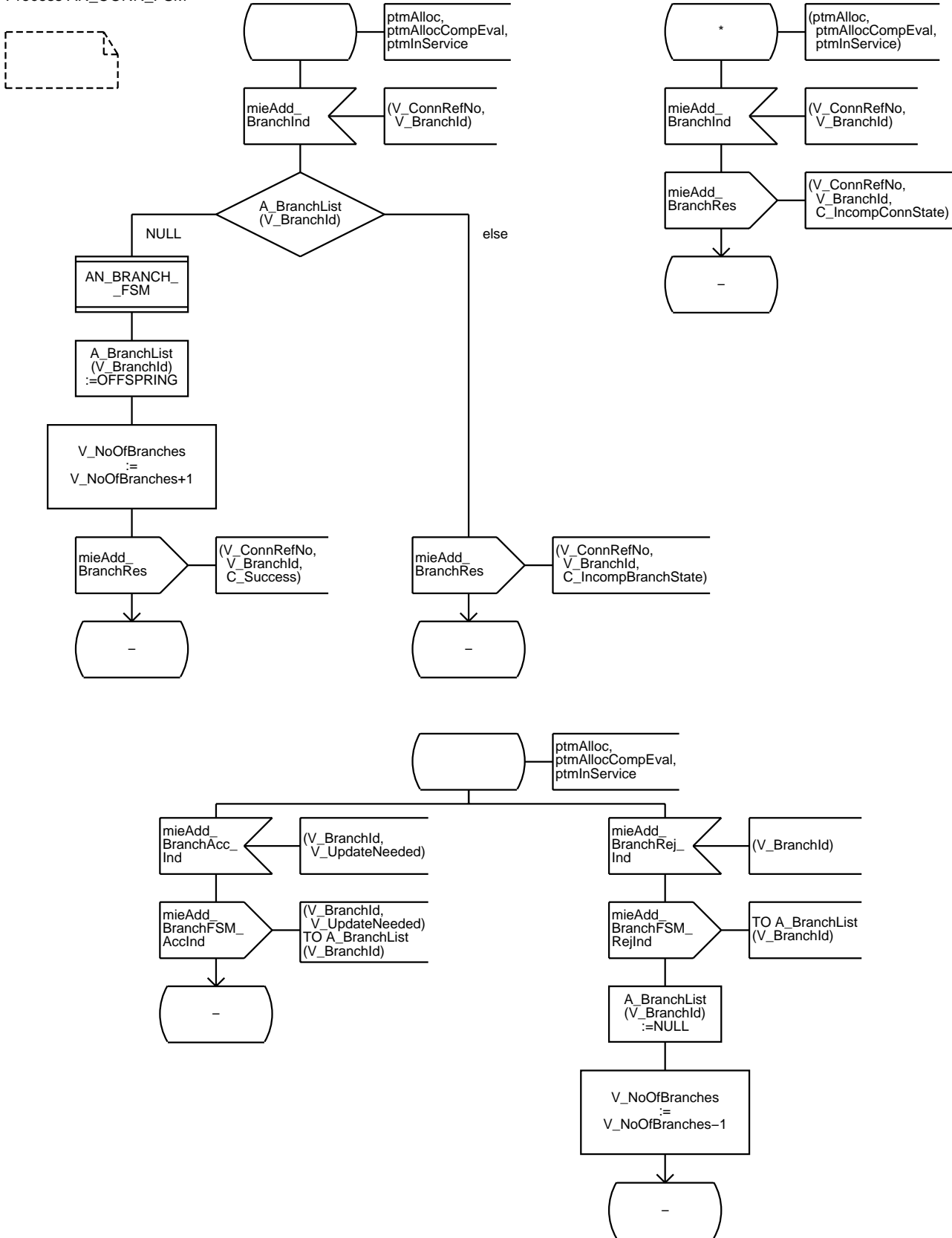
Process AN_CONN_FSM

7(12)



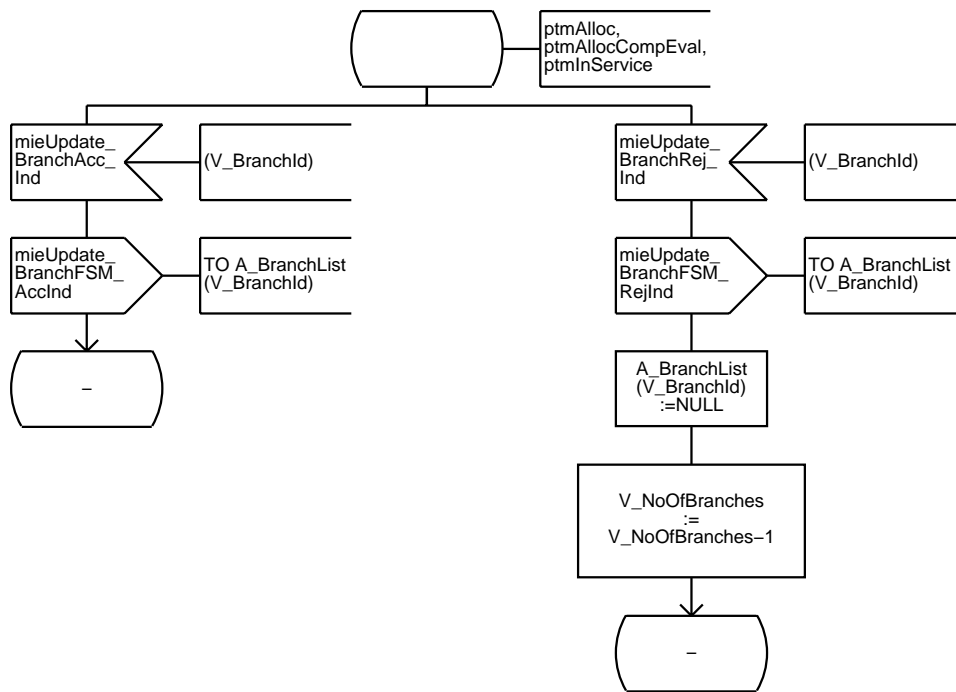
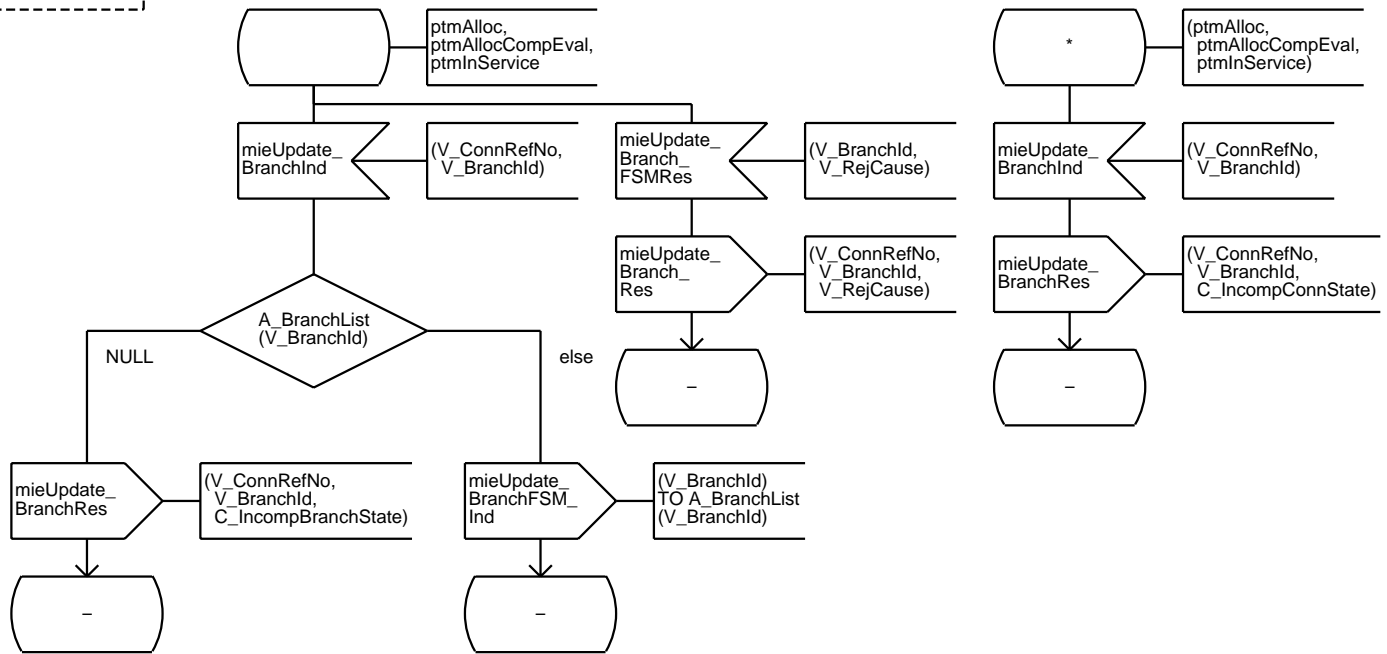
Process AN_CONN_FSM

8(12)



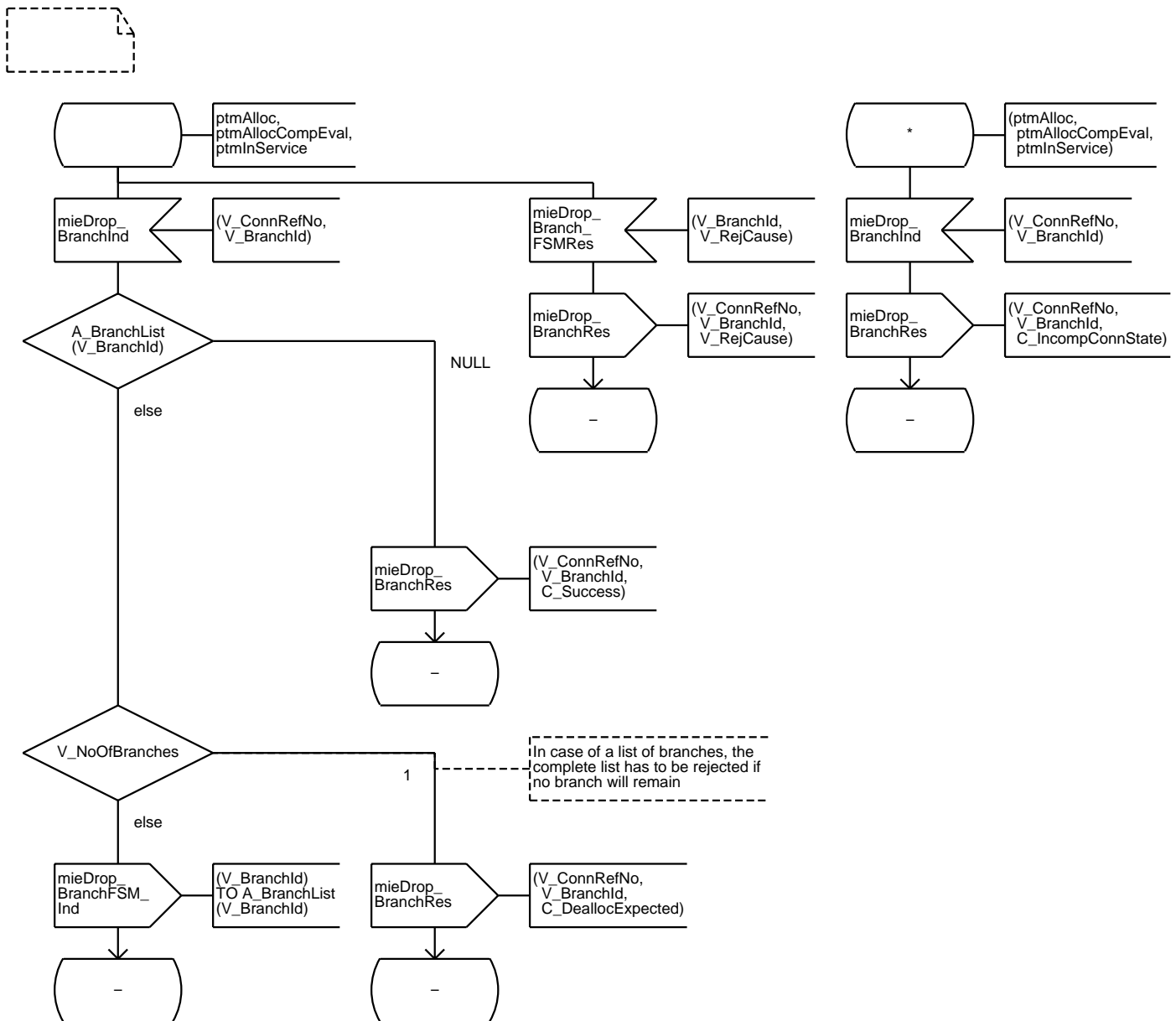
Process AN_CONN_FSM

9(12)



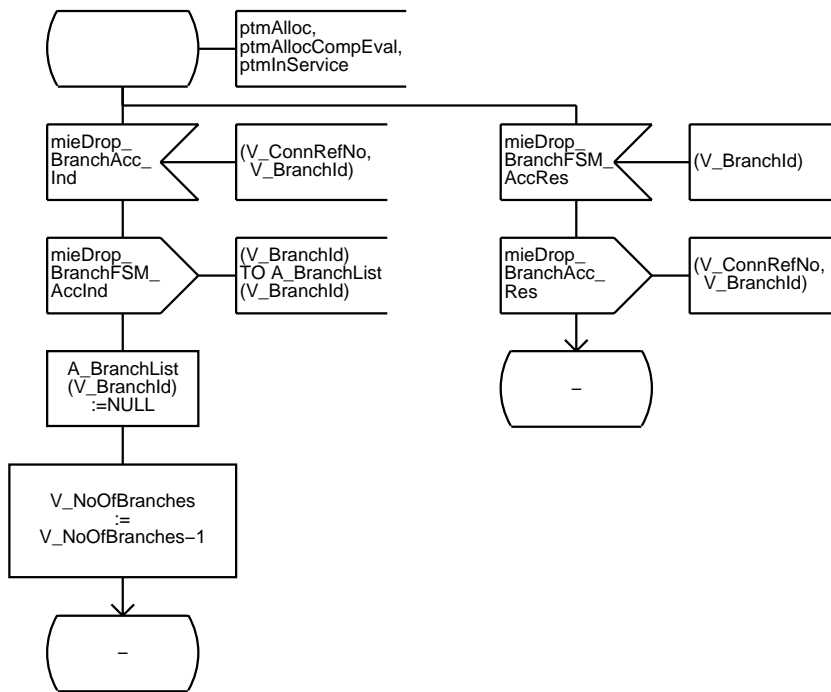
Process AN_CONN_FSM

10(12)



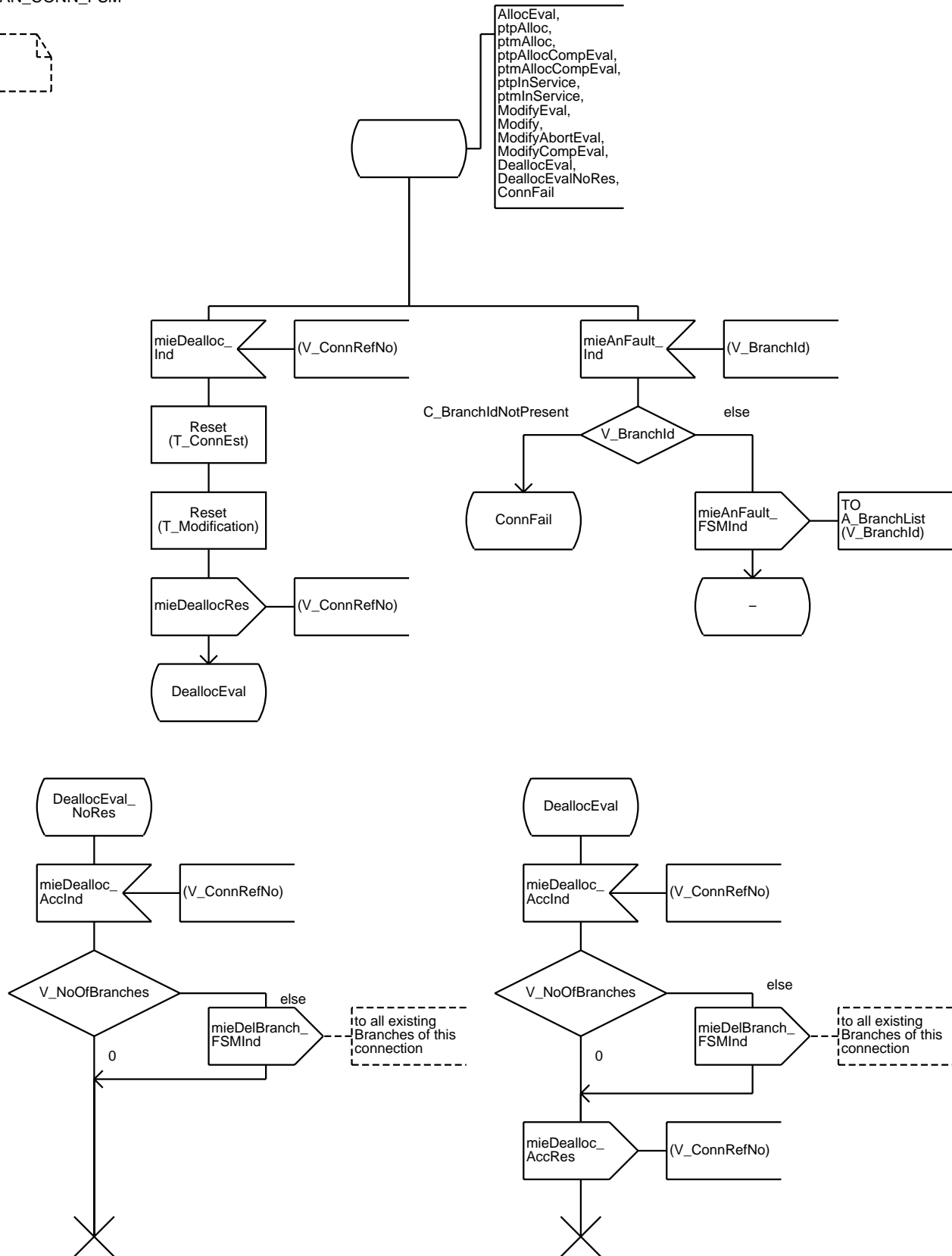
Process AN_CONN_FSM

11(12)



Process AN_CONN_FSM

12(12)



Process AN_BRANCH_FSM

1(4)



```

/* State descriptions
AddBranchEval:  Init State, this branch of the connection is allocated
InService:      The branch is established, i.e the allocation is completed
UpdateBranch:   The branch has to be updated
UpdateBranchEval: The update branch message is received but not evaluated yet
DropBranchEval: The drop branch message is received but not evaluated yet
BranchFail:     Failure state, only dealloc message is accepted
*/

```

```

/* Timerdefinitions for supervision of application */
TIMER
T_BranchEst      := 180; /* Default value for update branch supervision timer is 180 sec, Tolerance +/- 10% */

```

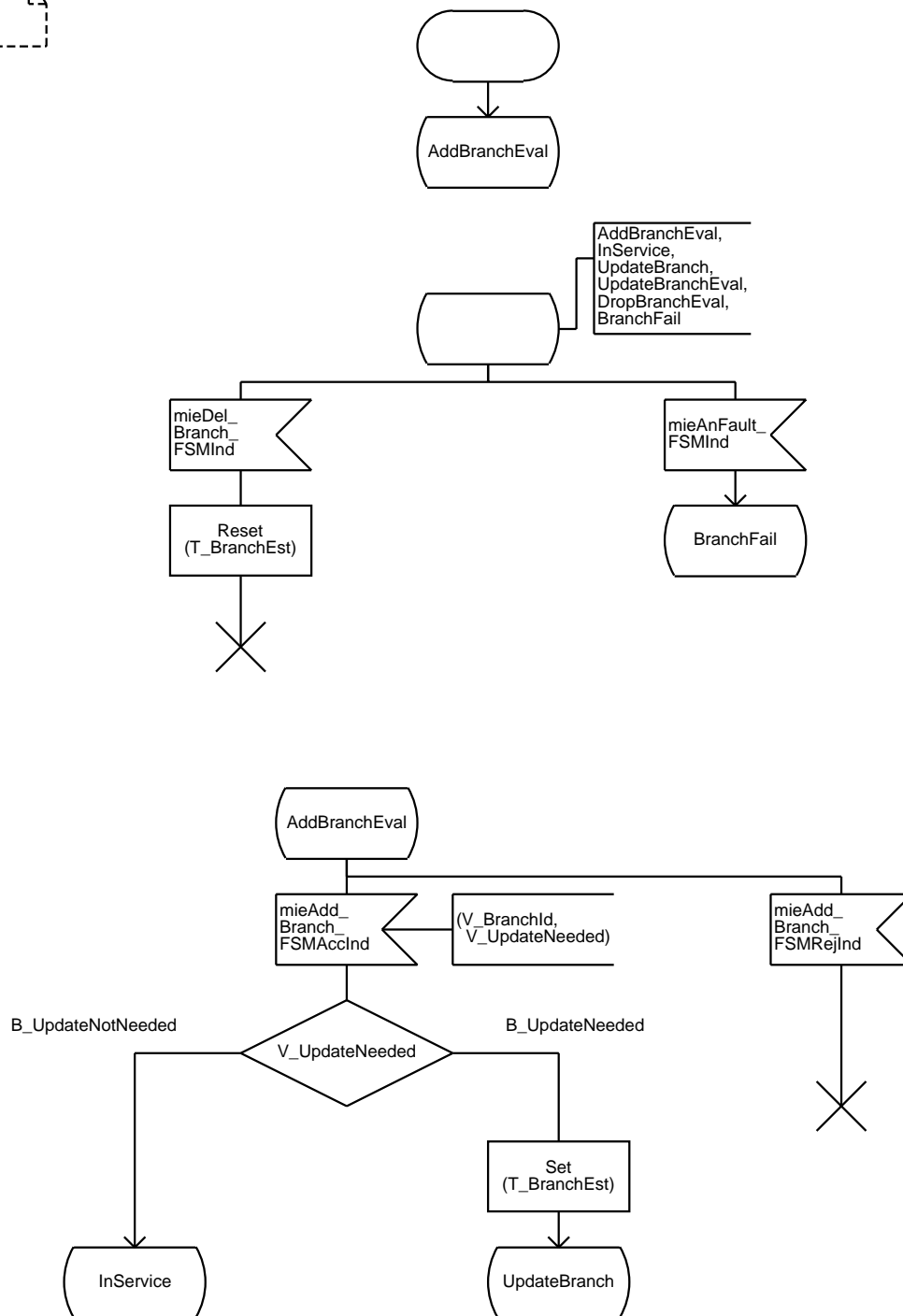
```

/*Primitive Data Declarations*/
/**/
/* Branch identifier */
DCL
V_BranchId IT_BranchId;
/**/
/* Update needed for addBranch */
DCL
V_UpdateNeeded IT_UpdateNeeded;

```

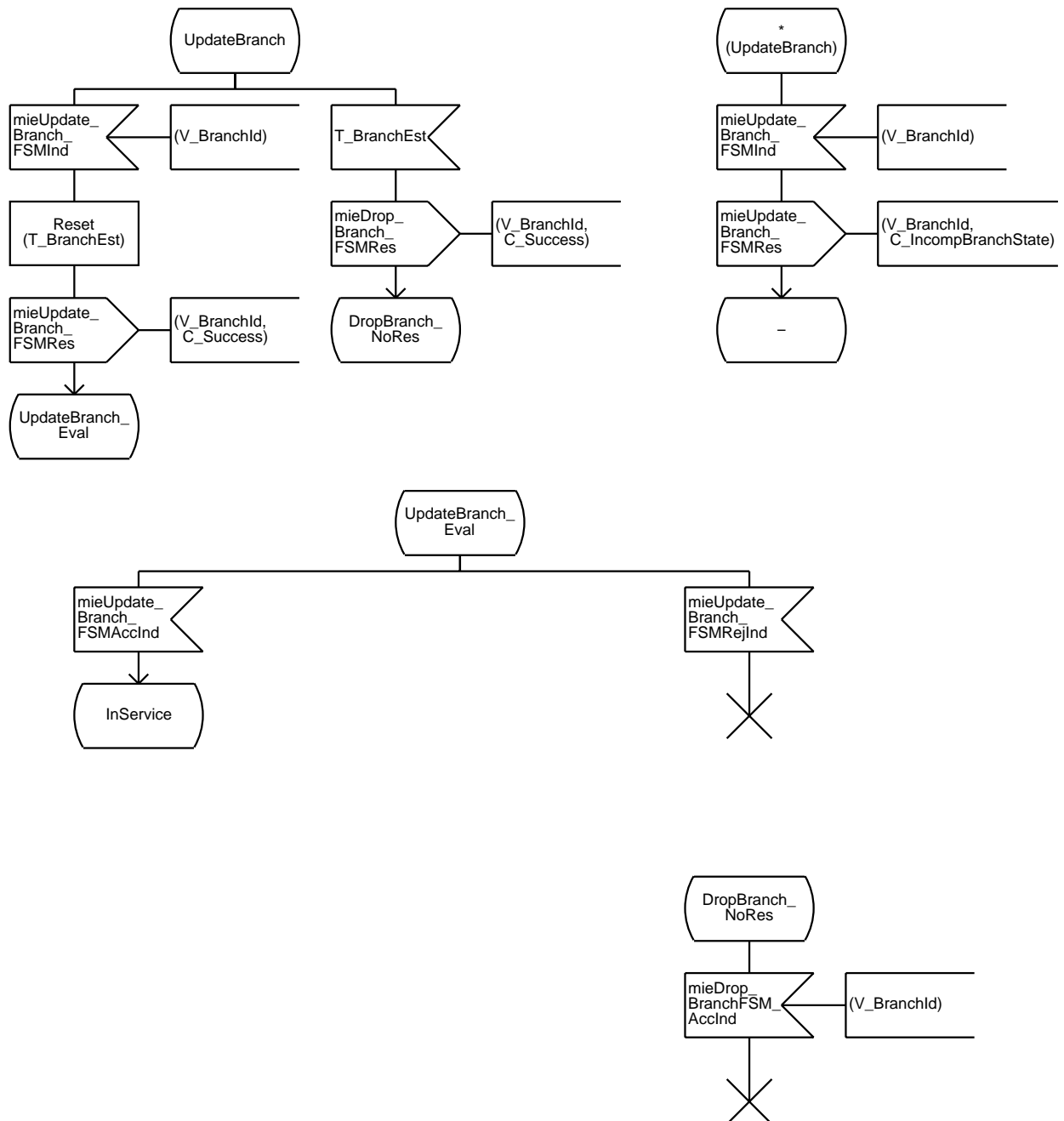
Process AN_BRANCH_FSM

2(4)



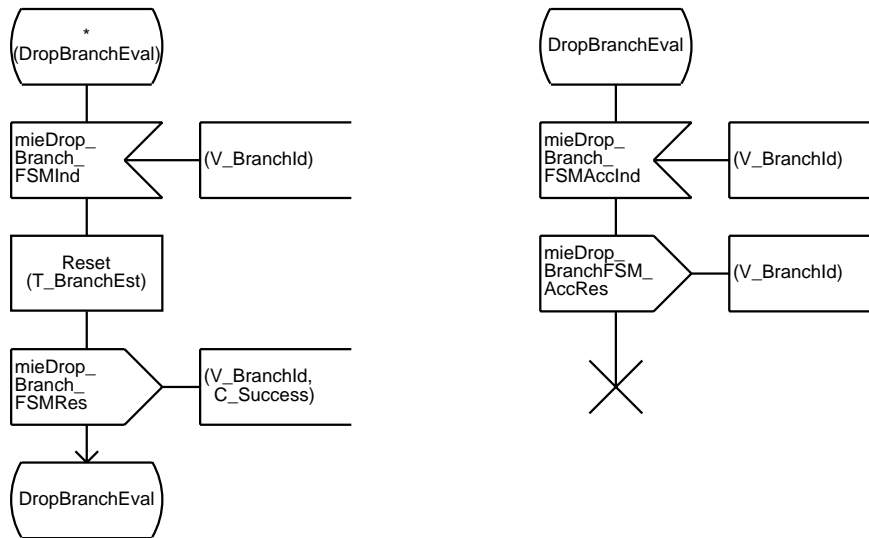
Process AN_BRANCH_FSM

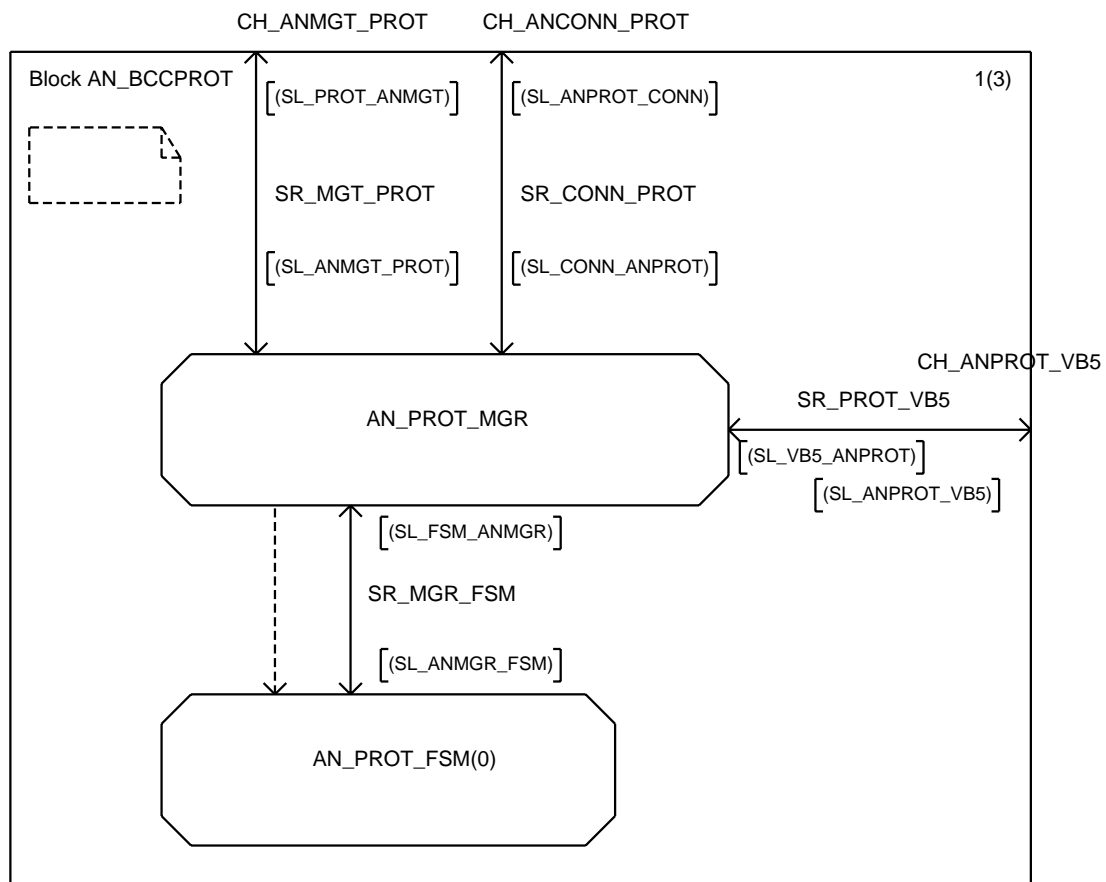
3(4)



Process AN_BRANCH_FSM

4(4)





Block AN_BCCPROT

2(3)



```
/* Declaration of primitives internal
to block AN_BCCPROT */
```

```
signal
mieAllocInd      (IT_ConnRefNo, IT_BranchId),
mieAllocAccRes   (IT_ConnRefNo, IT_BranchId),
mieAllocRejRes   (IT_ConnRefNo, IT_BranchId, IT_RejCause),
mieAllocCompInd  (IT_ConnRefNo),
mieAllocCompAccRes (IT_ConnRefNo),
mieAllocCompRejRes (IT_ConnRefNo, IT_RejCause),
mieBbccResetInd  ,
mieBbccResetAccRes (IT_ConnRefNo),
mieBbccResetRejRes (IT_ConnRefNo, IT_RejCause),
mieBbccPresyncInd ,
mieBbccPresyncAccRes (IT_ConnRefNo),
mieBbccPresyncRejRes (IT_ConnRefNo, IT_RejCause),
mieDeallocInd    (IT_ConnRefNo),
mieDeallocAccRes (IT_ConnRefNo),
mieModifyInd     (IT_ConnRefNo),
mieModifyAccRes  (IT_ConnRefNo),
mieModifyRejRes  (IT_ConnRefNo, IT_RejCause),
mieModifyCompInd (IT_ConnRefNo),
mieModifyCompAccRes (IT_ConnRefNo),
mieModifyCompRejRes (IT_ConnRefNo, IT_RejCause),
mieModifyAbortInd (IT_ConnRefNo),
mieModifyAbortAccRes (IT_ConnRefNo),
mieModifyAbortRejRes (IT_ConnRefNo, IT_RejCause),
mieAddBranchInd  (IT_ConnRefNo, IT_BranchId),
mieAddBranchAccRes (IT_ConnRefNo, IT_BranchId),
mieAddBranchRejRes (IT_ConnRefNo, IT_BranchId, IT_RejCause),
mieUpdateBranchInd (IT_ConnRefNo, IT_BranchId),
mieUpdateBranchAccRes (IT_ConnRefNo, IT_BranchId),
mieUpdateBranchRejRes (IT_ConnRefNo, IT_BranchId, IT_RejCause),
mieDropBranchInd (IT_ConnRefNo, IT_BranchId),
mieDropBranchAccRes (IT_ConnRefNo, IT_BranchId),
mieDropBranchRejRes (IT_ConnRefNo, IT_BranchId, IT_RejCause),
mieAnFaultReq    (IT_TransId, IT_ConnRefNo, IT_BranchId, IT_Response),
mieAnFaultAccConf (IT_ConnRefNo, IT_BranchId, IT_RejCause);
```

Block AN_BCCPROT

3(3)



```
/* Declaration of signallists
   internal to block AN_BCCPROT */
```

```
signallist SL_FSM_ANMGR =
mieAllocInd,
mieAllocCompInd,
mieBbccResetInd,
mieBbccPresyncInd,
mieDeallocInd,
mieModifyInd,
mieModifyCompInd,
mieModifyAbortInd,
mieAddBranchInd,
mieUpdateBranchInd,
mieDropBranchInd,
bccAllocAcc,
bccAllocRej,
bccAllocCompAcc,
bccAllocCompRej,
bccBbccResetAcc,
bccBbccResetRej,
bccPresyncAcc,
bccPresyncRej,
bccDeallocAcc,
bccModifyAcc,
bccModifyRej,
bccModifyCompAcc,
bccModifyCompRej,
bccModifyAbortAcc,
bccModifyAbortRej,
bccAddBranchAcc,
bccAddBranchRej,
bccUpdateBranchAcc,
bccUpdateBranchRej,
bccDropBranchAcc,
bccDropBranchRej,
bccProtocolError,
bccAnFault,
mieProtocolError,
mieAnFaultAccConf,
mieErrorInd;
```

```
signallist SL_ANMGR_FSM =
bccAlloc,
bccAllocComp,
bccBbccReset,
bccPresync,
bccDealloc,
bccModify,
bccModifyComp,
bccModifyAbort,
bccAddBranch,
bccUpdateBranch,
bccDropBranch,
bccUnknownMessage,
bccUnknownProtdisc,
bccMsgTooShort,
bccTransIdErr,
bccAnFaultAcc,
mieAllocAccRes,
mieAllocRejRes,
mieAllocCompAccRes,
mieAllocCompRejRes,
mieBbccResetAccRes,
mieBbccResetRejRes,
mieBbccPresyncAccRes,
mieBbccPresyncRejRes,
mieDeallocAccRes,
mieModifyAccRes,
mieModifyRejRes,
mieModifyCompAccRes,
mieModifyCompRejRes,
mieModifyAbortAccRes,
mieModifyAbortRejRes,
mieAddBranchAccRes,
mieAddBranchRejRes,
mieUpdateBranchAccRes,
mieUpdateBranchRejRes,
mieDropBranchAccRes,
mieDropBranchRejRes,
mieAnFaultReq,
mieAbortReq;
```

Process AN_PROT_MGR

1(22)

/*AN BCC protocol management entity
controls instantiation of
protocol FSMs (process AN_PROT_FSM) */

/* If a message is repeated by the remote side and an acknowledgement is already sent, i.e. the corresponding PROT_FSM is deleted, the process PROT_MGR has to take care that the same acknowledgement as before is sent to the remote side. A repeated message by the remote side can be detected by a transaction identifier recently used*/

/* The generation of the transaction identifier, i.e. the increment by one for each invoke message, is for simulation purposes only. In an implementation, any other method which is appropriate may be applied */

/* Data type definitions internal to the process */
/**/
/* This data type is used for translation of the remote transaction identifier to the process identifier (PID)
of the process handling the corresponding transaction in the AN */
NEWTYPE AT_RemTransList ARRAY
(IT_TransId, PID)
ENDNEWTYPE AT_RemTransList;
/**/
/* This data type is used for translation of the local transaction identifier to the process identifier (PID)
of the process handling the corresponding transaction in the AN */
NEWTYPE AT_LocTransList ARRAY
(IT_TransId, PID)
ENDNEWTYPE AT_LocTransList;

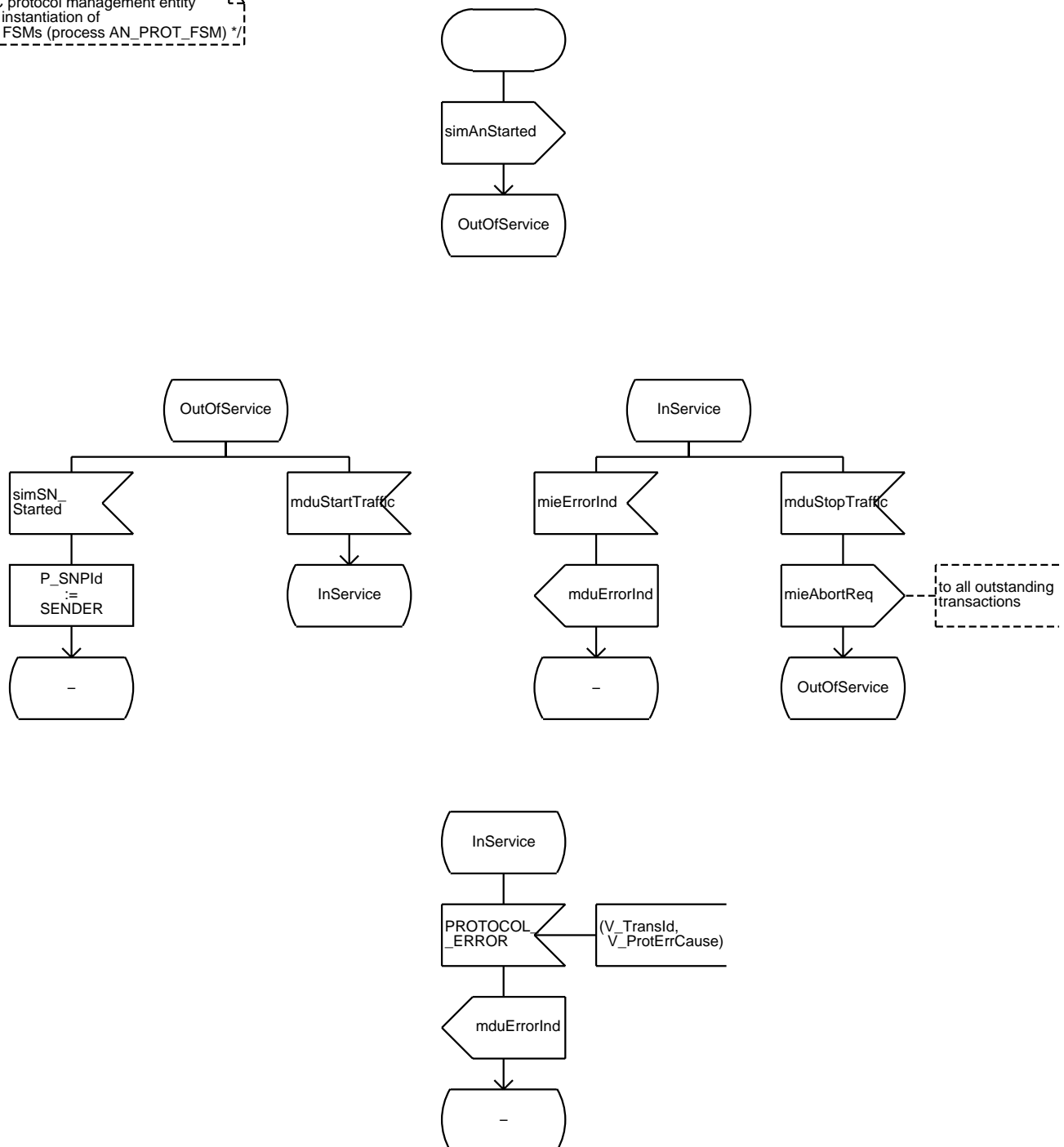
/* Primitive Data Declarations */
/**/
/* Protocol error cause */
DCL
V_ProtErrCause IT_ProtErrCause;
/**/
/* Transaction identifier */
DCL
V_TransId IT_TransId;
/**/
/* Protocol process identifier */
DCL
P_ProtId PT_ProtId;
/**/
/* Connection reference number */
DCL
V_ConnRefNo IT_ConnRefNo;
/**/
/* Branch identifier */
DCL
V_BranchId IT_BranchId;
/**/
/* Reject Cause */
DCL
V_RejCause IT_RejCause;
/**/
/* Response */
DCL
V_Response IT_Response;

/* Process Data Declarations */
/**/
/* List of AN_PROT_FSMs */
DCL
A_RemTransList AT_RemTransList,
A_LocTransList AT_LocTransList;
/**/
/* PID of the receiver process in the SN */
DCL
P_SNPId PID;
/**/
/* Counter for transaction identifier */
DCL
V_NewTransId IT_TransId := 1;

Process AN_PROT_MGR

2(22)

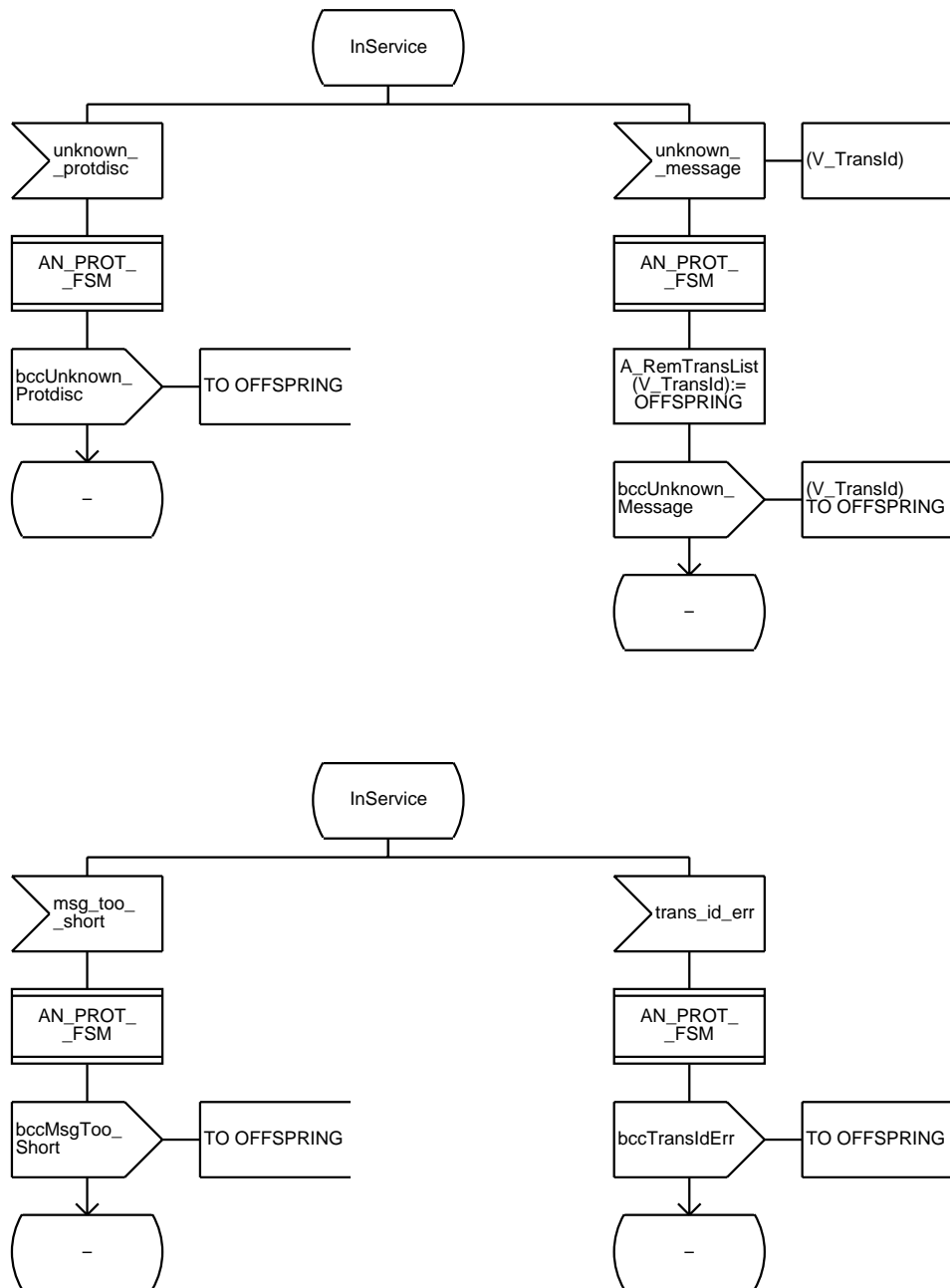
/*AN BCC protocol management entity
controls instantiation of
protocol FSMs (process AN_PROT_FSM) */



Process AN_PROT_MGR

3(22)

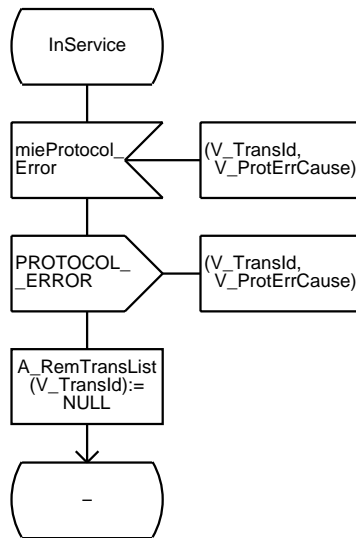
/*AN BCC protocol management entity
controls instantiation of
protocol FSMs (process AN_PROT_FSM) */



Process AN_PROT_MGR

4(22)

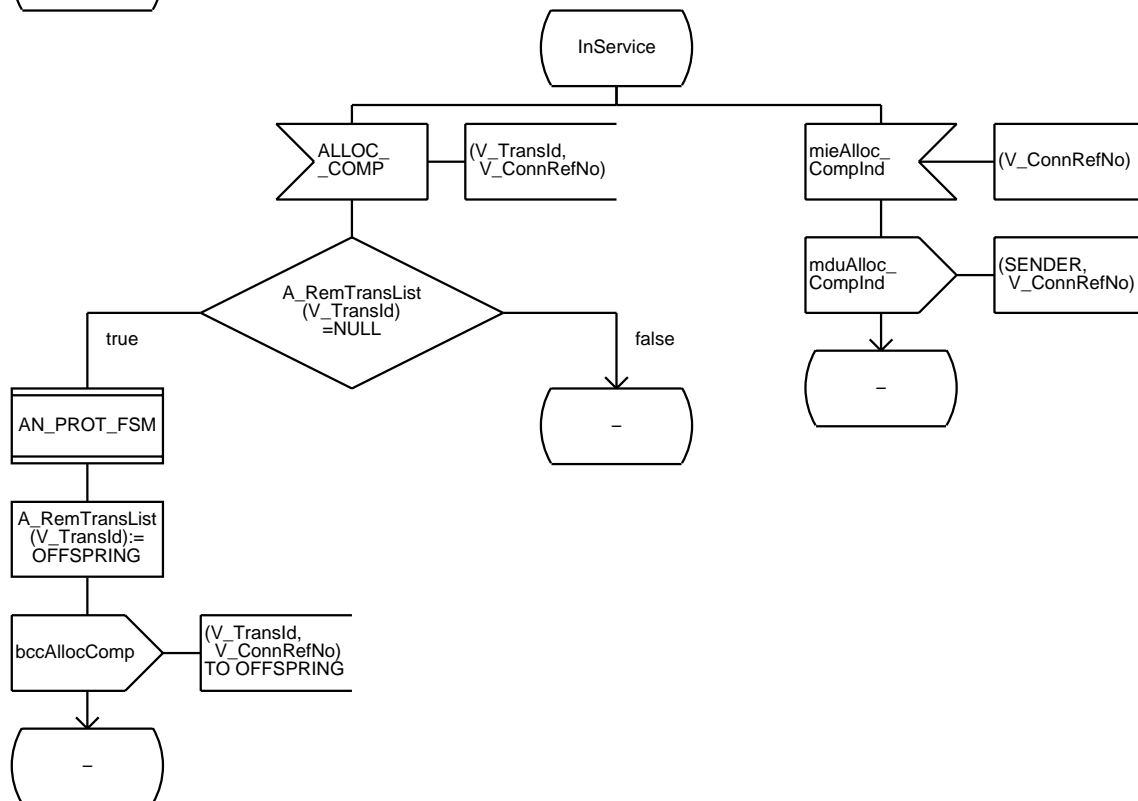
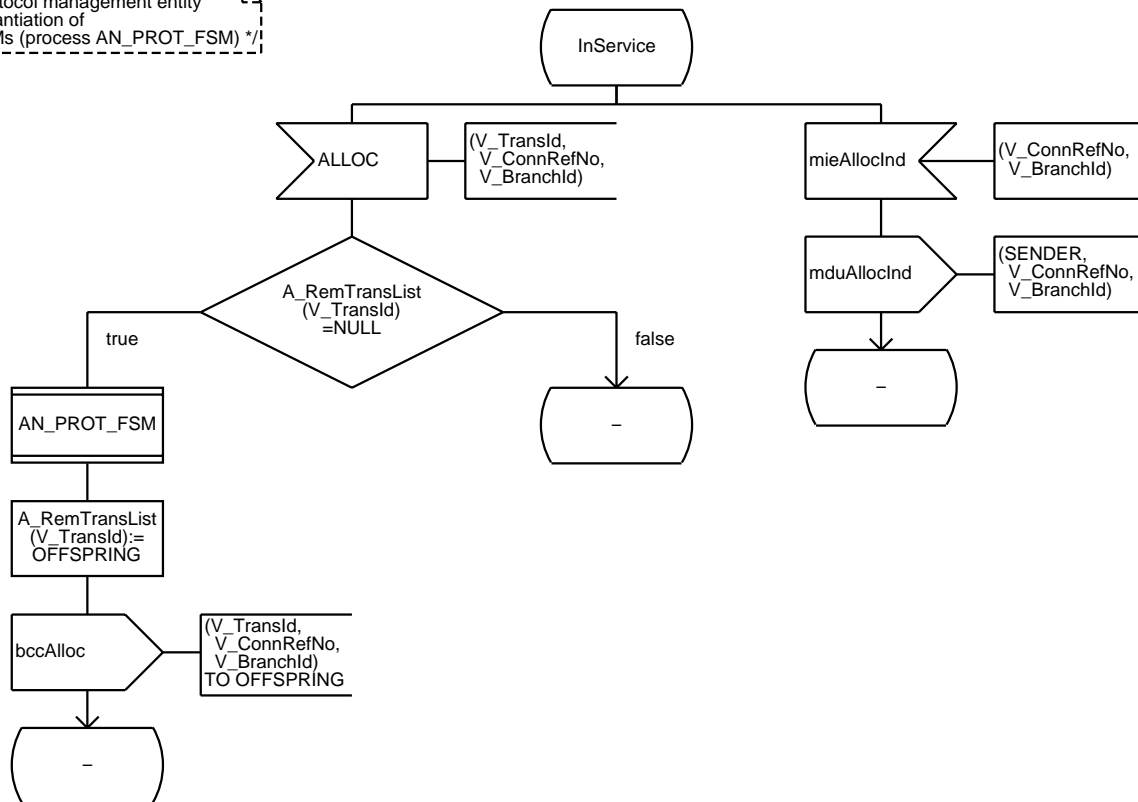
/*AN BCC protocol management entity
controls instantiation of
protocol FSMs (process AN_PROT_FSM) */



Process AN_PROT_MGR

5(22)

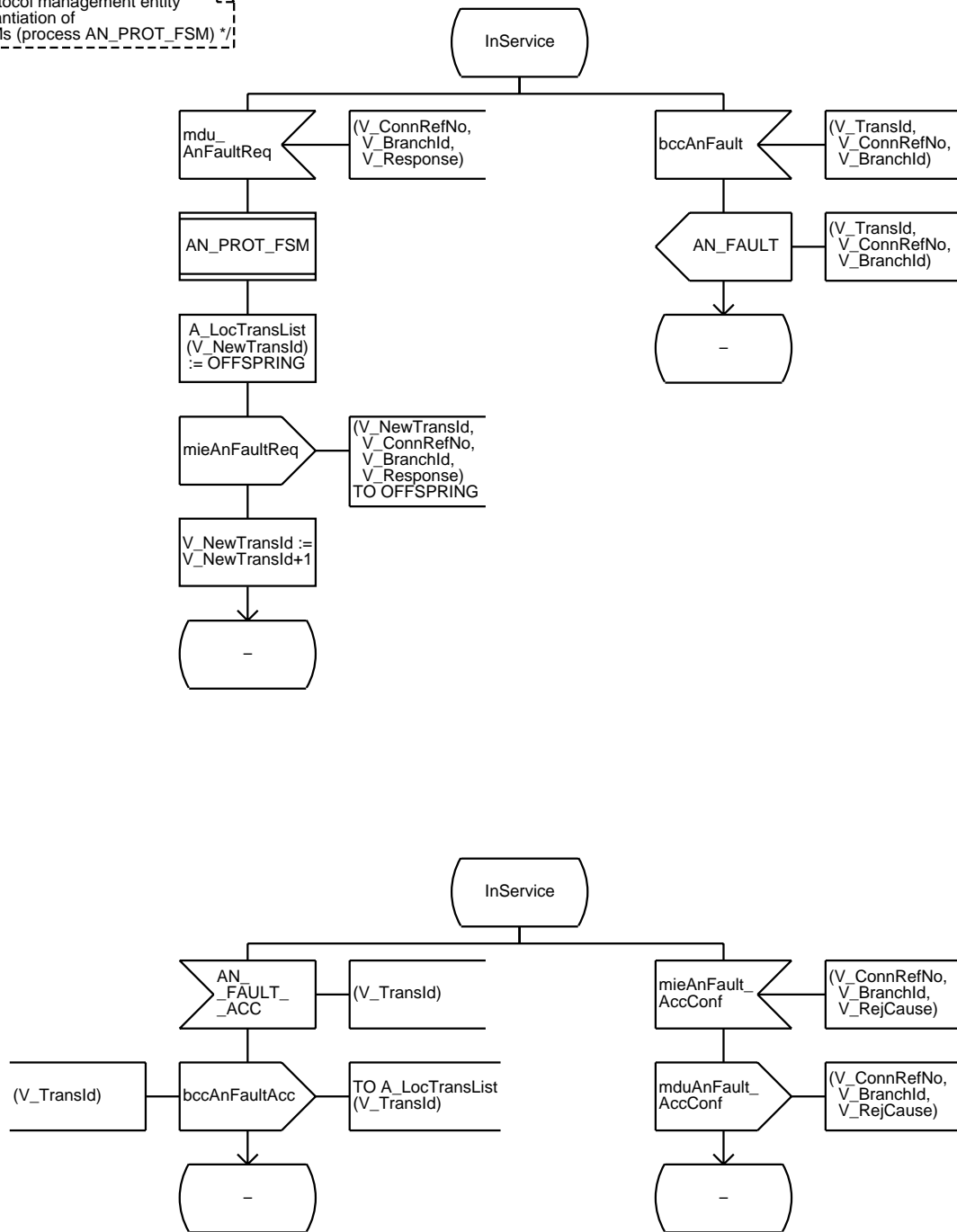
/*AN BCC protocol management entity
controls instantiation of
protocol FSMs (process AN_PROT_FSM) */



Process AN_PROT_MGR

6(22)

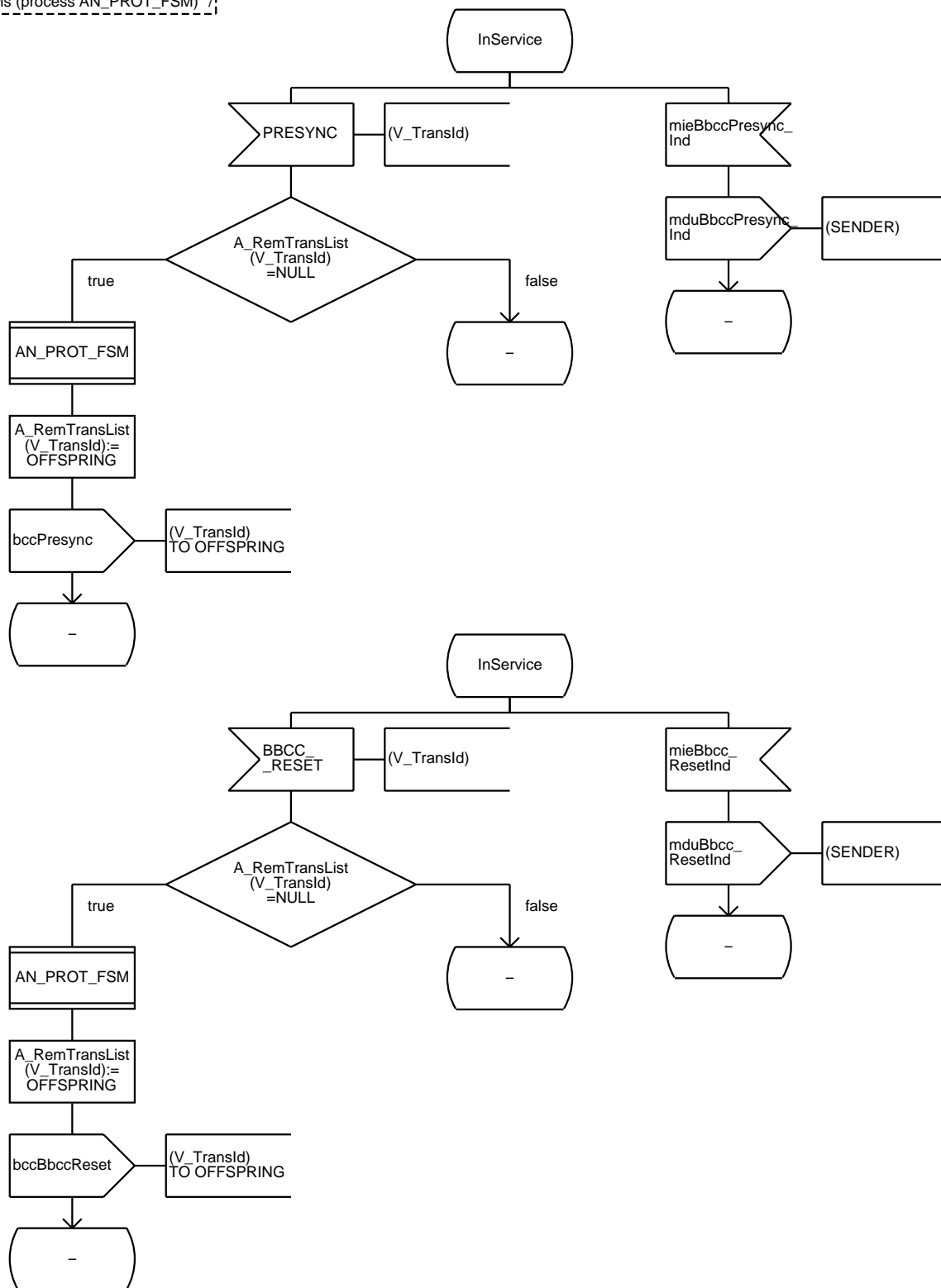
/*AN BCC protocol management entity
controls instantiation of
protocol FSMs (process AN_PROT_FSM) */



Process AN_PROT_MGR

7(22)

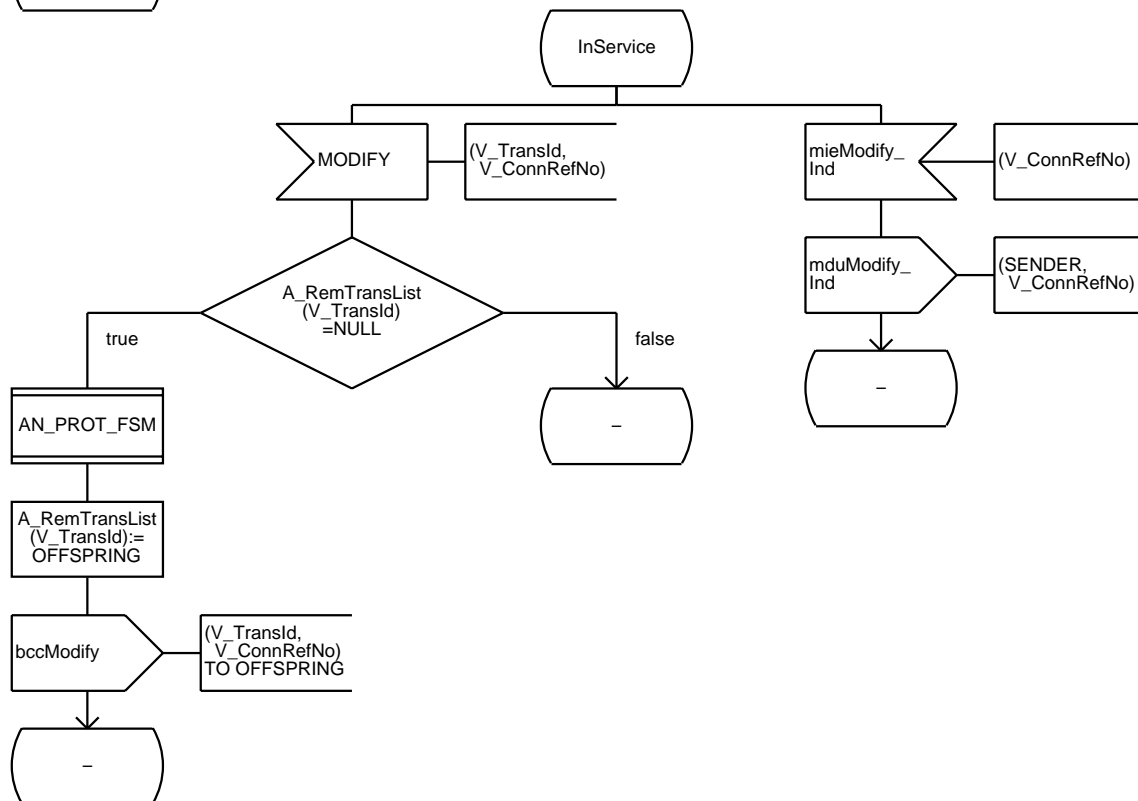
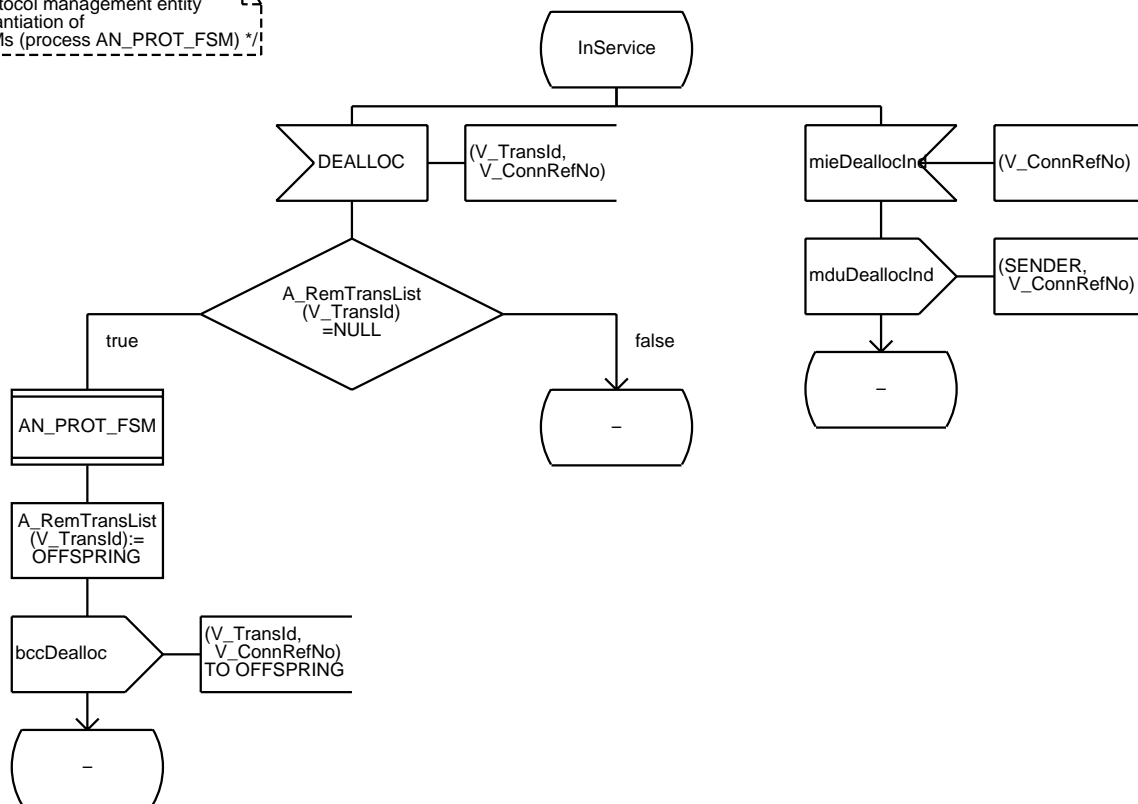
/*AN BCC protocol management entity
controls instantiation of
protocol FSMs (process AN_PROT_FSM) */



Process AN_PROT_MGR

8(22)

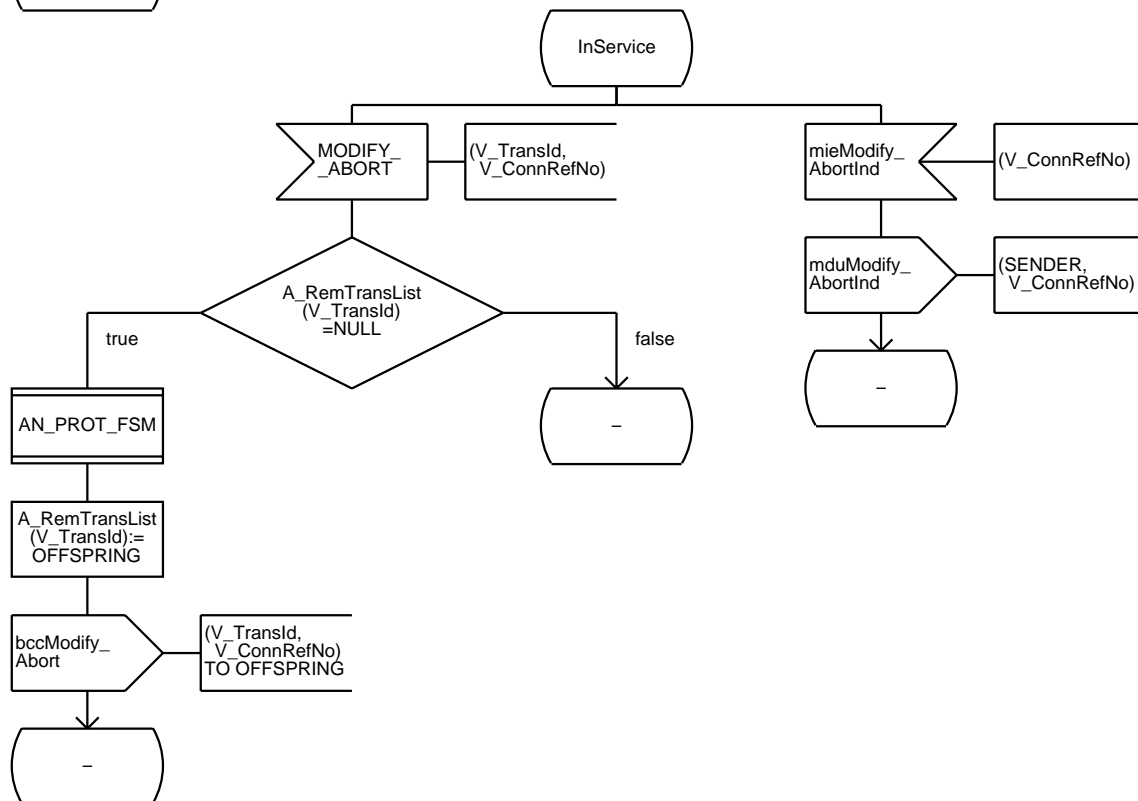
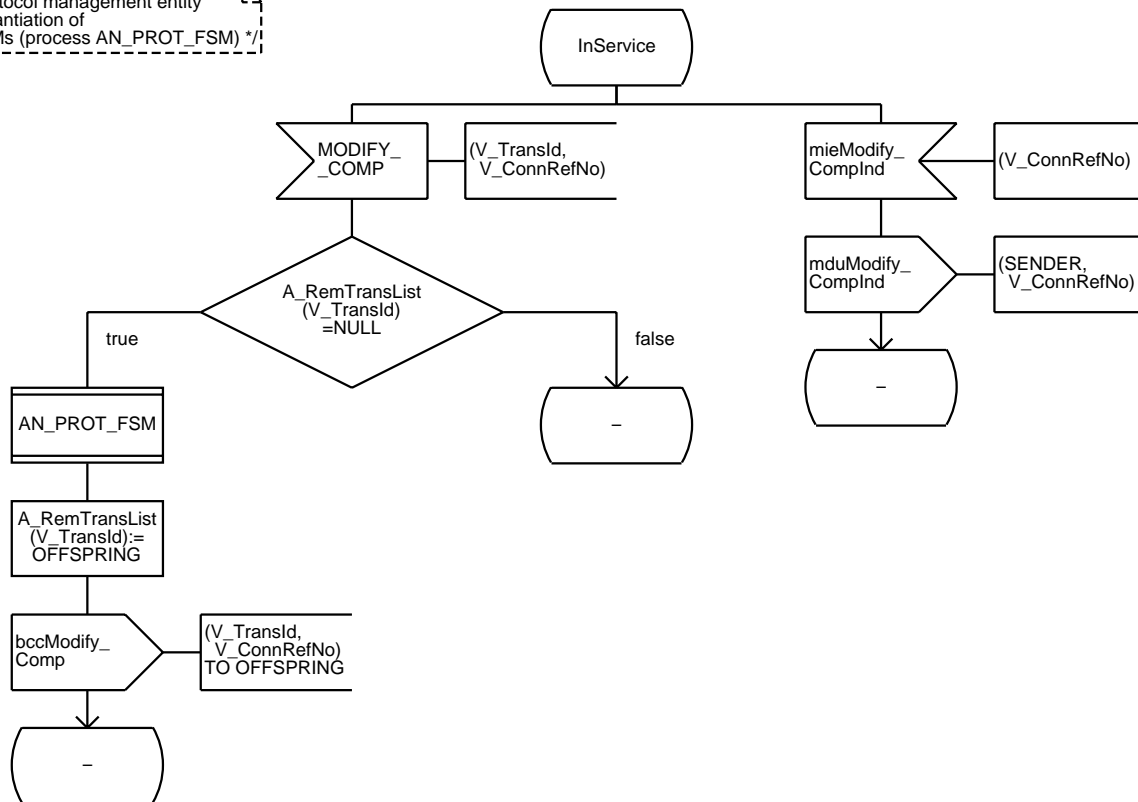
/*AN BCC protocol management entity
controls instantiation of
protocol FSMs (process AN_PROT_FSM) */



Process AN_PROT_MGR

9(22)

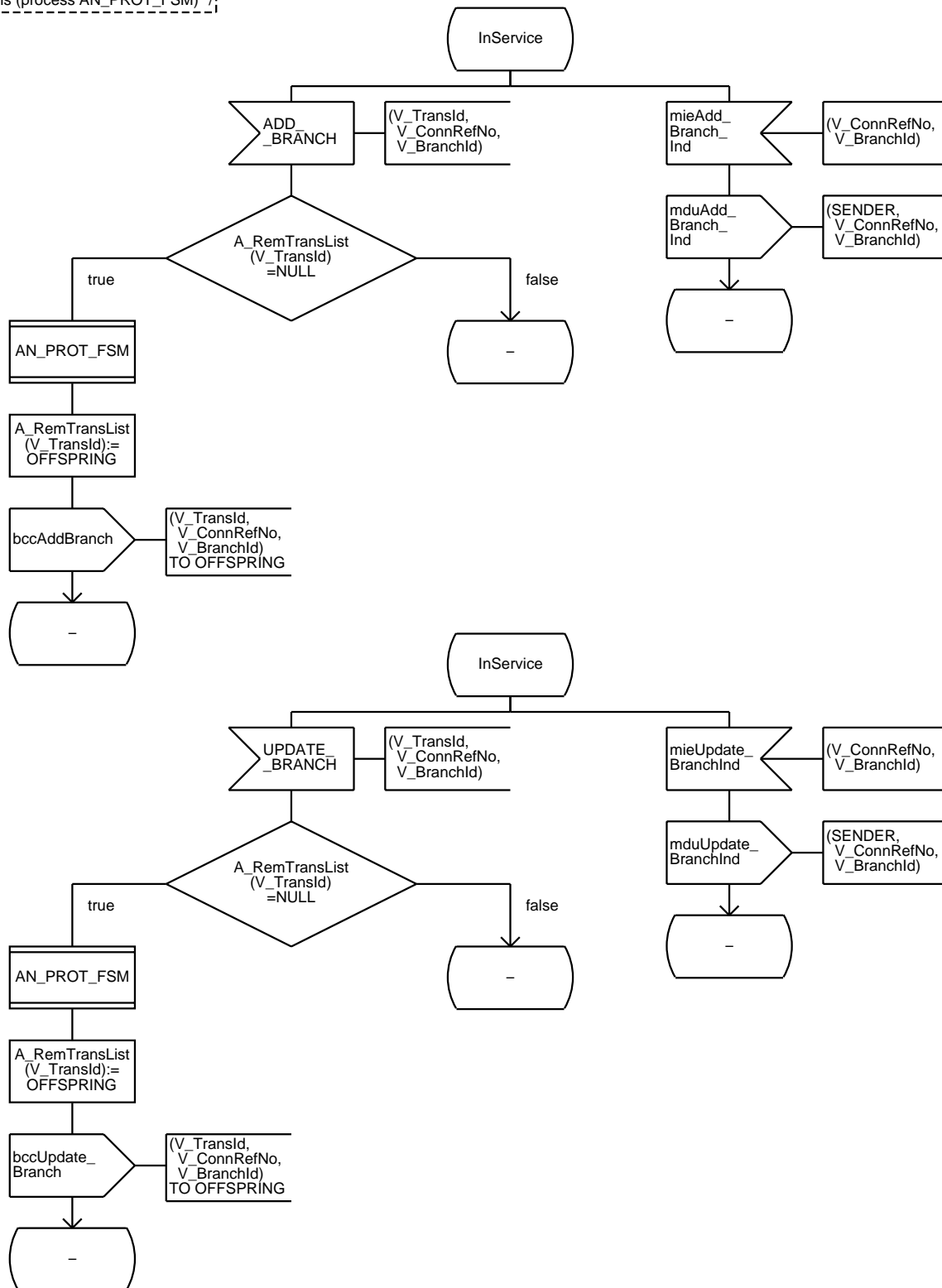
/*AN BCC protocol management entity
controls instantiation of
protocol FSMs (process AN_PROT_FSM) */



Process AN_PROT_MGR

10(22)

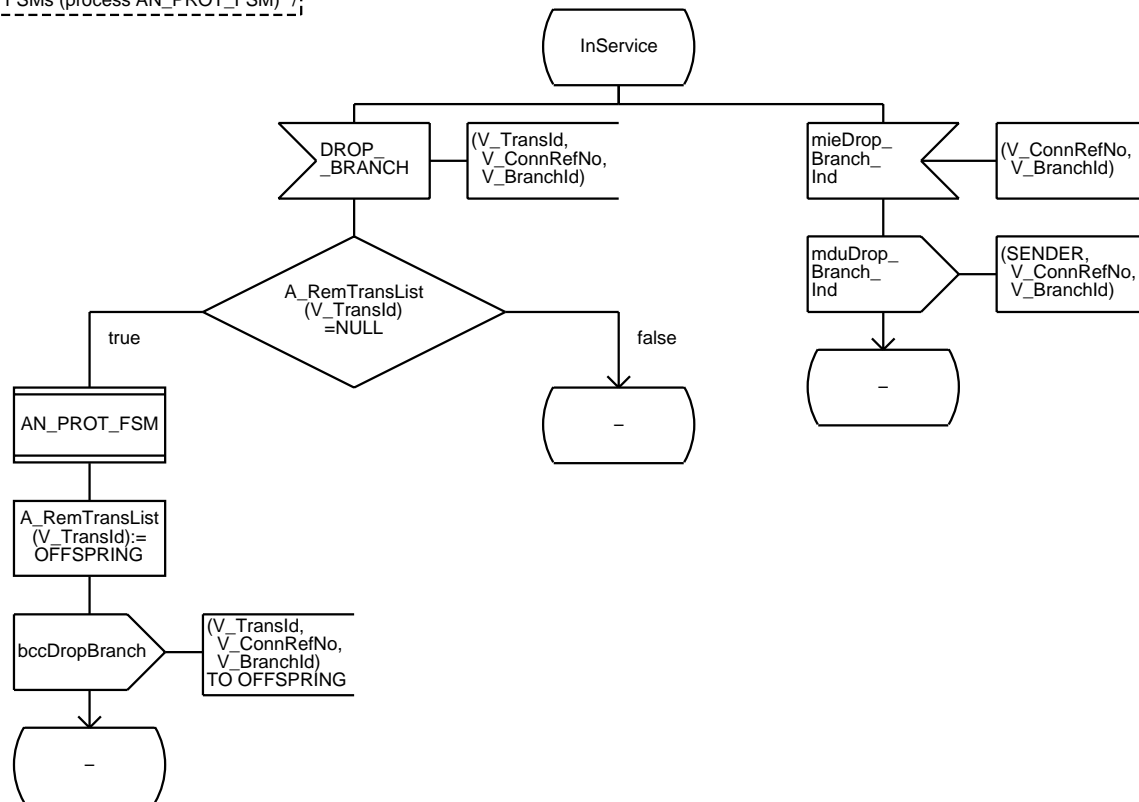
/* AN BCC protocol management entity
controls instantiation of
protocol FSMs (process AN_PROT_FSM) */



Process AN_PROT_MGR

11(22)

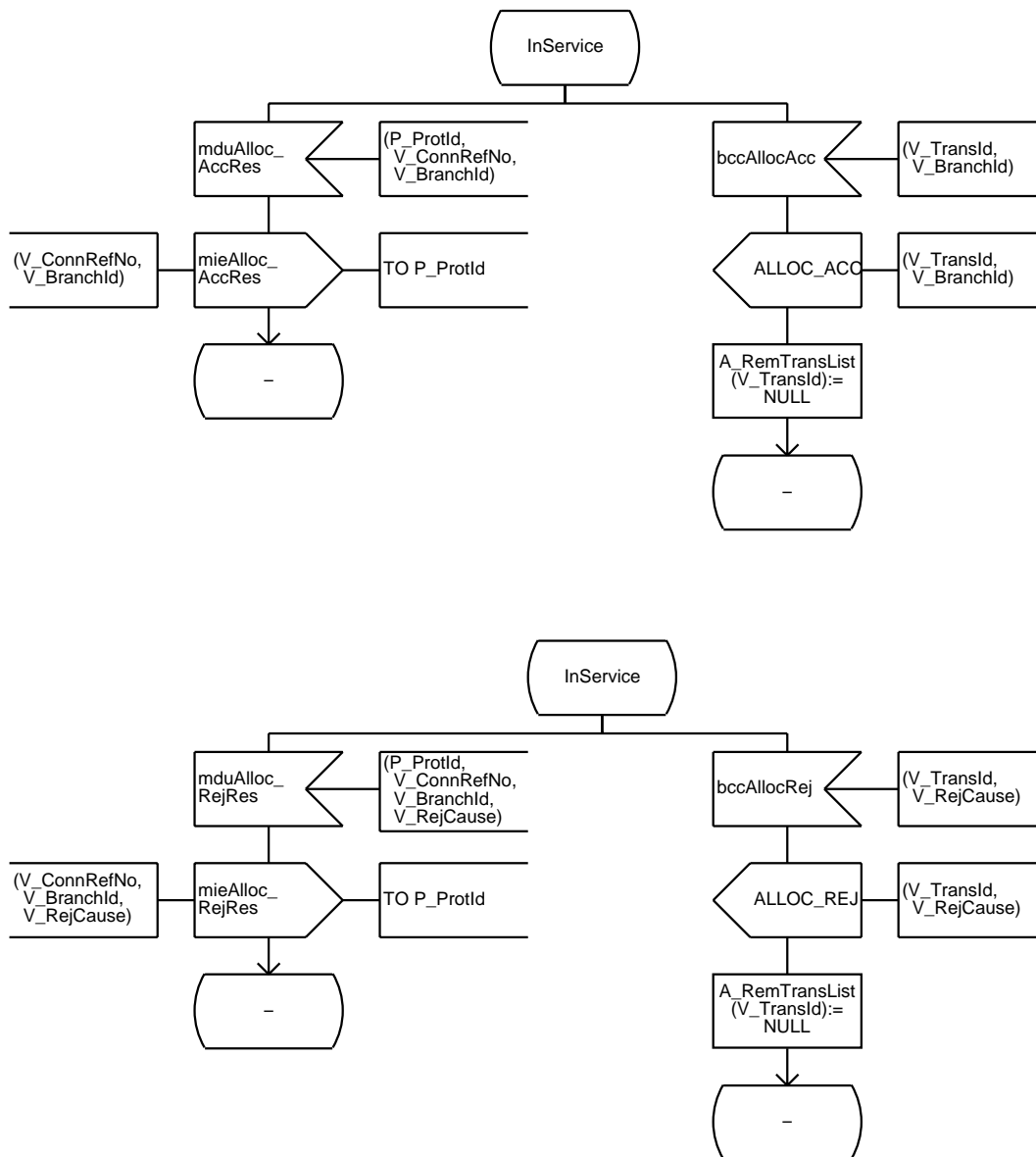
/*AN BCC protocol management entity
controls instantiation of
protocol FSMs (process AN_PROT_FSM) */



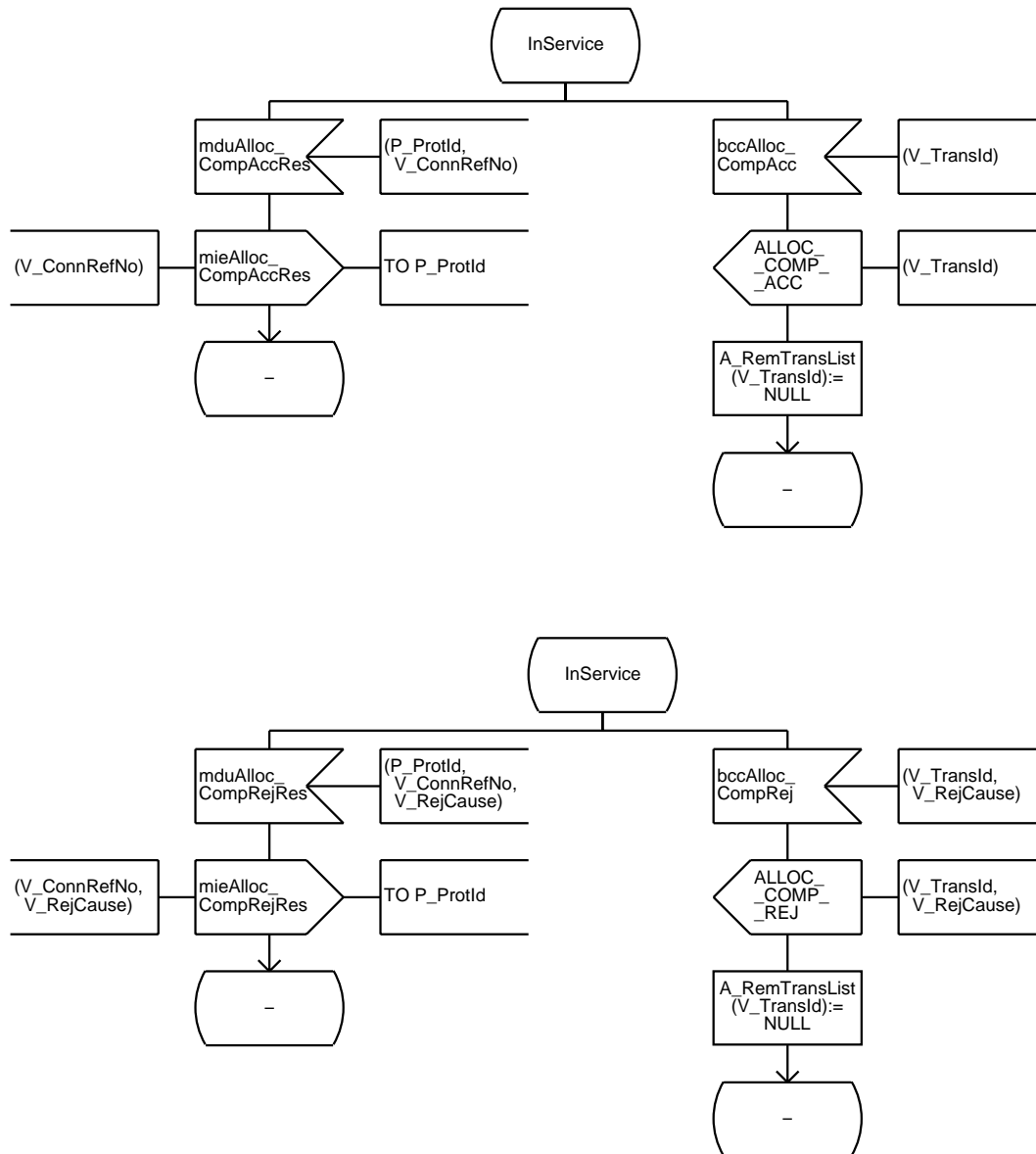
Process AN_PROT_MGR

12(22)

/*AN BCC protocol management entity
controls instantiation of
protocol FSMs (process AN_PROT_FSM) */



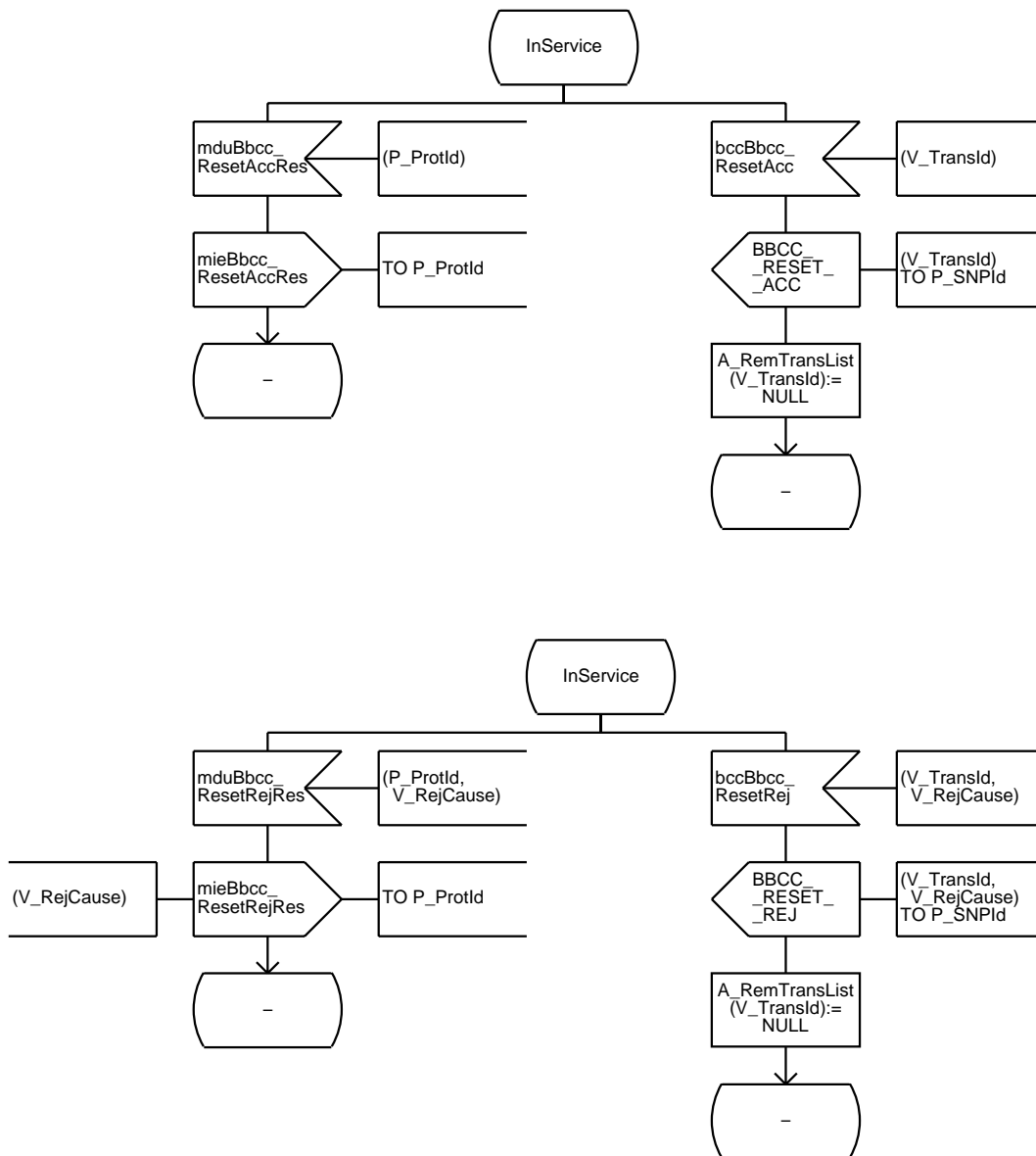
/*AN BCC protocol management entity
controls instantiation of
protocol FSMs (process AN_PROT_FSM) */



Process AN_PROT_MGR

14(22)

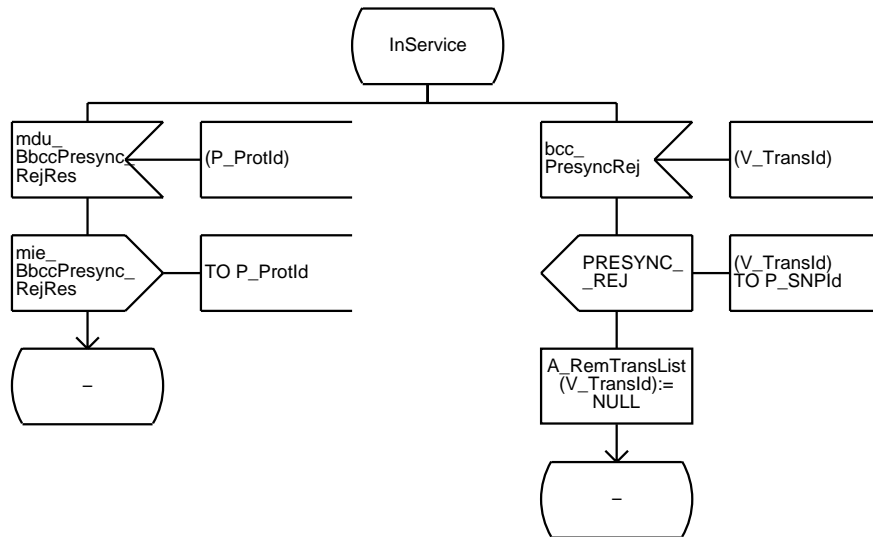
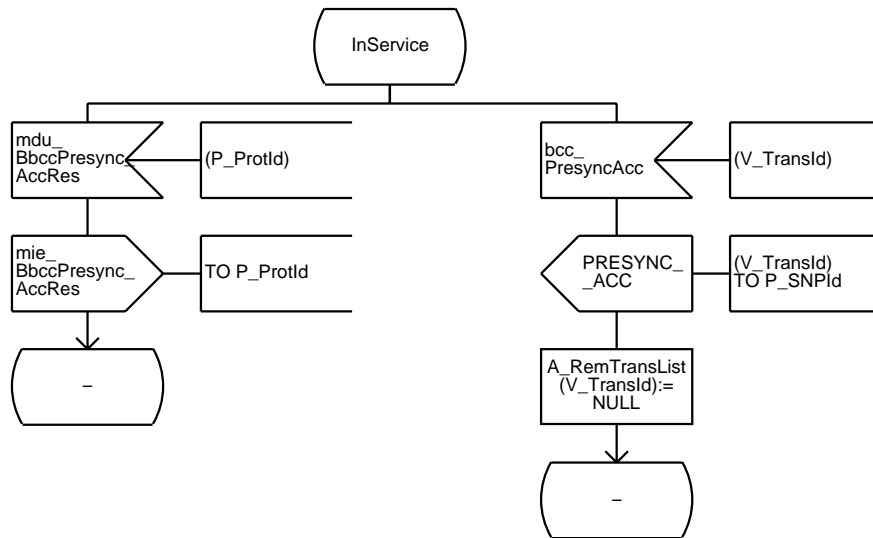
/*AN BCC protocol management entity
controls instantiation of
protocol FSMs (process AN_PROT_FSM) */



Process AN_PROT_MGR

15(22)

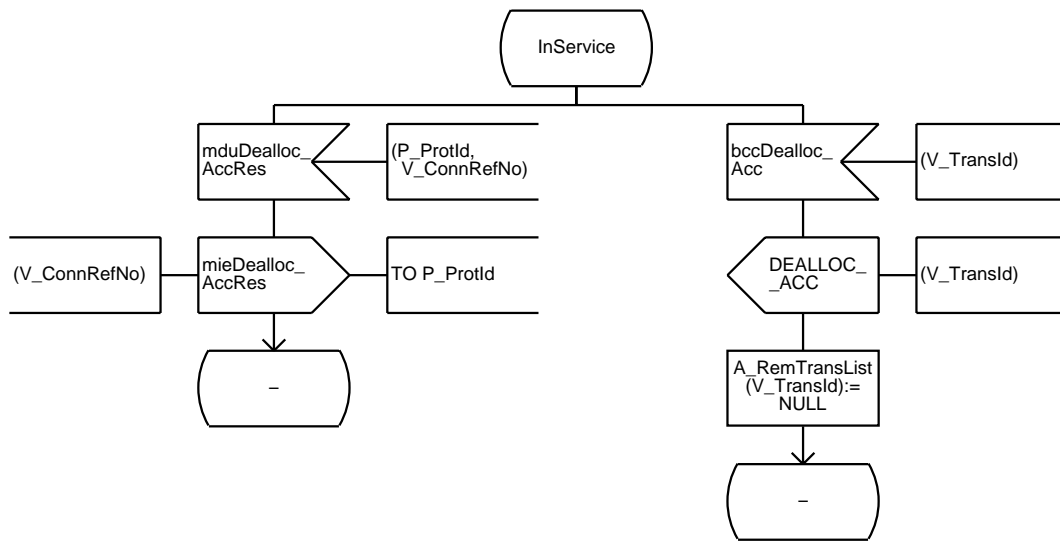
/*AN BCC protocol management entity
controls instantiation of
protocol FSMs (process AN_PROT_FSM) */



Process AN_PROT_MGR

16(22)

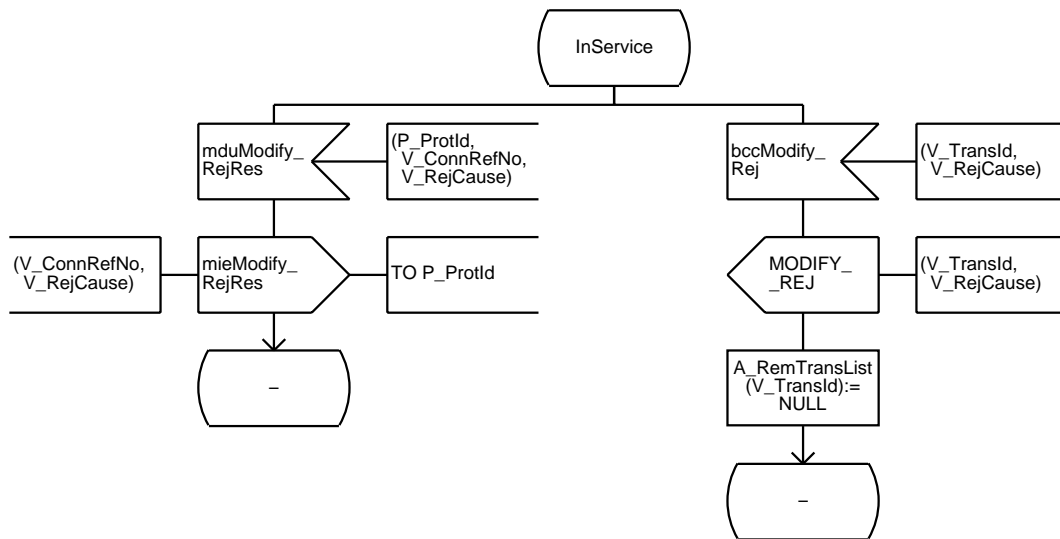
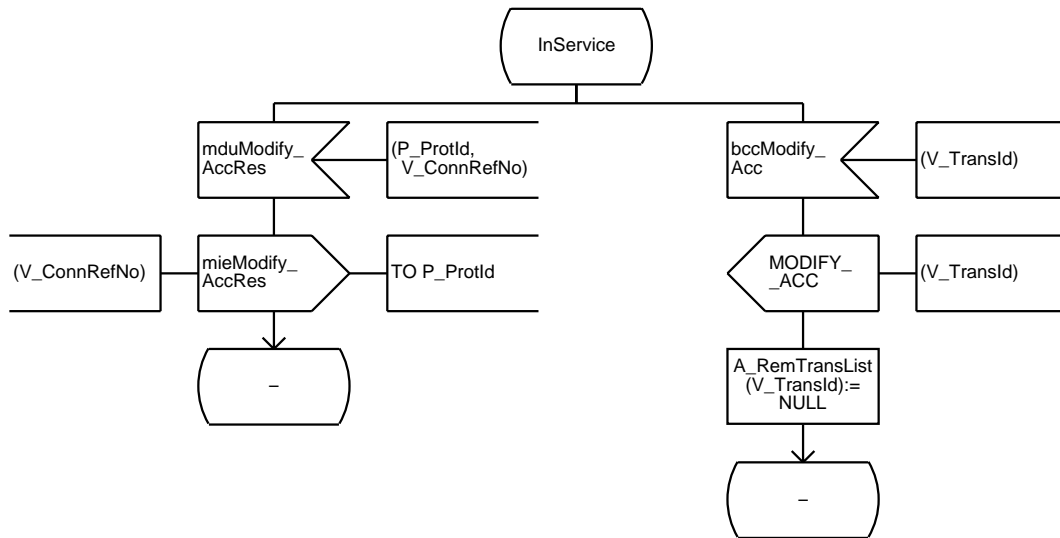
/*AN BCC protocol management entity
controls instantiation of
protocol FSMs (process AN_PROT_FSM) */



Process AN_PROT_MGR

17(22)

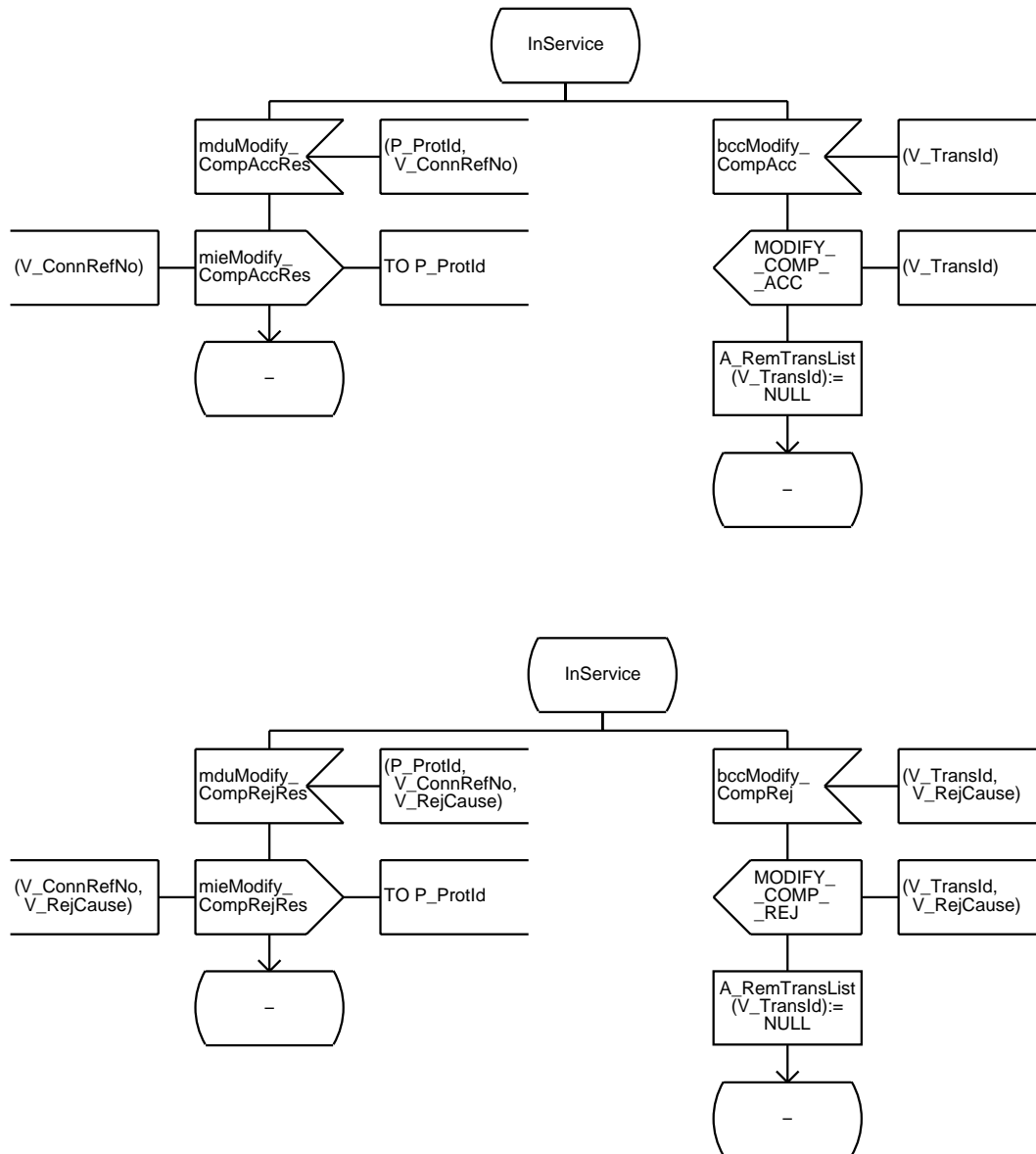
/*AN BCC protocol management entity
controls instantiation of
protocol FSMs (process AN_PROT_FSM) */



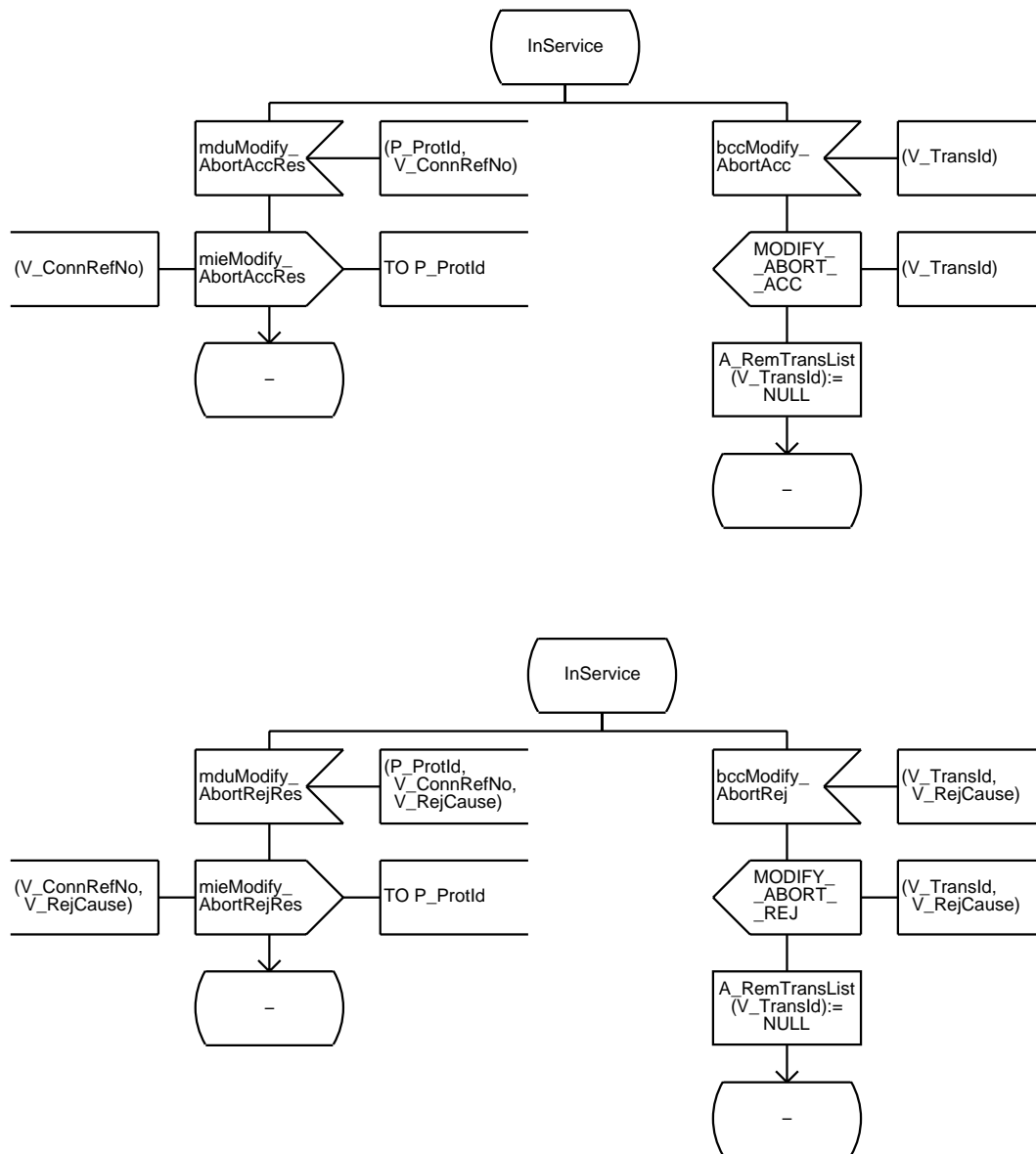
Process AN_PROT_MGR

18(22)

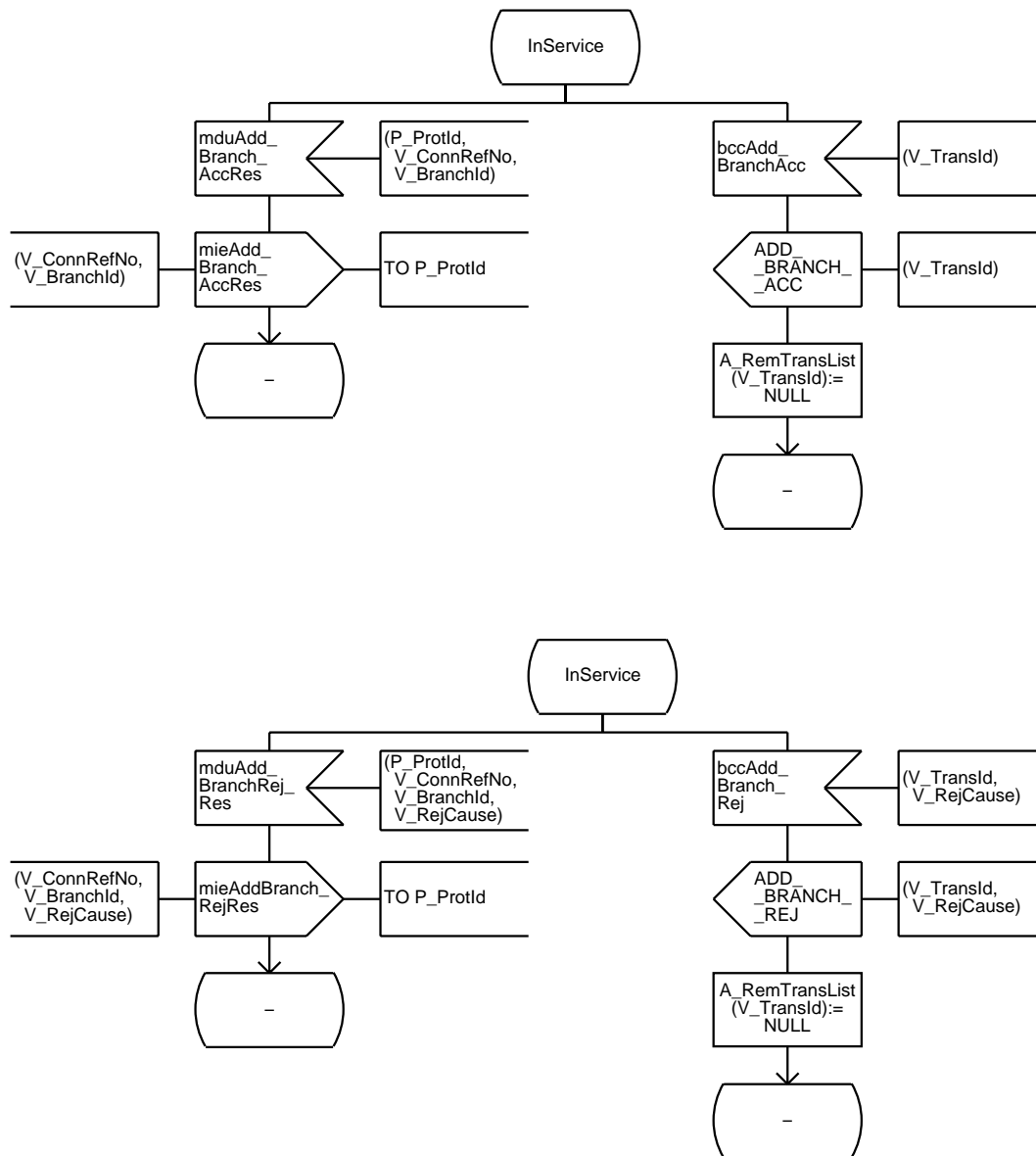
/*AN BCC protocol management entity
controls instantiation of
protocol FSMs (process AN_PROT_FSM) */



/*AN BCC protocol management entity
controls instantiation of
protocol FSMs (process AN_PROT_FSM) */



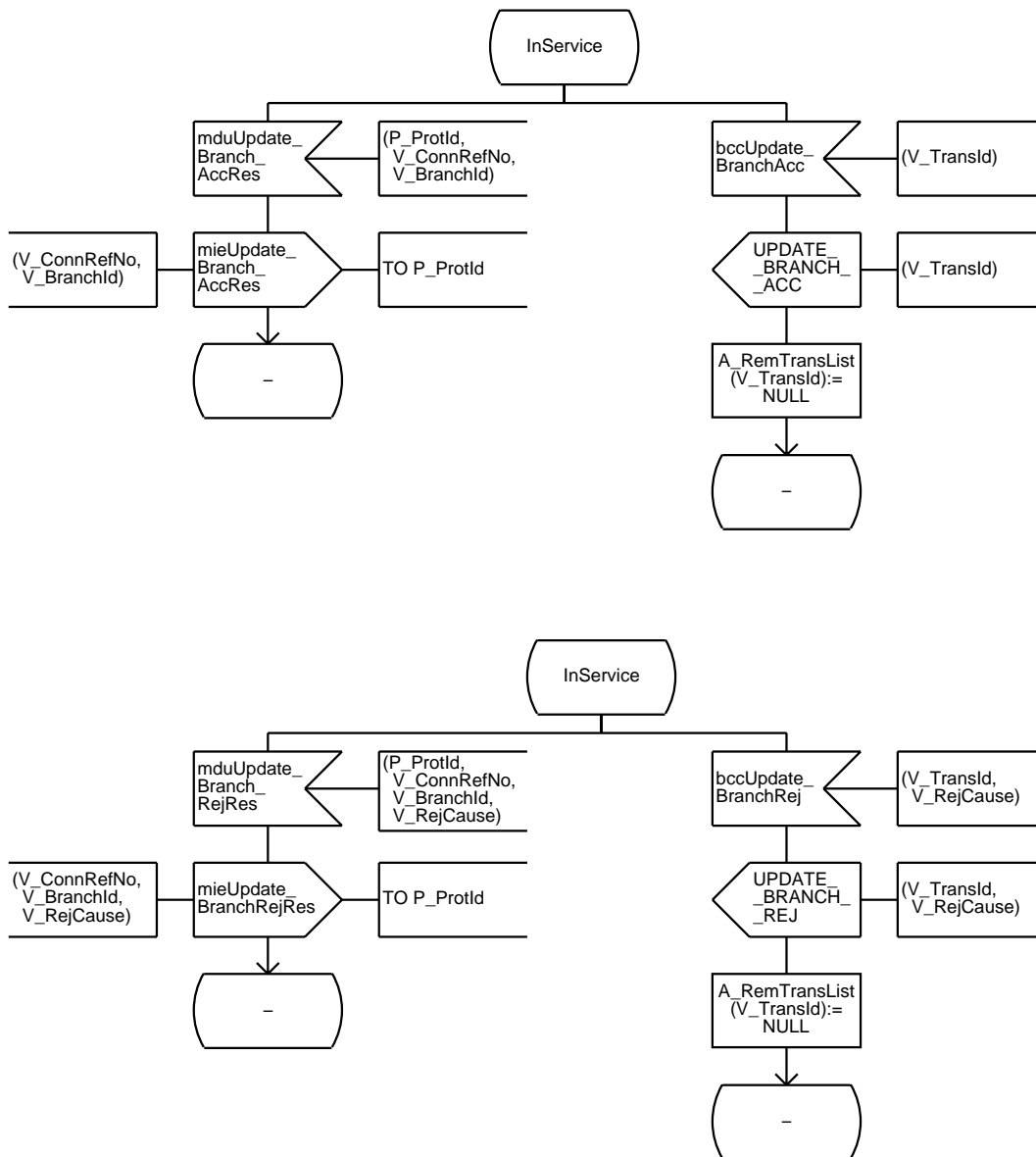
/*AN BCC protocol management entity
controls instantiation of
protocol FSMs (process AN_PROT_FSM) */



Process AN_PROT_MGR

21(22)

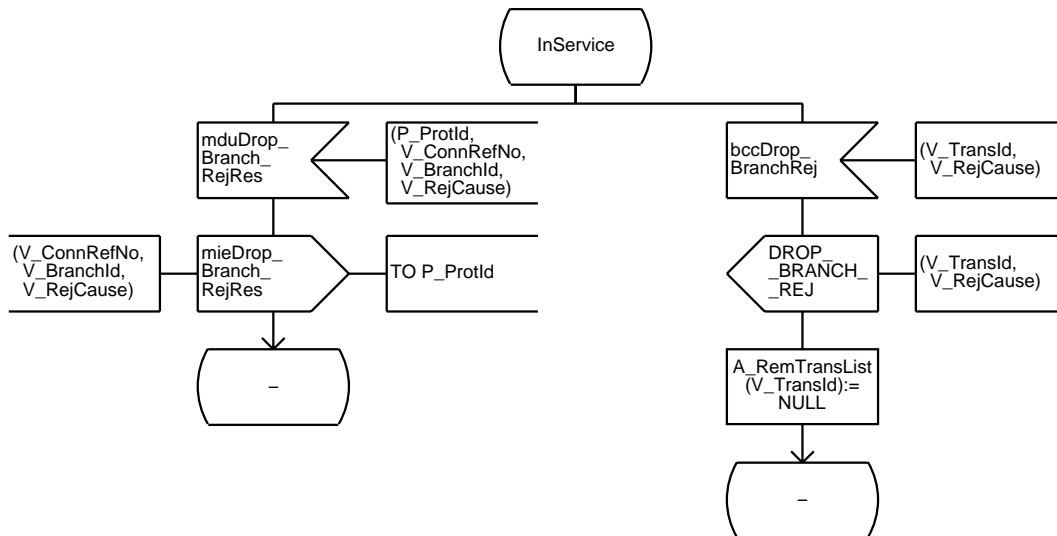
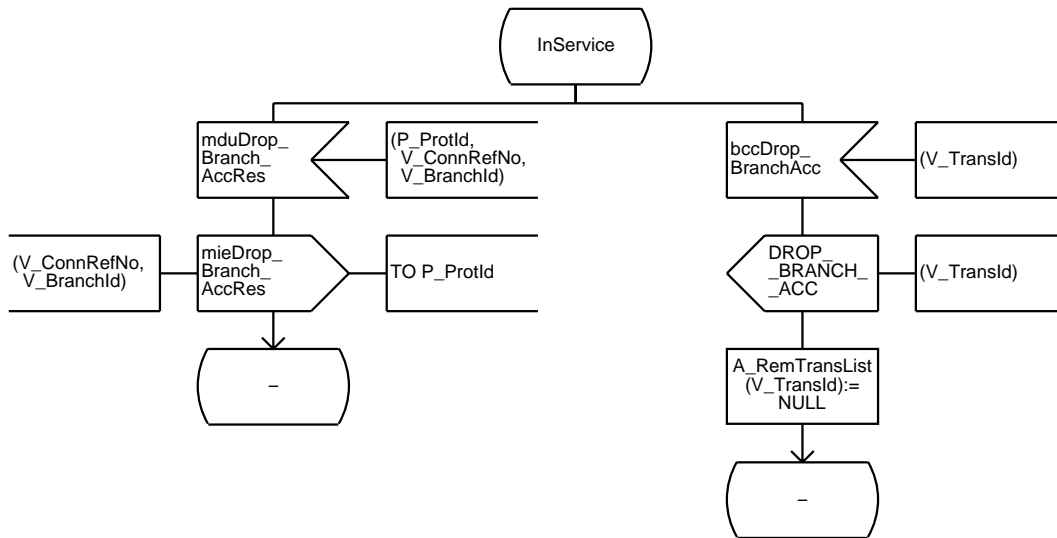
/*AN BCC protocol management entity
controls instantiation of
protocol FSMs (process AN_PROT_FSM) */



Process AN_PROT_MGR

22(22)

/*AN BCC protocol management entity
controls instantiation of
protocol FSMs (process AN_PROT_FSM) */



Process AN_PROT_FSM

1(15)

/*AN_PROT_FSM is responsible for sending
receiving BCC messages*/

/* Definitions and declarations for
AN_PROT_FSM */

/* State descriptions

InService: Init State of AN_PROT_FSM, process is waiting for initial AN_SYSMGT primitives or SN messages
AwaitLocAck: AN_PROT_FSM has passed a SN request to AN_SYSMGT and is now awaiting the AN_SYSMGT response
AwaitRemAck: AN_PROT_FSM has sent a VB5 message to the SN and is now awaiting the SN response
*/

/* Timerdefinitions */

TIMER

T_AnFault := 15; /* Default value for AN_FAULT supervision timer is 10 sec, Tolerance +/- 10% */

/* AN_PROT_FSM internal variables and constants*/

/**/

/* timer handling */

DCL V_ExpCount NATURAL; /*number of actual message repetitions*/

/*possible values*/

SYNONYM C_ExpMax INTEGER = 1; /*maximum number of message repetitions is 1*/

/**/

/*syntax check handling*/

DCL V_SynResult IT_SynResult; /*syntax check result, provided by procedure IE_CHECK*/

/* Signal data declarations*/

/**/

/* Transaction identifier from/to SN*/

DCL

V_TransId IT_TransId;

/**/

/* Connection reference number */

DCL

V_ConnRefNo IT_ConnRefNo;

/**/

/* Branch identifier */

DCL

V_BranchId IT_BranchId;

/**/

/* Reject cause */

DCL

V_RejCause IT_RejCause;

/**/

/* Protocol error cause */

DCL

V_ProtErrCause IT_ProtErrCause;

/**/

/* Response */

DCL

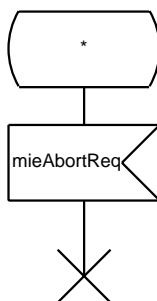
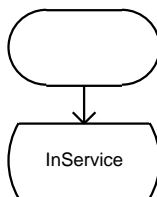
V_Response IT_Response;

/* Internal data declarations */

Process AN_PROT_FSM

2(15)

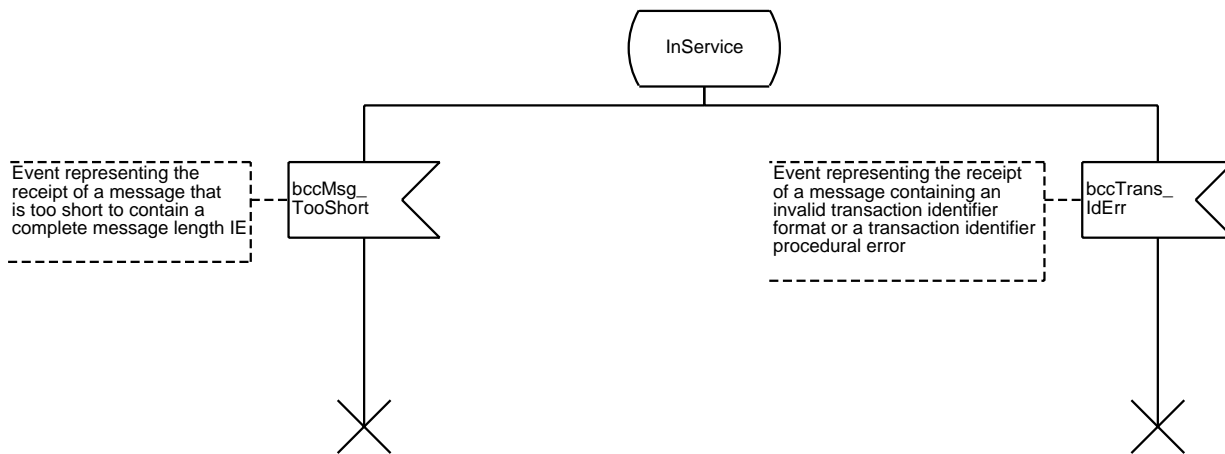
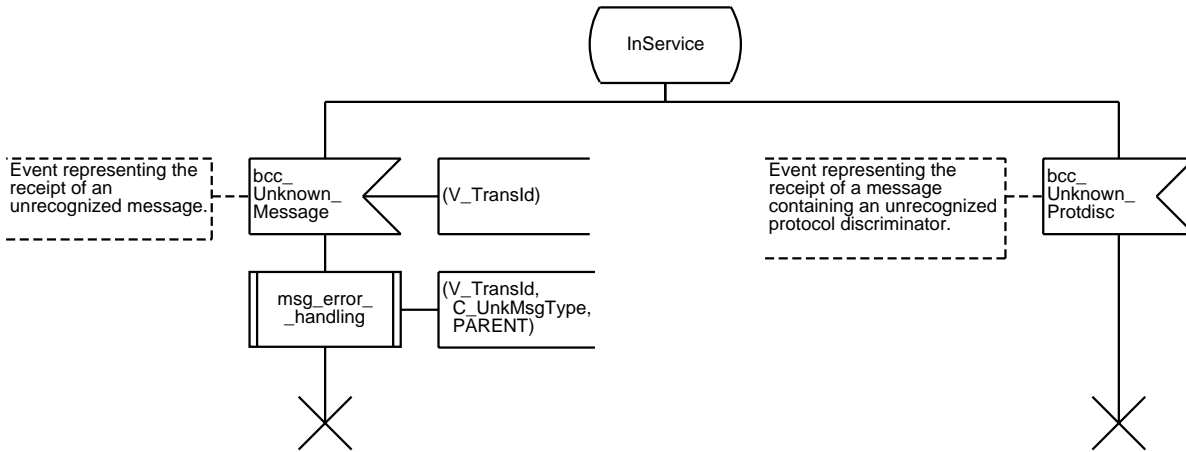
/*AN_PROT_FSM is responsible for sending
receiving BCC messages*/



Process AN_PROT_FSM

3(15)

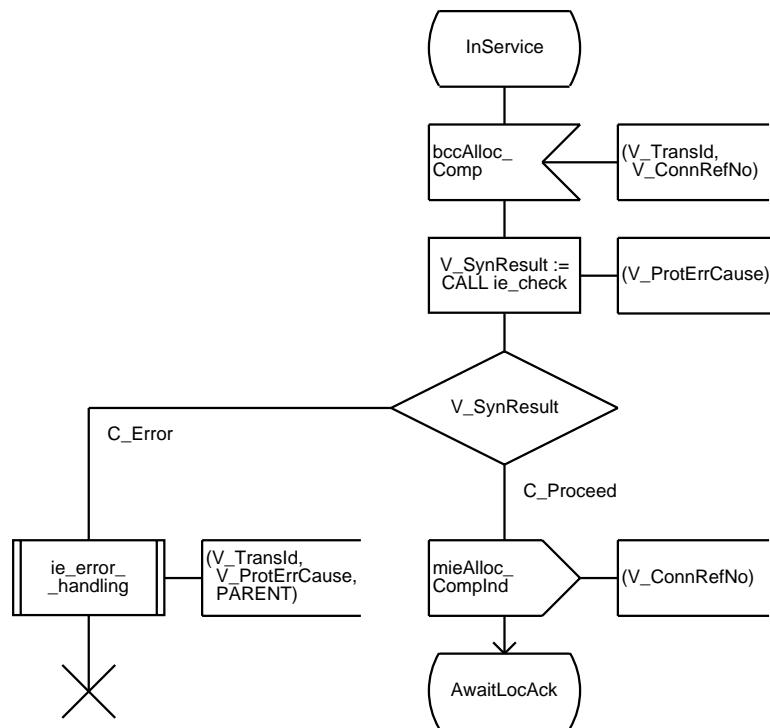
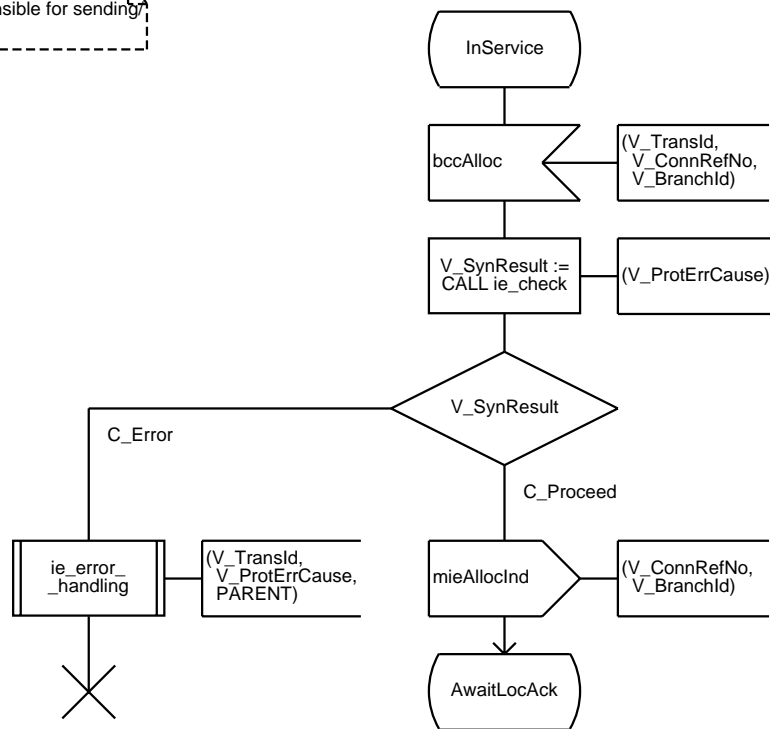
/*AN_PROT_FSM is responsible for sending/
receiving BCC messages*/



Process AN_PROT_FSM

4(15)

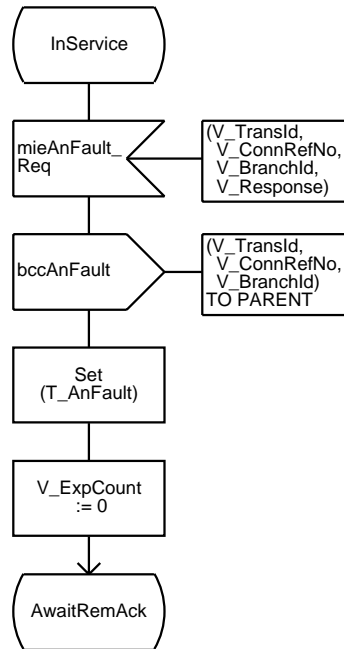
/*AN_PROT_FSM is responsible for sending
receiving BCC messages*/



Process AN_PROT_FSM

5(15)

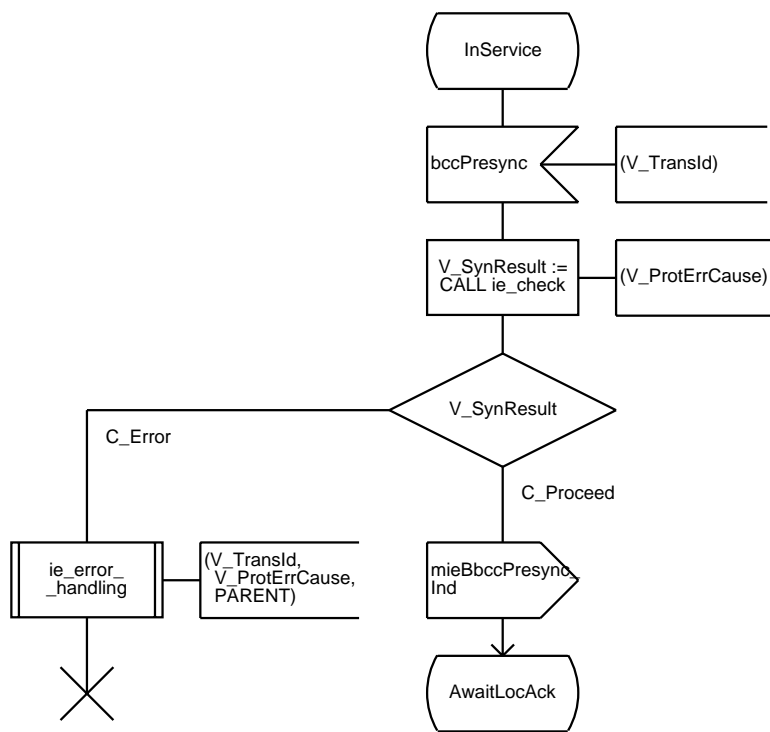
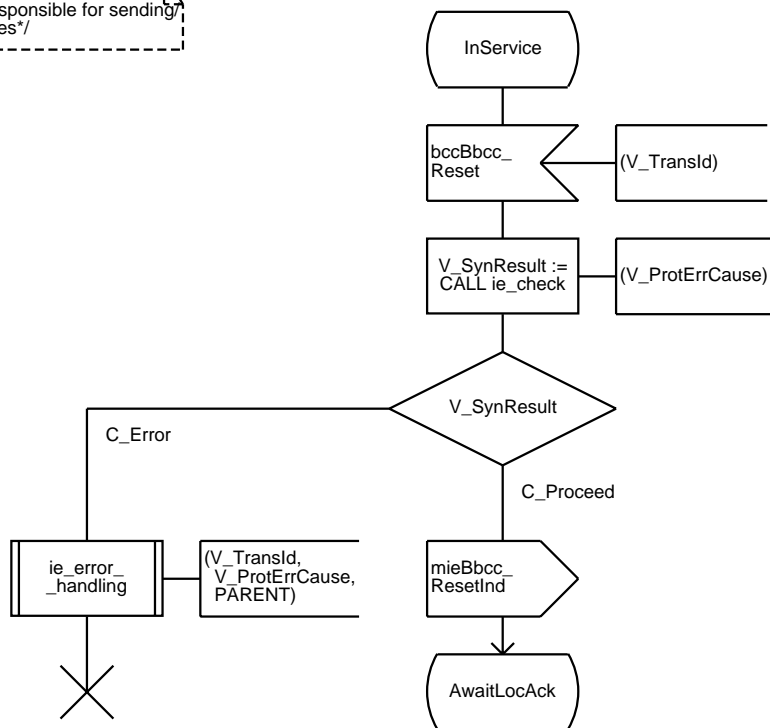
/*AN_PROT_FSM is responsible for sending/
receiving BCC messages*/



Process AN_PROT_FSM

6(15)

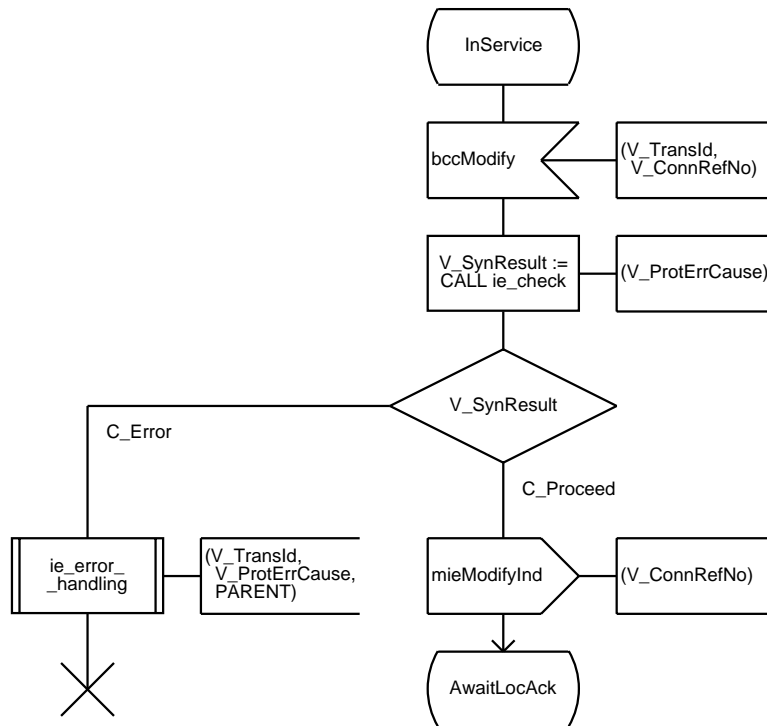
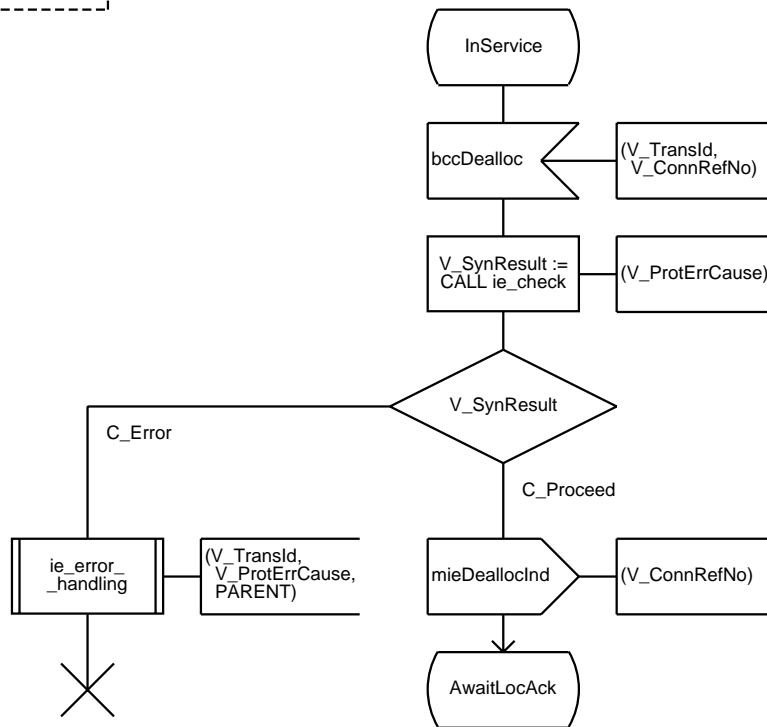
/*AN_PROT_FSM is responsible for sending/
receiving BCC messages*/



Process AN_PROT_FSM

7(15)

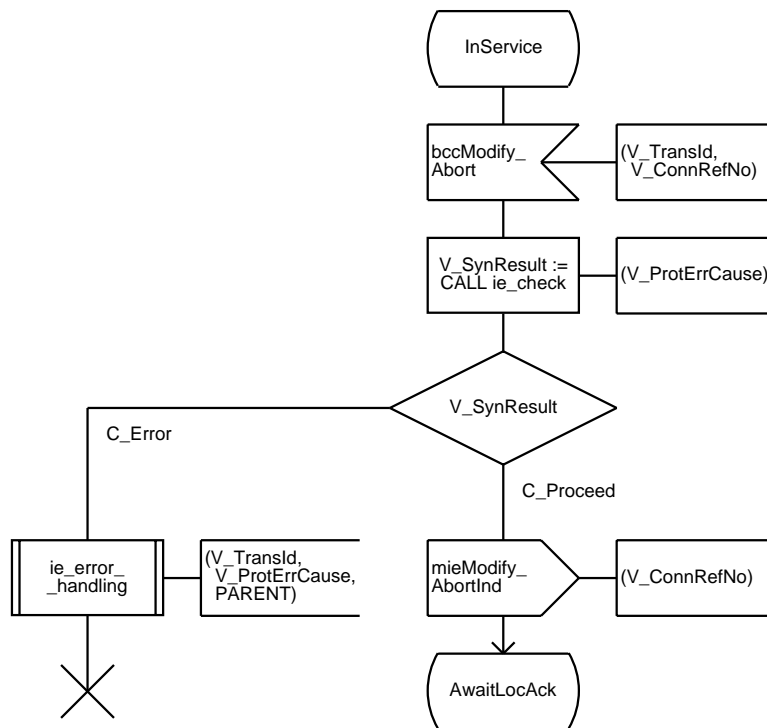
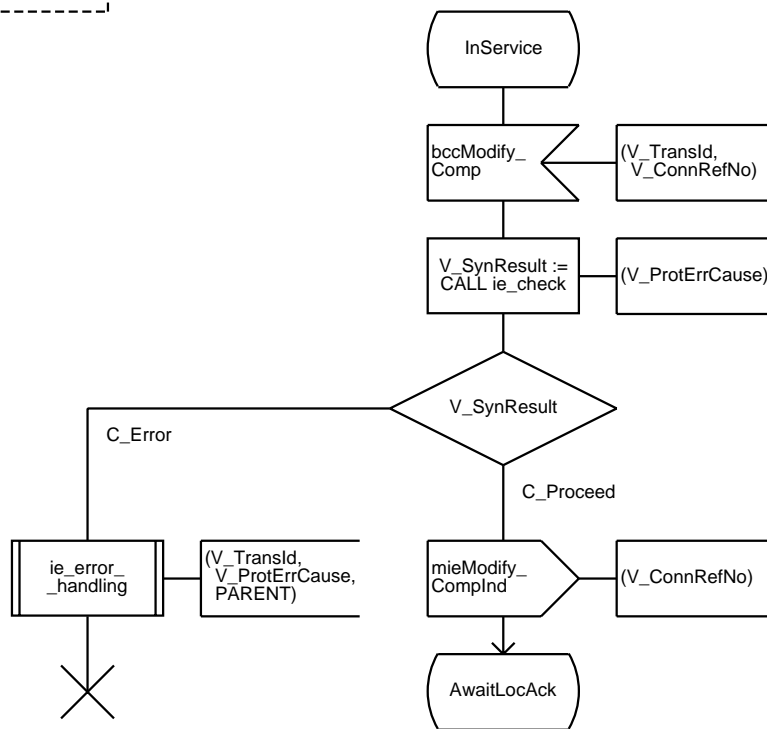
/*AN_PROT_FSM is responsible for sending/
receiving BCC messages*/



Process AN_PROT_FSM

8(15)

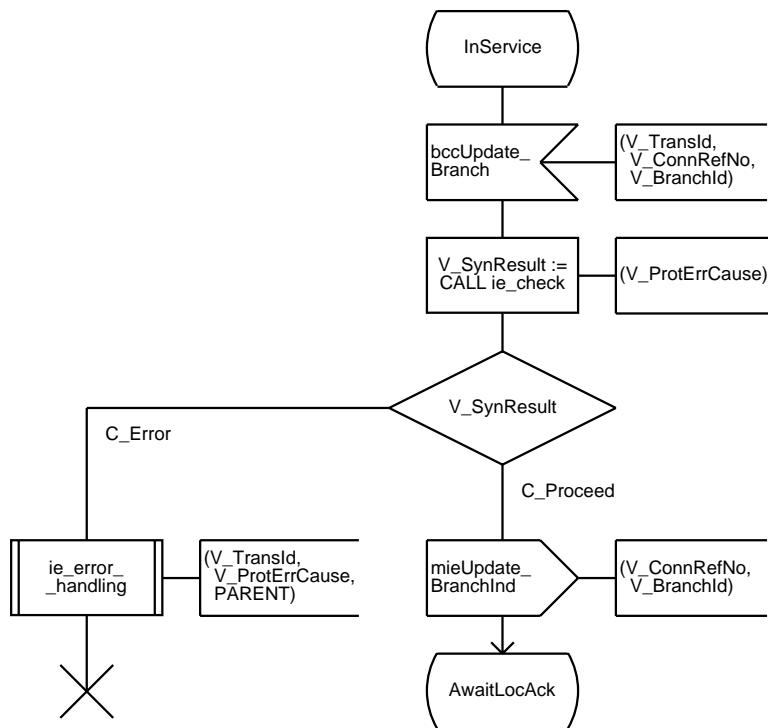
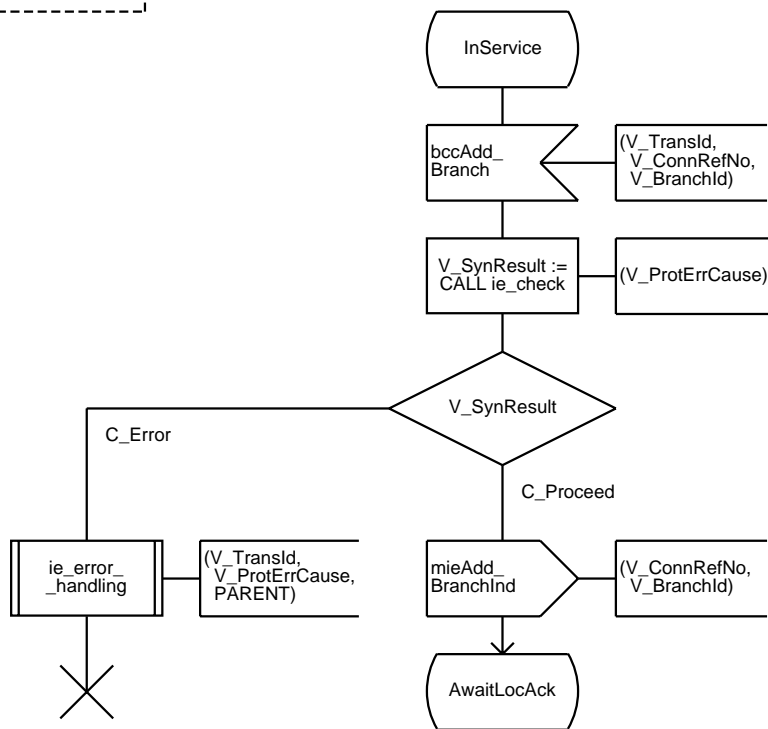
/*AN_PROT_FSM is responsible for sending/
receiving BCC messages*/



Process AN_PROT_FSM

9(15)

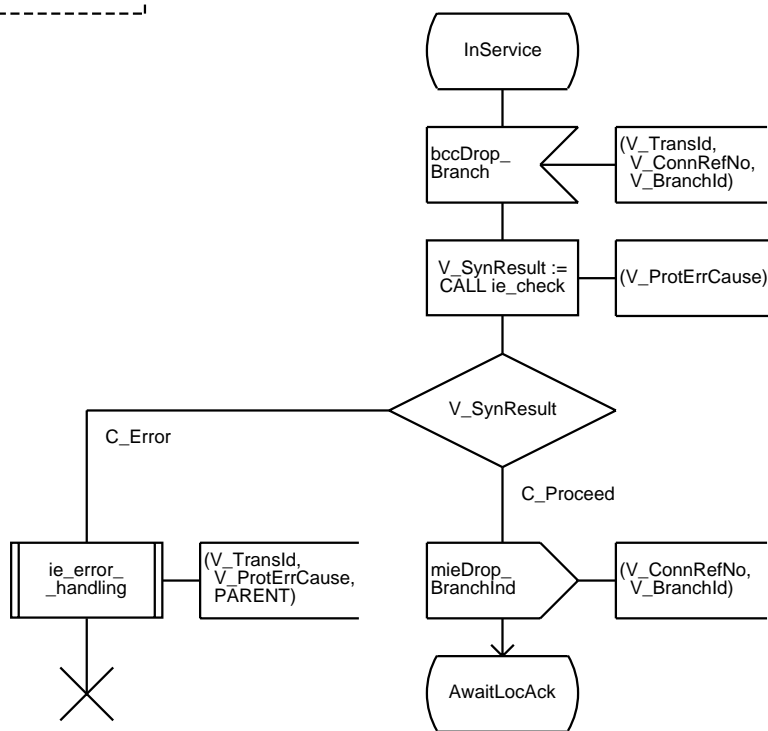
/*AN_PROT_FSM is responsible for sending/
receiving BCC messages*/



Process AN_PROT_FSM

10(15)

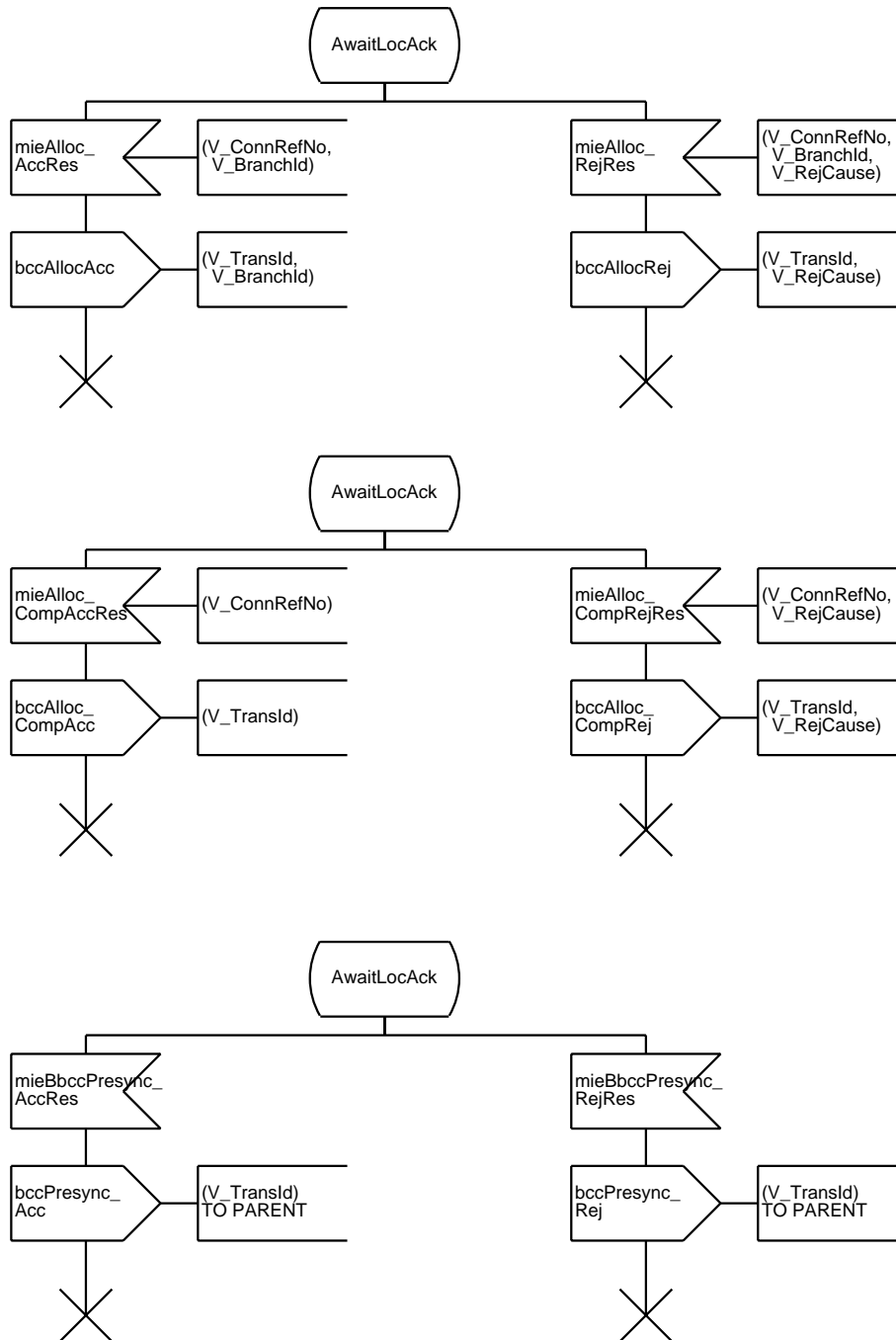
/*AN_PROT_FSM is responsible for sending/
receiving BCC messages*/



Process AN_PROT_FSM

11(15)

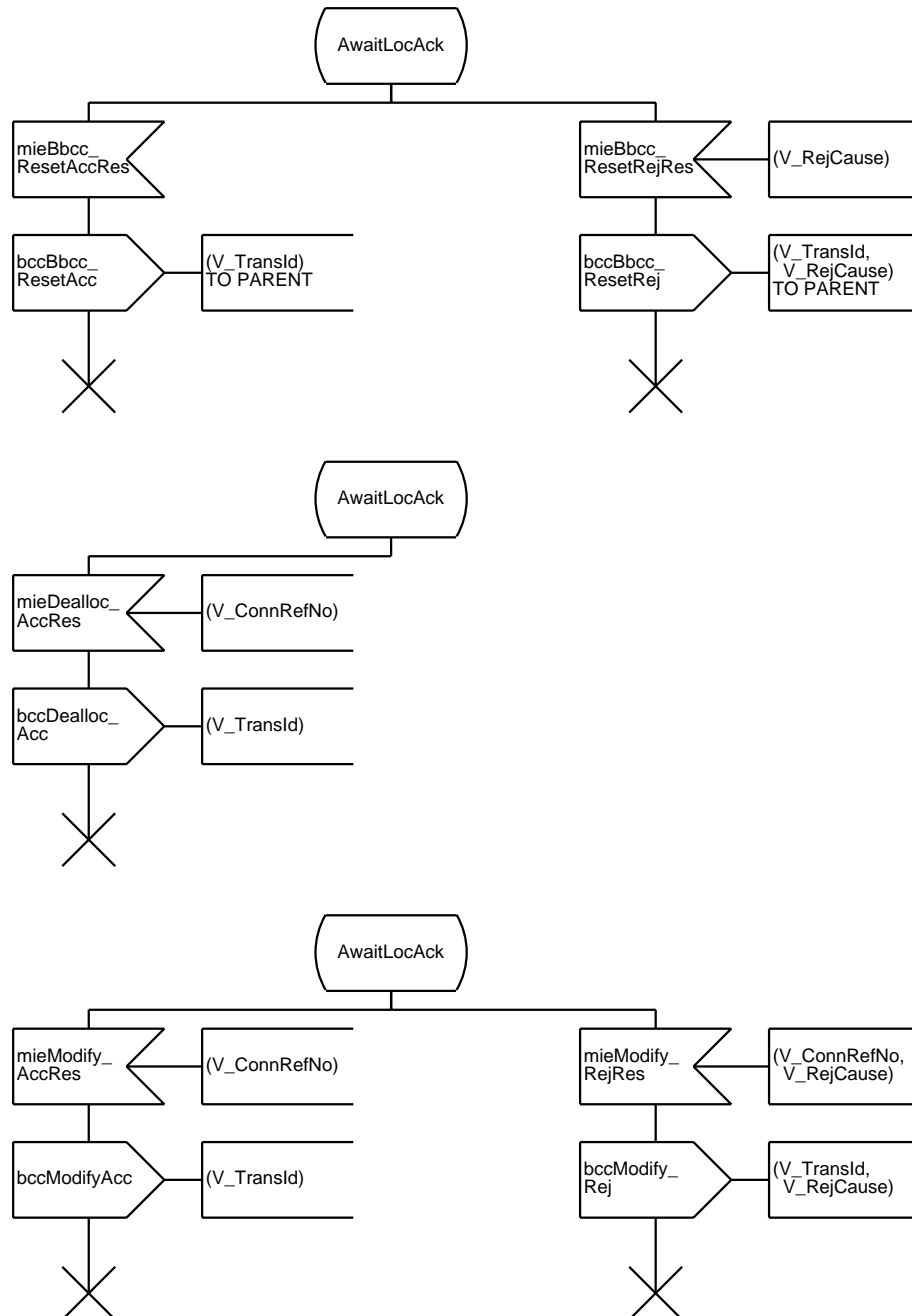
/*AN_PROT_FSM is responsible for sending/
receiving BCC messages*/



Process AN_PROT_FSM

12(15)

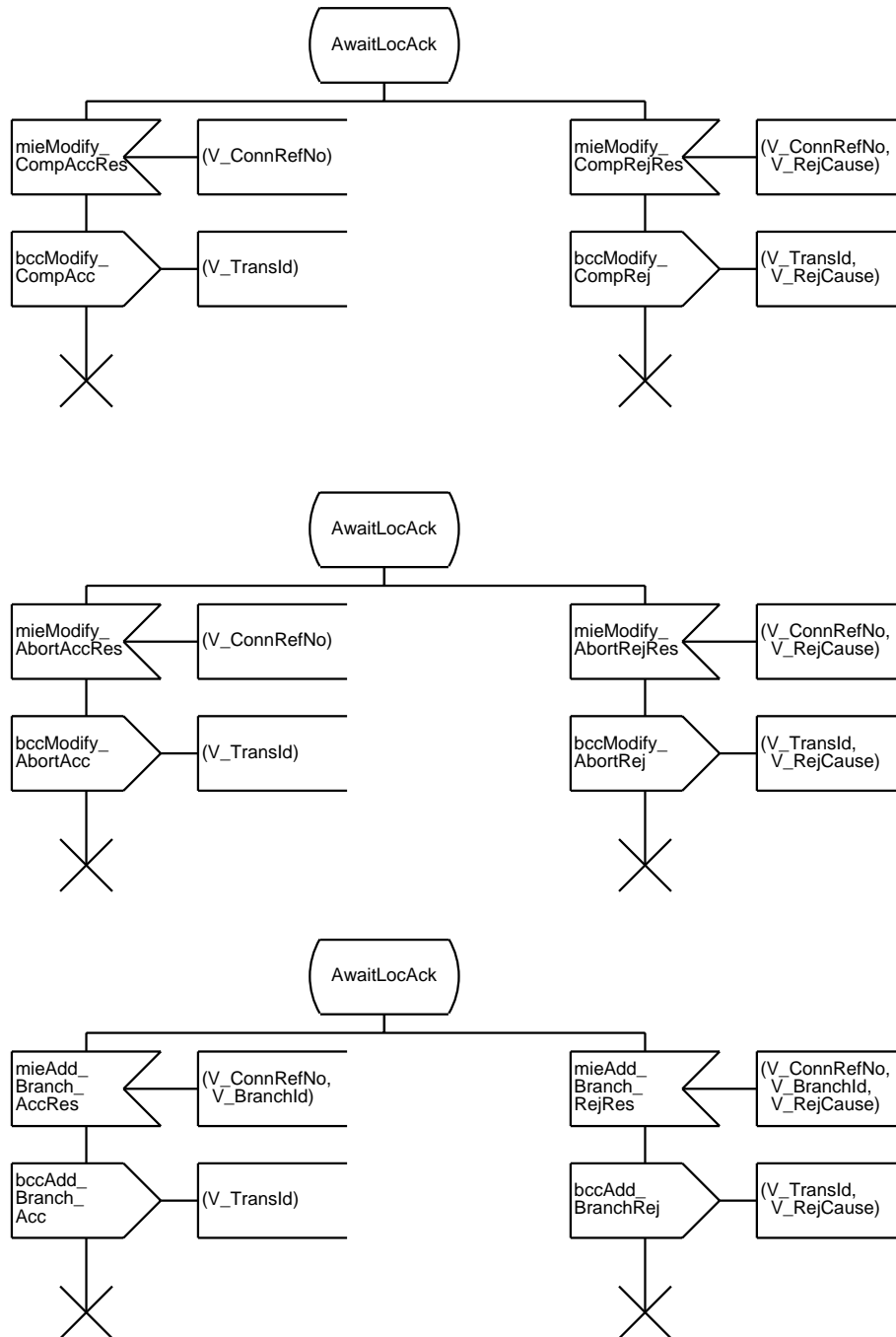
/*AN_PROT_FSM is responsible for sending/
receiving BCC messages*/



Process AN_PROT_FSM

13(15)

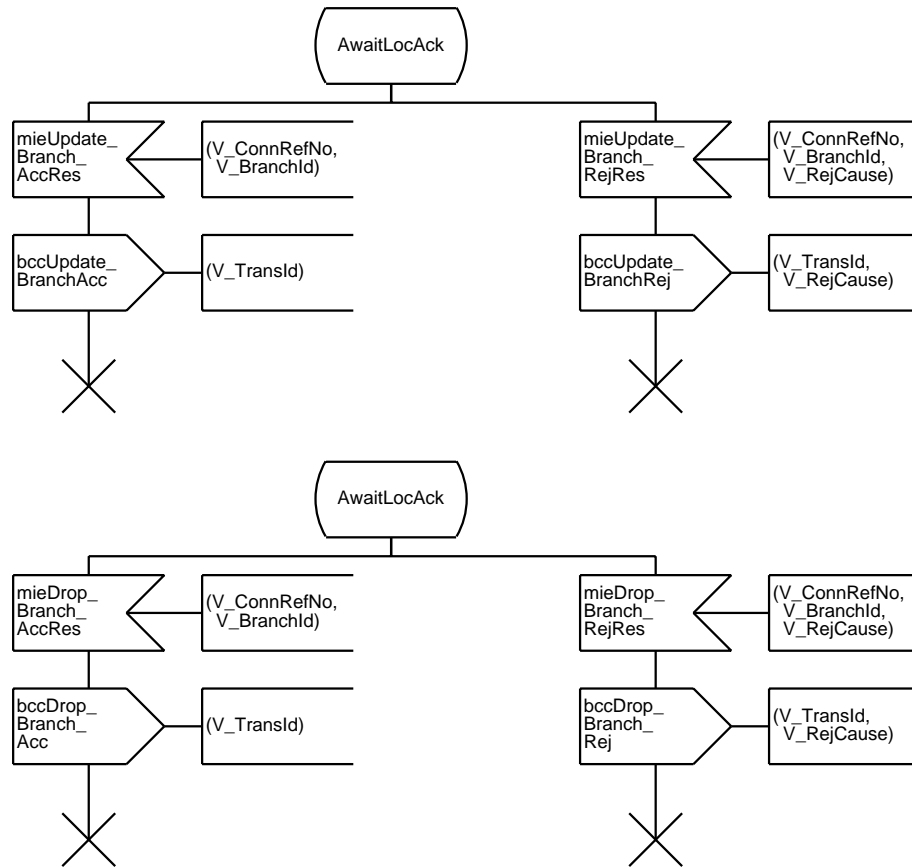
*AN_PROT_FSM is responsible for sending/
receiving BCC messages*



Process AN_PROT_FSM

14(15)

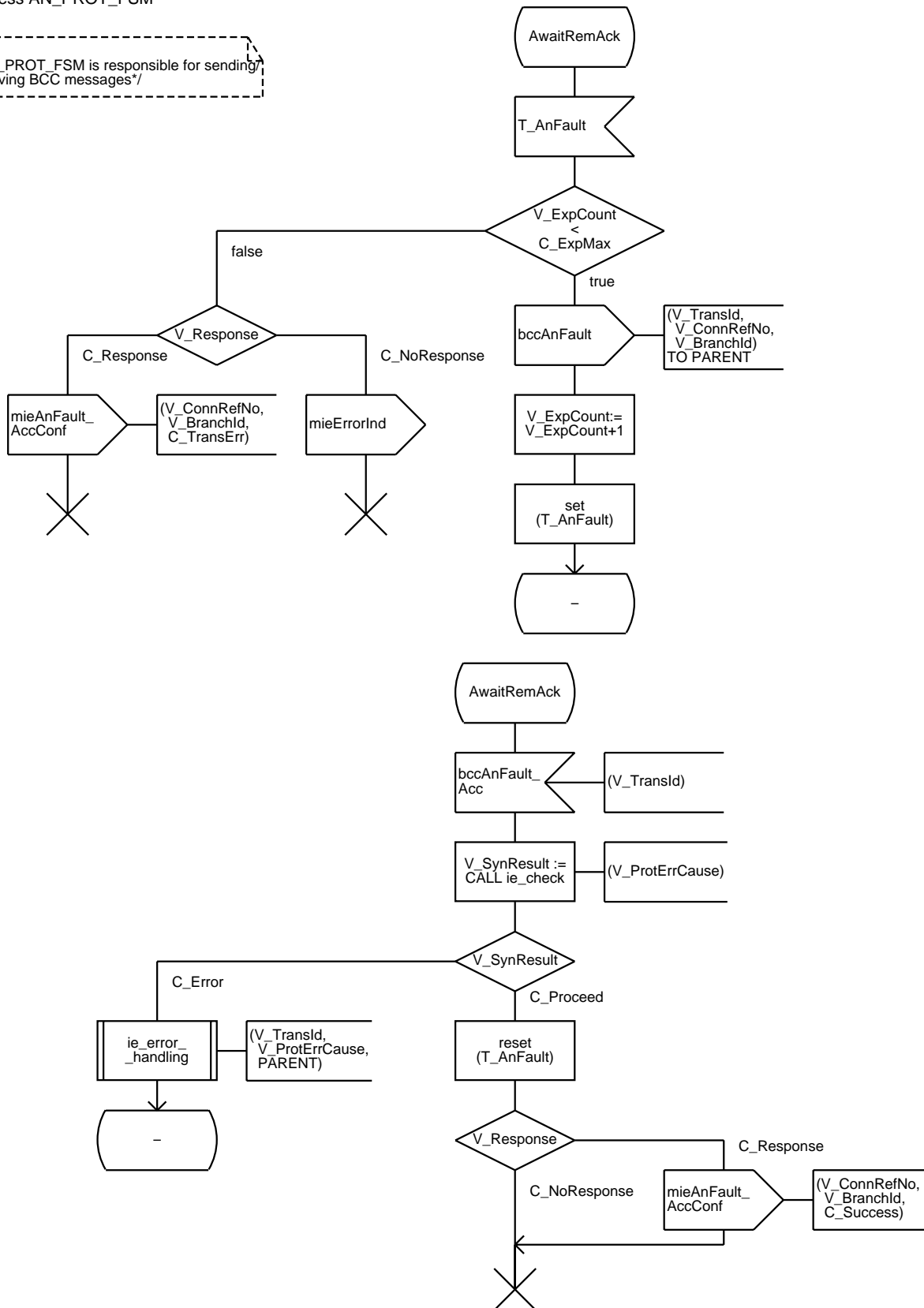
/*AN_PROT_FSM is responsible for sending/
receiving BCC messages*/

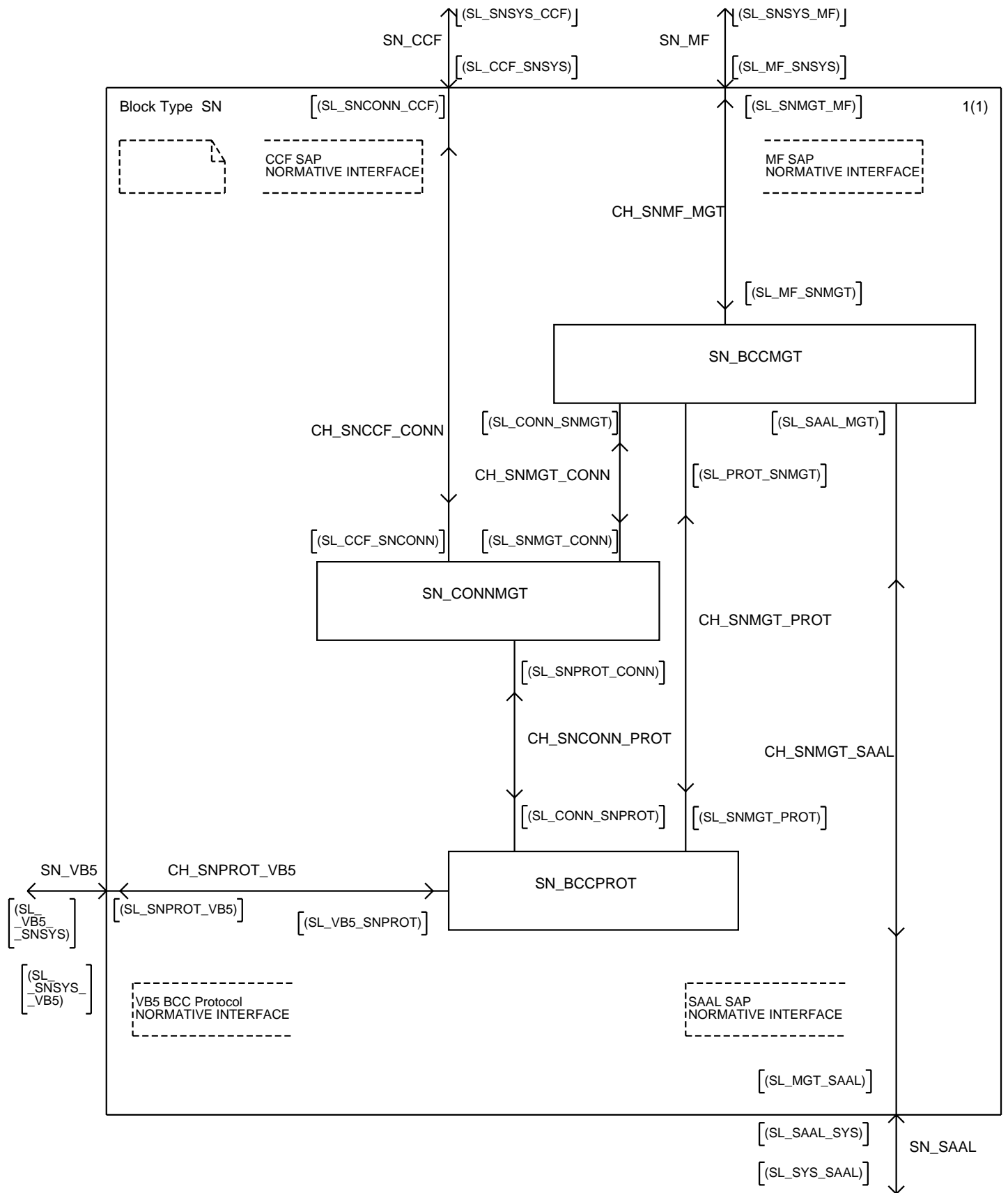


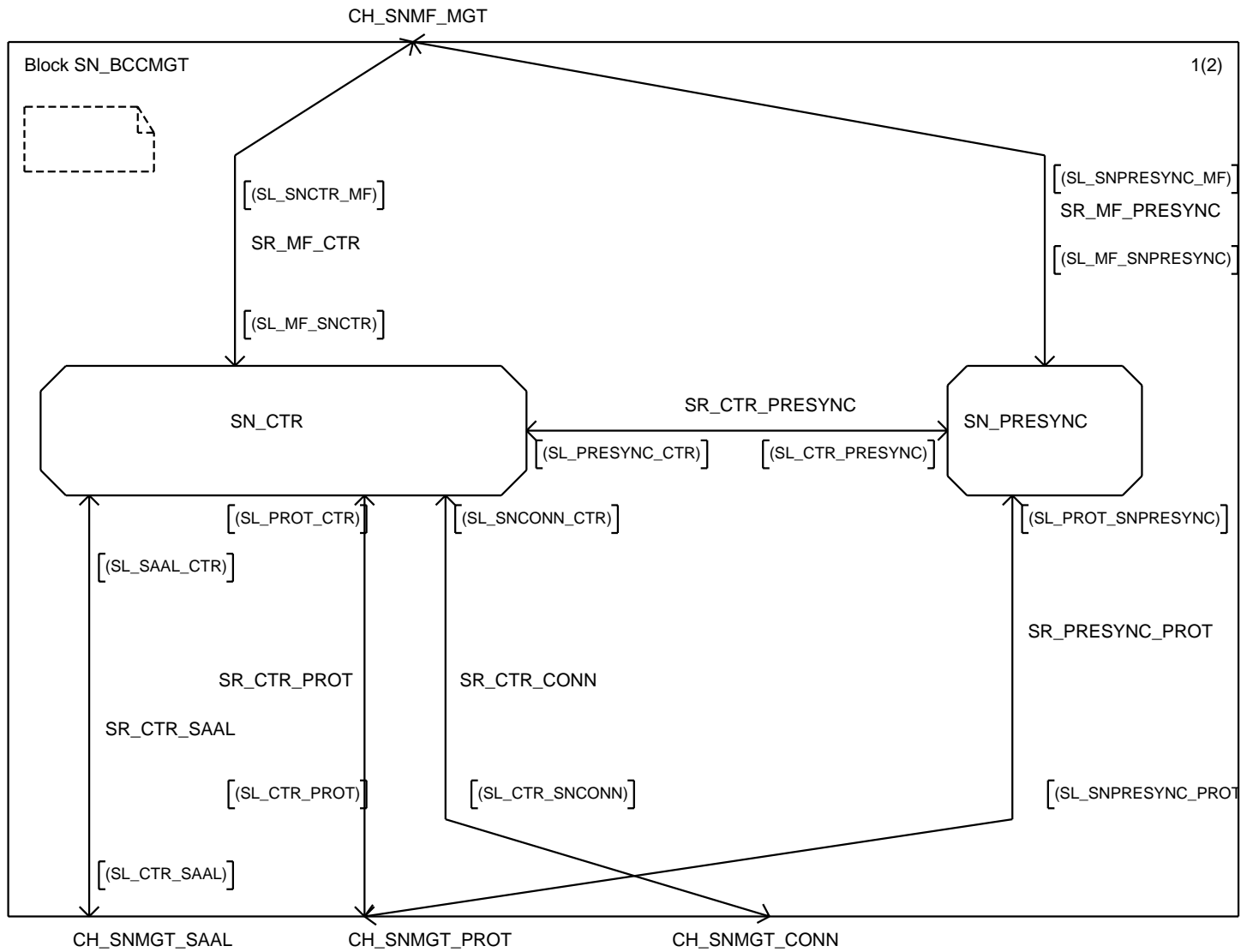
Process AN_PROT_FSM

15(15)

/*AN_PROT_FSM is responsible for sending/
receiving BCC messages*/







Block SN_BCCMGT

2(2)



/* Declaration of primitives internal
to block SN_BCCMGT */

signal
mieBbccPresyncReq,
mieBbccPresyncAccConf,
mieBbccPresyncRejConf (IT_RejCause);

/* Declaration of signallists
internal to block SN_CONNMGT */

signallist SL_PRESYNC_CTR =
mieBbccPresyncAccConf,
mieBbccPresyncRejConf;

signallist SL_CTR_PRESYNC =
mieBbccPresyncReq,
mieStartTraffic,
mieStopTraffic;

Process SN_CTR

1(8)

```

/* VB5.2 BCC management entity
Process CTR */

```

```

/* State descriptions
OutOfService:      Initial state, no SAAL connection available
RstrtAwaitEstConf: Process is waiting for establish confirmation of the peer entity, restart procedure
StrtpAwaitEstConf: Process is waiting for establish confirmation of the peer entity, startup procedure
SaalEst:           SAAL in operation
Presync:           Process is waiting for presync confirmation, restart procedure
RstrtReset:        Process is waiting for reset confirmation, restart procedure
StrtpReset:        Process is waiting for reset confirmation, startup procedure
InService:         Interface is in operational state
AwaitRelConf:      Process is waiting for release confirmation of the peer entity
*/

```

```

/* Timerdefinitions */
TIMER
T_BbccStartup := 180; /* Default value for startup supervision timer is 180 sec, Tolerance +/- 10% */
T_BbccRestart := 180; /* Default value for restart supervision timer is 180 sec, Tolerance +/- 10% */

```

```

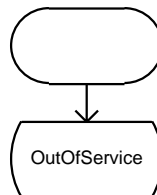
/* Data type definitions internal to the process */
/**/

```

```

/* Primitive Data Declarations */
/**/
/* Reject Cause */
DCL
V_RejCause IT_RejCause;

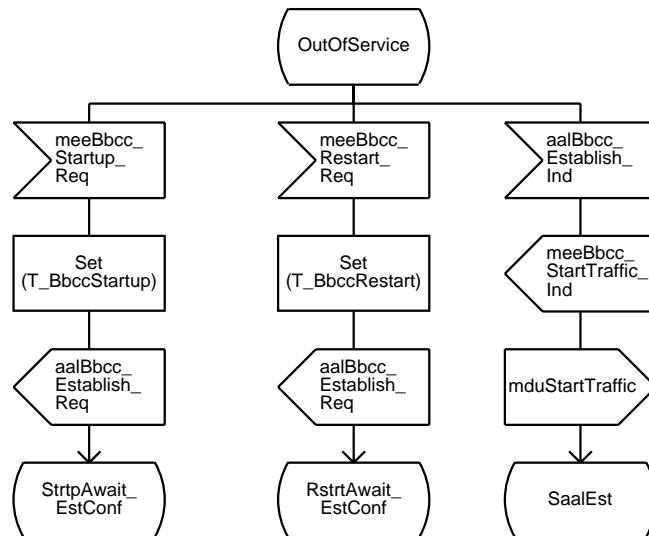
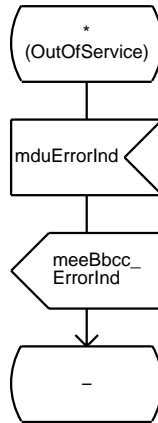
```



Process SN_CTR

2(8)

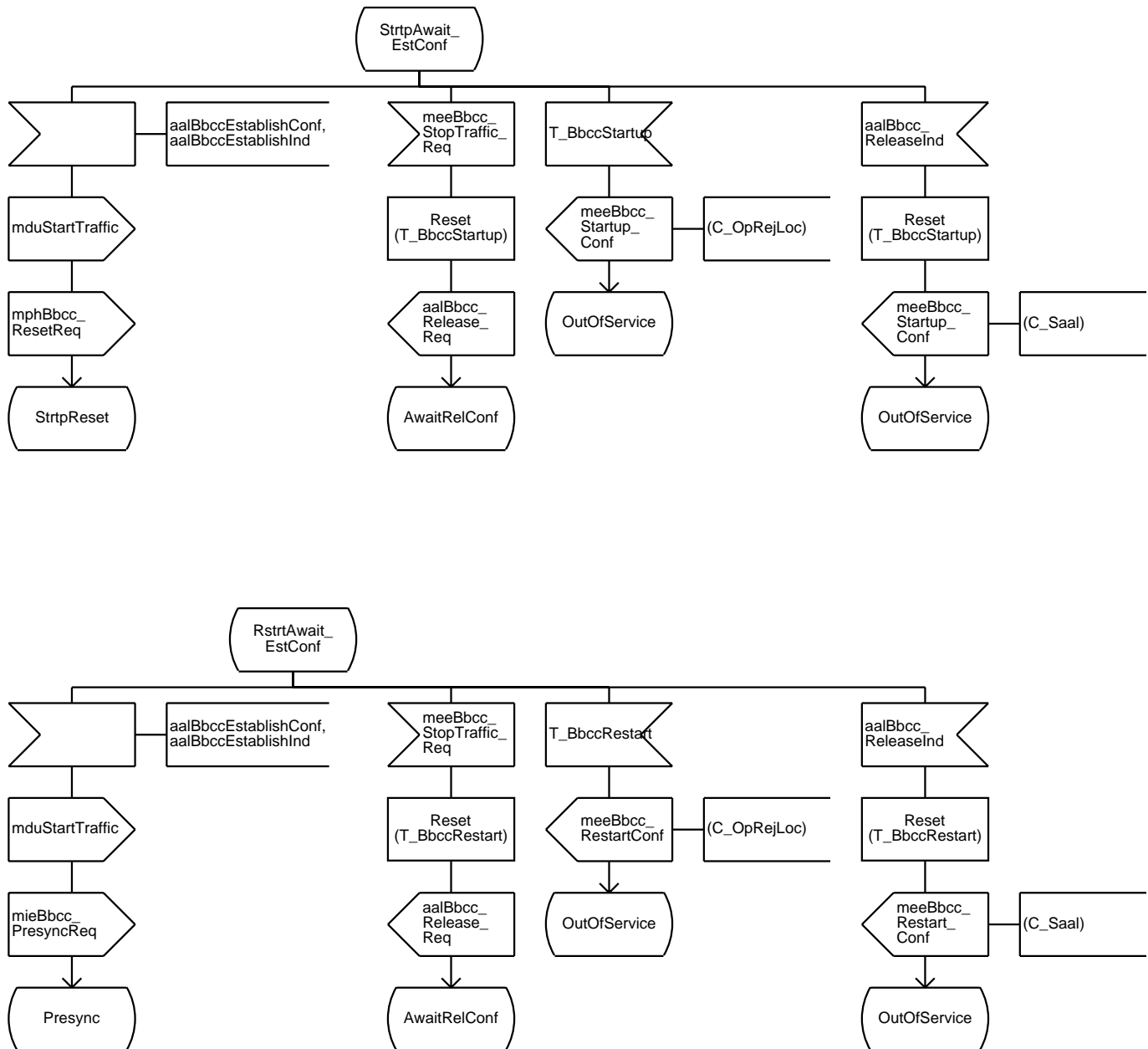
/* VB5.2 BCC management entity
Process CTR */



Process SN_CTR

3(8)

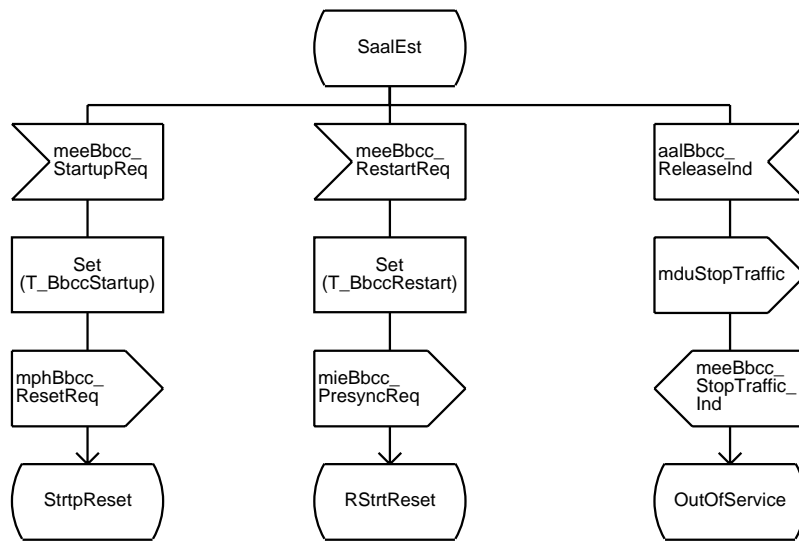
/* VB5.2 BCC management entity
Process CTR */



Process SN_CTR

4(8)

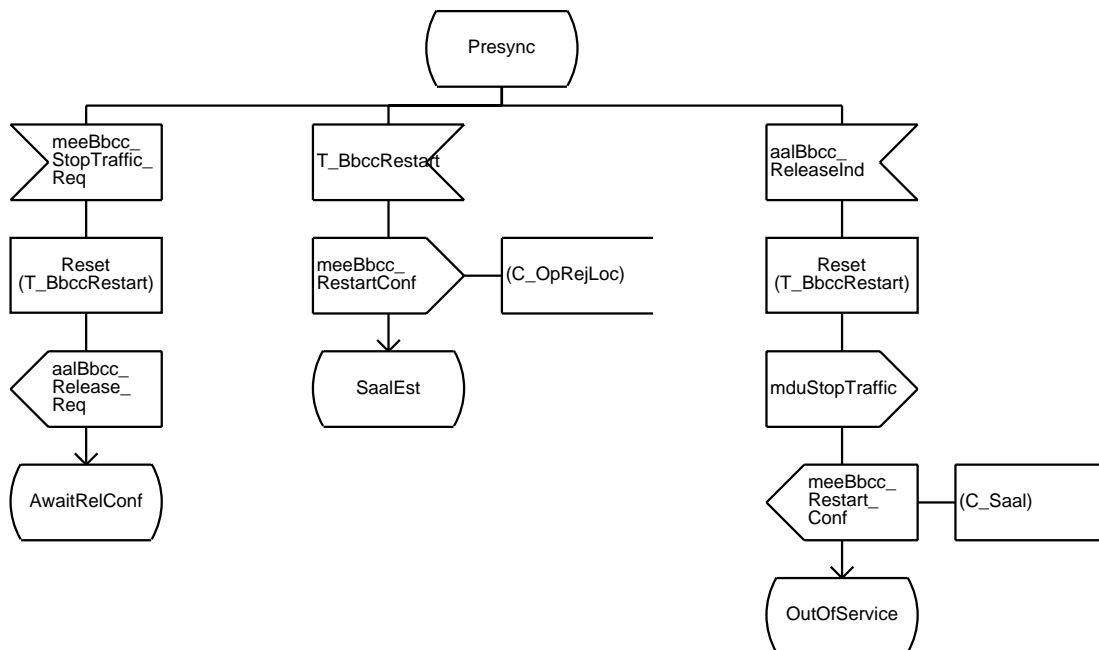
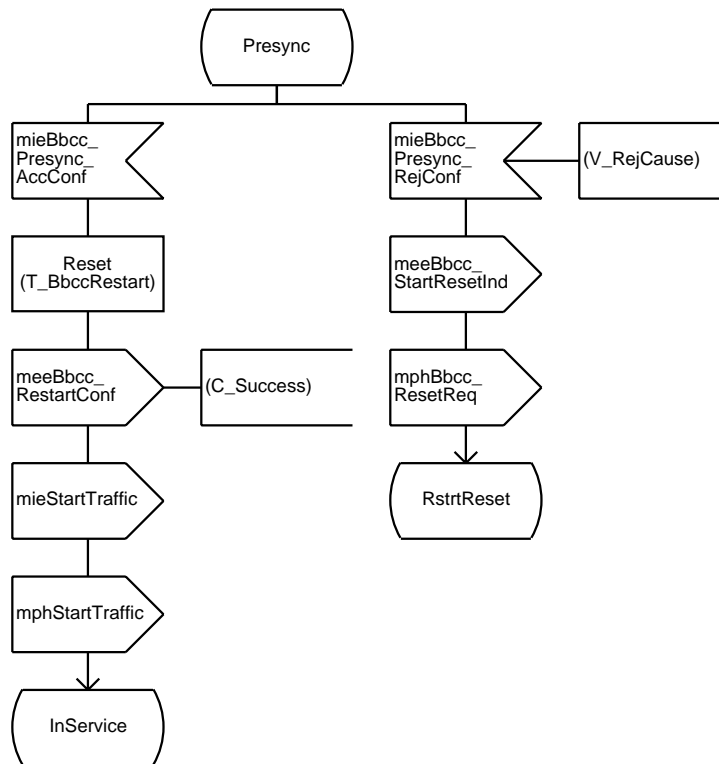
/* VB5.2 BCC management entity
Process CTR */



Process SN_CTR

5(8)

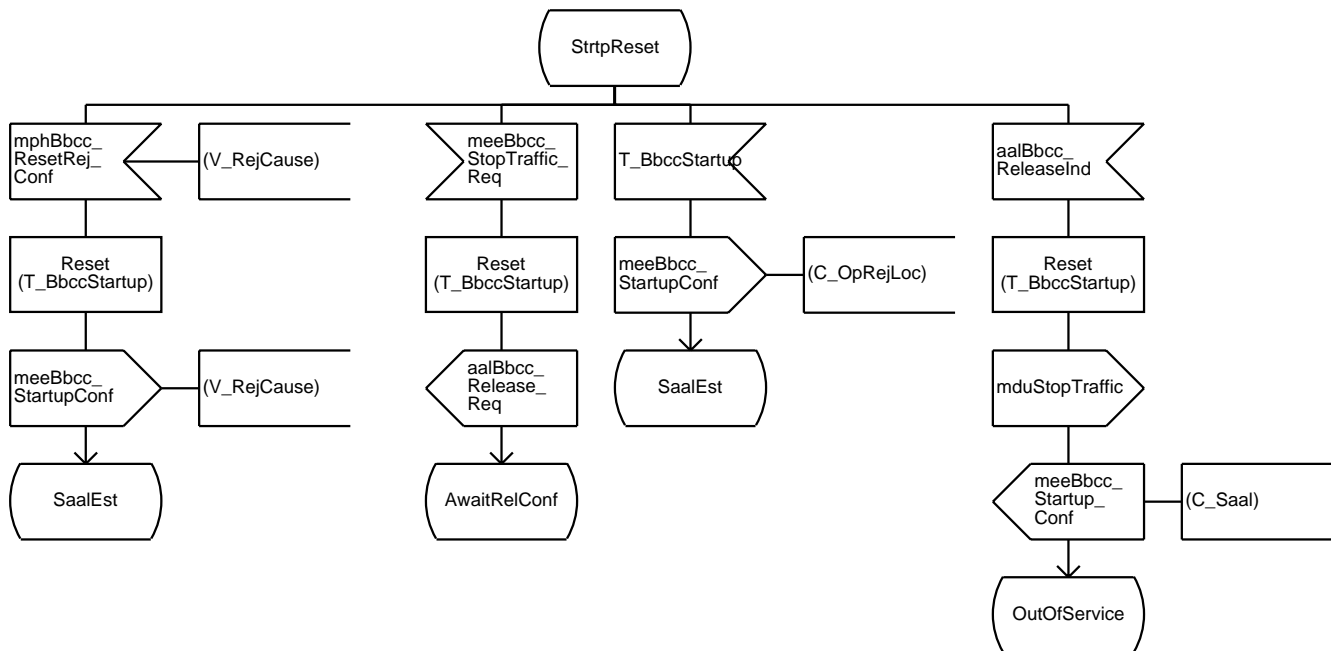
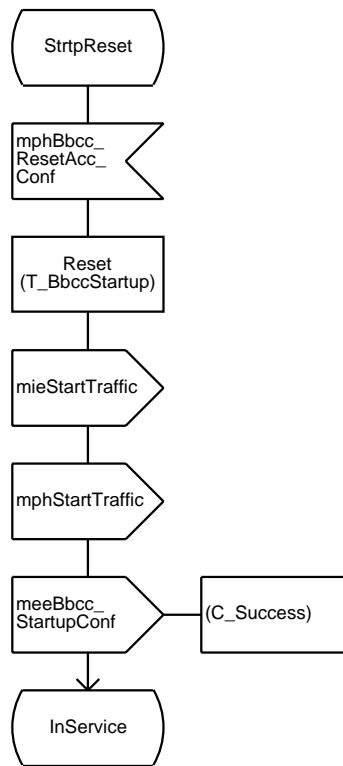
/* VB5.2 BCC management entity
Process CTR */



Process SN_CTR

6(8)

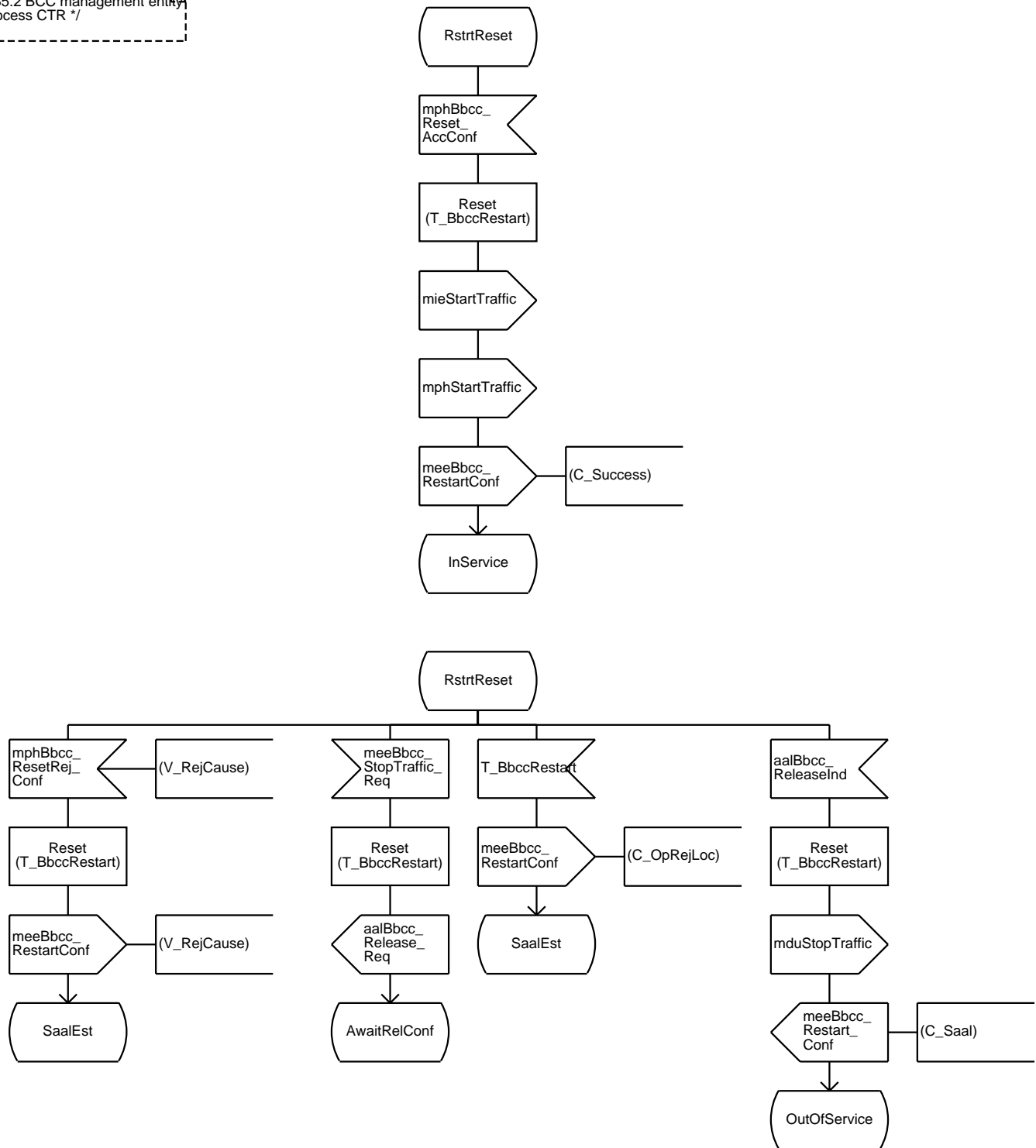
/* VB5.2 BCC management entity
Process CTR */



Process SN_CTR

7(8)

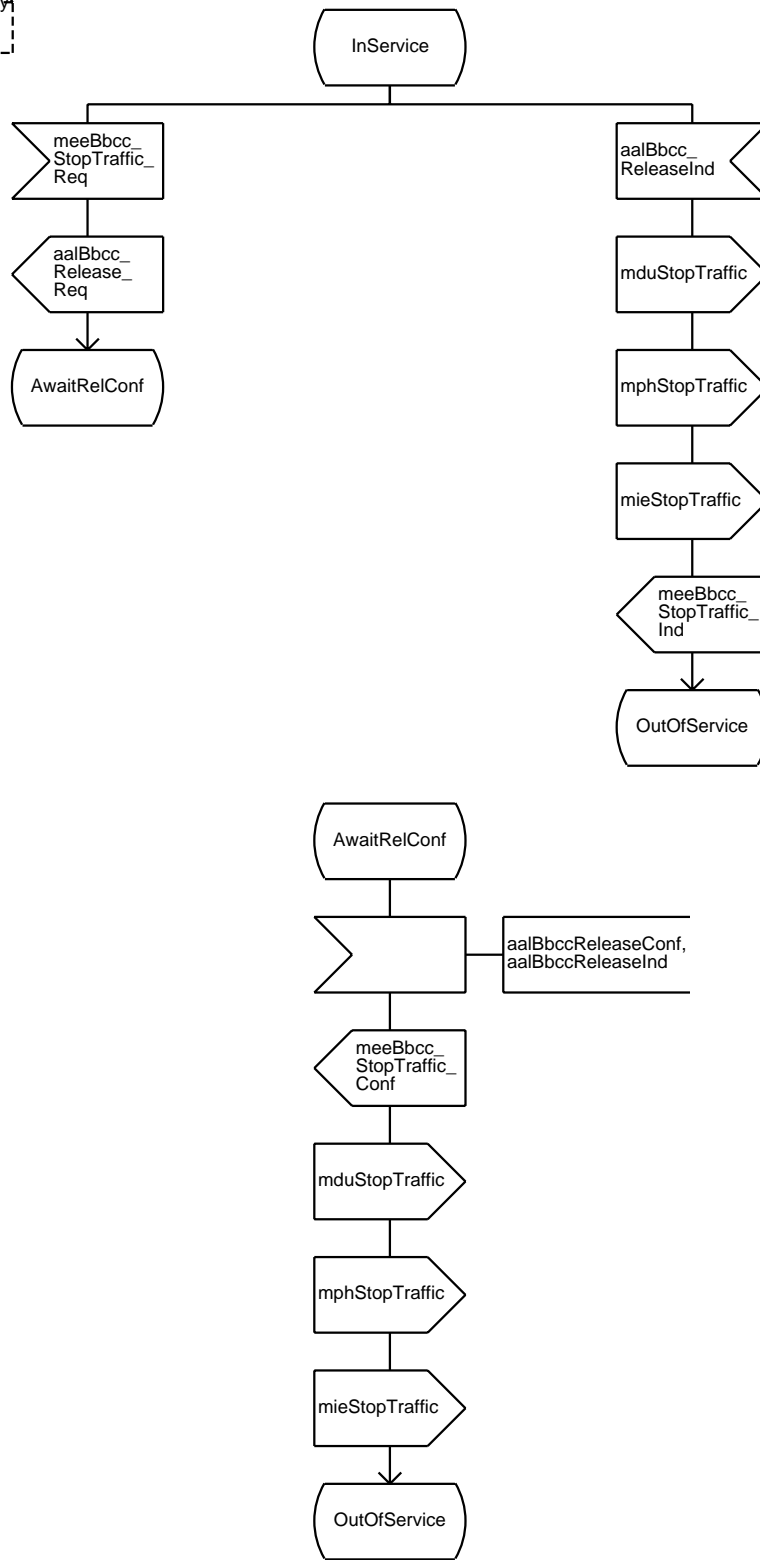
/* VB5.2 BCC management entity
Process CTR */



Process SN_CTR

8(8)

/* VB5.2 BCC management entity
Process CTR */



Process SN_PRESYNC

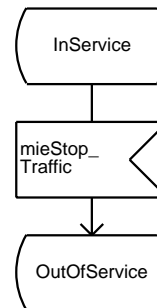
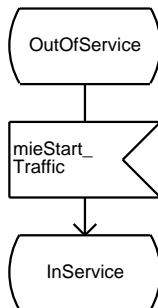
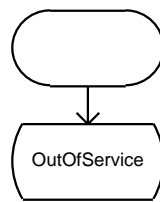
1(2)



```
/* Data type definitions internal to the process */  
/**/
```

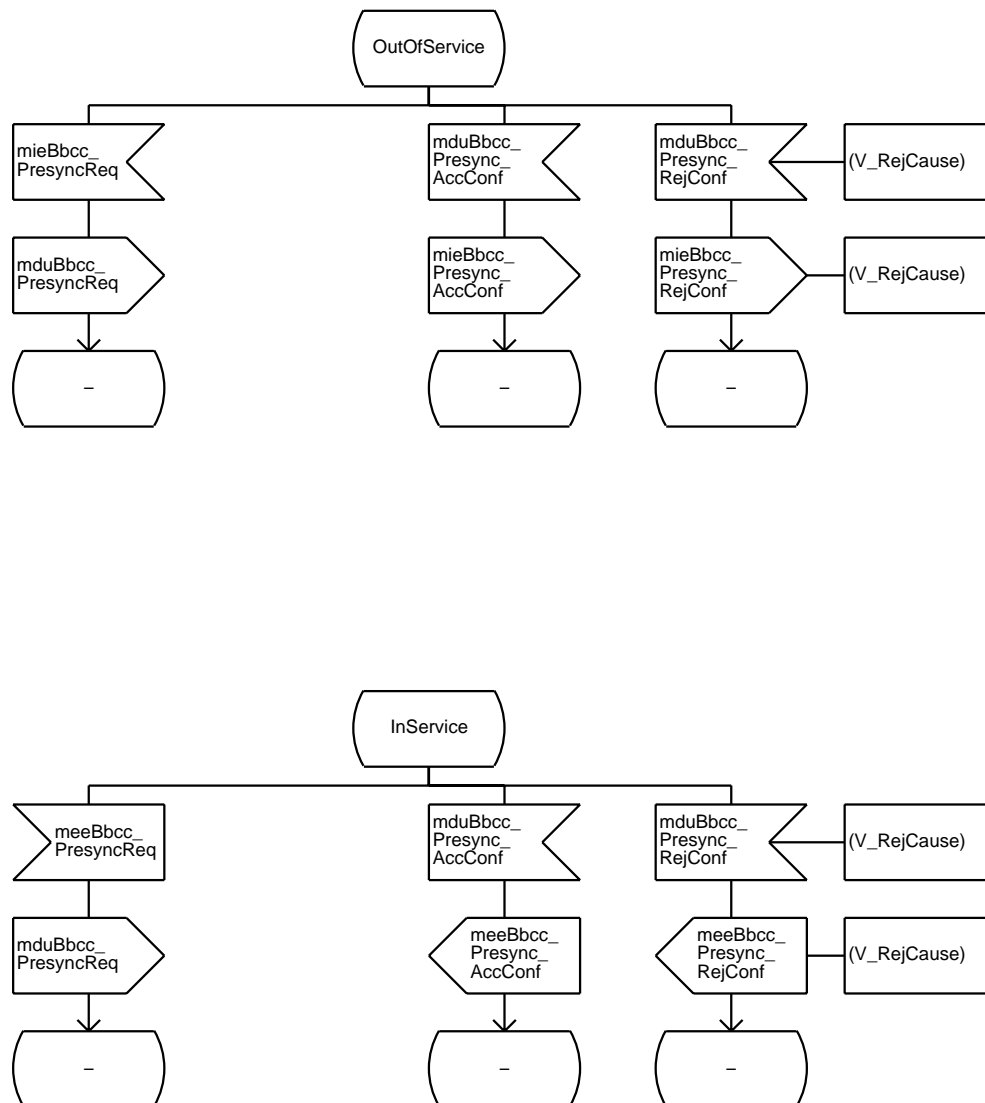
```
/* Primitive Data Declarations */  
/**/  
/* Reject Cause */  
DCL  
V_RejCause IT_RejCause;
```

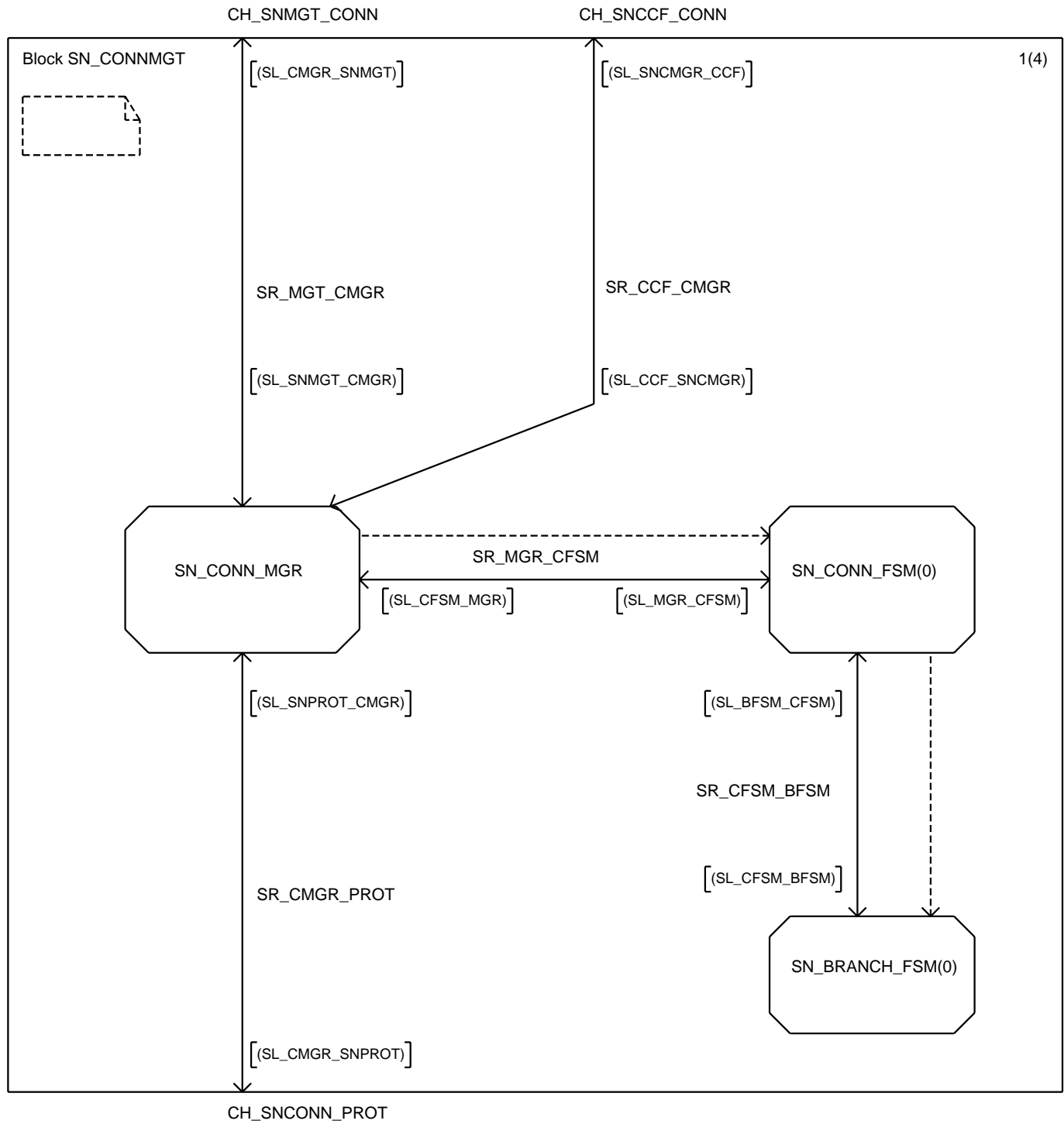
```
/* Process Data Declarations */  
/**/
```



Process SN_PRESYNC

2(2)





Block SN_CONNMGT

2(4)



/* Declaration of primitives internal
to block SN_CONNMGT, between SN_CONN_MGR
and SN_CONN_FSM*/

```
signal
mieAllocInd      (IT_BranchId),
mieAllocCompInd,
mieModifyInd,
mieModifyCompInd,
mieModifyAbortInd,
mieAddBranchInd  (IT_BranchId),
mieUpdateBranchInd (IT_BranchId),
mieDropBranchInd (IT_BranchId),
mieDeallocInd    (IT_BranchId);
```

```
signal
mieAllocAccReq      (IT_ConnRefNo, IT_BranchId),
mieAllocAccConf     (IT_ConnRefNo, IT_BranchId),
mieAllocRejReq      (IT_ConnRefNo, IT_BranchId, IT_RejCause),
mieAllocRejConf     (IT_ConnRefNo, IT_BranchId, IT_RejCause),
mieAllocCompAccReq  (IT_ConnRefNo),
mieAllocCompAccConf (IT_ConnRefNo),
mieAllocCompRejReq  (IT_ConnRefNo, IT_RejCause),
mieAllocCompRejConf (IT_ConnRefNo, IT_RejCause),
mieDeallocAccReq    (IT_ConnRefNo),
mieDeallocAccConf   (IT_ConnRefNo),
mieDeallocRejReq    (IT_ConnRefNo, IT_RejCause),
mieDeallocRejConf   (IT_ConnRefNo, IT_RejCause),
mieModifyAccReq     (IT_ConnRefNo),
mieModifyAccConf    (IT_ConnRefNo),
mieModifyRejReq     (IT_ConnRefNo, IT_RejCause),
mieModifyRejConf    (IT_ConnRefNo, IT_RejCause),
mieModifyCompAccReq (IT_ConnRefNo),
mieModifyCompAccConf (IT_ConnRefNo),
mieModifyCompRejReq (IT_ConnRefNo, IT_RejCause),
mieModifyCompRejConf (IT_ConnRefNo, IT_RejCause),
mieModifyAbortAccReq (IT_ConnRefNo),
mieModifyAbortAccConf (IT_ConnRefNo),
mieModifyAbortRejReq (IT_ConnRefNo, IT_RejCause),
mieModifyAbortRejConf (IT_ConnRefNo, IT_RejCause),
mieAddBranchAccReq  (IT_ConnRefNo, IT_BranchId),
mieAddBranchAccConf (IT_ConnRefNo, IT_BranchId),
mieAddBranchRejReq  (IT_ConnRefNo, IT_BranchId, IT_RejCause),
mieAddBranchRejConf (IT_ConnRefNo, IT_BranchId, IT_RejCause),
mieUpdateBranchAccReq (IT_ConnRefNo, IT_BranchId),
mieUpdateBranchAccConf (IT_ConnRefNo, IT_BranchId),
mieUpdateBranchRejReq (IT_ConnRefNo, IT_BranchId, IT_RejCause),
mieUpdateBranchRejConf (IT_ConnRefNo, IT_BranchId, IT_RejCause),
mieDropBranchAccReq (IT_ConnRefNo, IT_BranchId),
mieDropBranchAccConf (IT_ConnRefNo, IT_BranchId),
mieDropBranchRejReq (IT_ConnRefNo, IT_BranchId, IT_RejCause),
mieDropBranchRejConf (IT_ConnRefNo, IT_BranchId, IT_RejCause);
```

Block SN_CONNMGT

3(4)



/* Declaration of primitives internal
to block SN_CONNMGT, between SN_CONN_FSM
and SN_BRANCH_FSM */

```
signal
mieUpdateBranchFSMInd ,
mieDropBranchFSMInd ,
mieAddBranchFSMAccReq (IT_BranchId),
mieFirstAddBranchFSMAccInd(IT_BranchId),
mieAddBranchFSMAccConf (IT_BranchId),
mieAddBranchFSMRejReq (IT_BranchId, IT_RejCause),
mieAddBranchFSMRejConf (IT_BranchId, IT_RejCause),
mieUpdateBranchFSMAccReq (IT_BranchId),
mieUpdateBranchFSMAccConf (IT_BranchId),
mieUpdateBranchFSMRejReq (IT_BranchId, IT_RejCause),
mieUpdateBranchFSMRejConf (IT_BranchId, IT_RejCause),
mieDropBranchFSMAccReq (IT_BranchId),
mieDropBranchFSMAccConf (IT_BranchId),
mieDropBranchFSMRejReq (IT_BranchId, IT_RejCause),
mieDropBranchFSMRejConf (IT_BranchId, IT_RejCause),
mieDelBranchFSMInd ;
```

Block SN_CONNMGT

4(4)



```
/* Declaration of signallists
   internal to block SN_CONNMGT */
```

```
signallist SL_MGR_CFSM =
mieAllocInd,
mieAllocCompInd,
mieModifyInd,
mieModifyCompInd,
mieModifyAbortInd,
mieAddBranchInd,
mieUpdateBranchInd,
mieDropBranchInd,
mieDeallocInd,
mieAllocAccReq,
mieAllocRejReq,
mieAllocCompAccReq,
mieAllocCompRejReq,
mieDeallocAccReq,
mieDeallocRejReq,
mieModifyAccReq,
mieModifyRejReq,
mieModifyCompAccReq,
mieModifyCompRejReq,
mieModifyAbortAccReq,
mieModifyAbortRejReq,
mieAddBranchAccReq,
mieAddBranchRejReq,
mieUpdateBranchAccReq,
mieUpdateBranchRejReq,
mieDropBranchAccReq,
mieDropBranchRejReq,
mieResetInd;
```

```
signallist SL_CFSM_MGR =
mieAllocAccConf,
mieAllocRejConf,
mieAllocCompAccConf,
mieAllocCompRejConf,
mieDeallocAccConf,
mieDeallocRejConf,
mieModifyAccConf,
mieModifyRejConf,
mieModifyCompAccConf,
mieModifyCompRejConf,
mieModifyAbortAccConf,
mieModifyAbortRejConf,
mieAddBranchAccConf,
mieAddBranchRejConf,
mieUpdateBranchAccConf,
mieUpdateBranchRejConf,
mieDropBranchAccConf,
mieDropBranchRejConf;
```

```
signallist SL_CFSM_BFSM =
mieUpdateBranchFSMInd,
mieDropBranchFSMInd,
mieAddBranchFSMAccReq,
mieFirstAddBranchFSMAccInd,
mieAddBranchFSMRejReq,
mieUpdateBranchFSMAccReq,
mieUpdateBranchFSMRejReq,
mieDropBranchFSMAccReq,
mieDropBranchFSMRejReq,
mieDelBranchFSMInd;
```

```
signallist SL_BFSM_CFSM =
mieAddBranchFSMAccConf,
mieAddBranchFSMRejConf,
mieUpdateBranchFSMAccConf,
mieUpdateBranchFSMRejConf,
mieDropBranchFSMAccConf,
mieDropBranchFSMRejConf;
```

Process SN_CONN_MGR

1(13)



```

/* Data type definitions internal to the process */
/**/
/* This data type is used for translation of the connection reference number to the process identifier (PID)
of the connection FSM */
NEWTYPE AT_ConnPIdList ARRAY
(IT_ConnRefNo,PID)
ENDNEWTYPE AT_ConnPIdList;

```

```

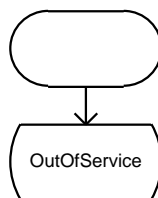
/* Primitive Data Declarations */
/**/
/* List of connections */
DCL
A_ConnList AT_ConnList;
/**/
/* Connection reference number */
DCL
V_ConnRefNo IT_ConnRefNo;
/**/
/* Branch identifier */
DCL
V_BranchId IT_BranchId;
/**/
/* Reject Cause */
DCL
V_RejCause IT_RejCause;

```

```

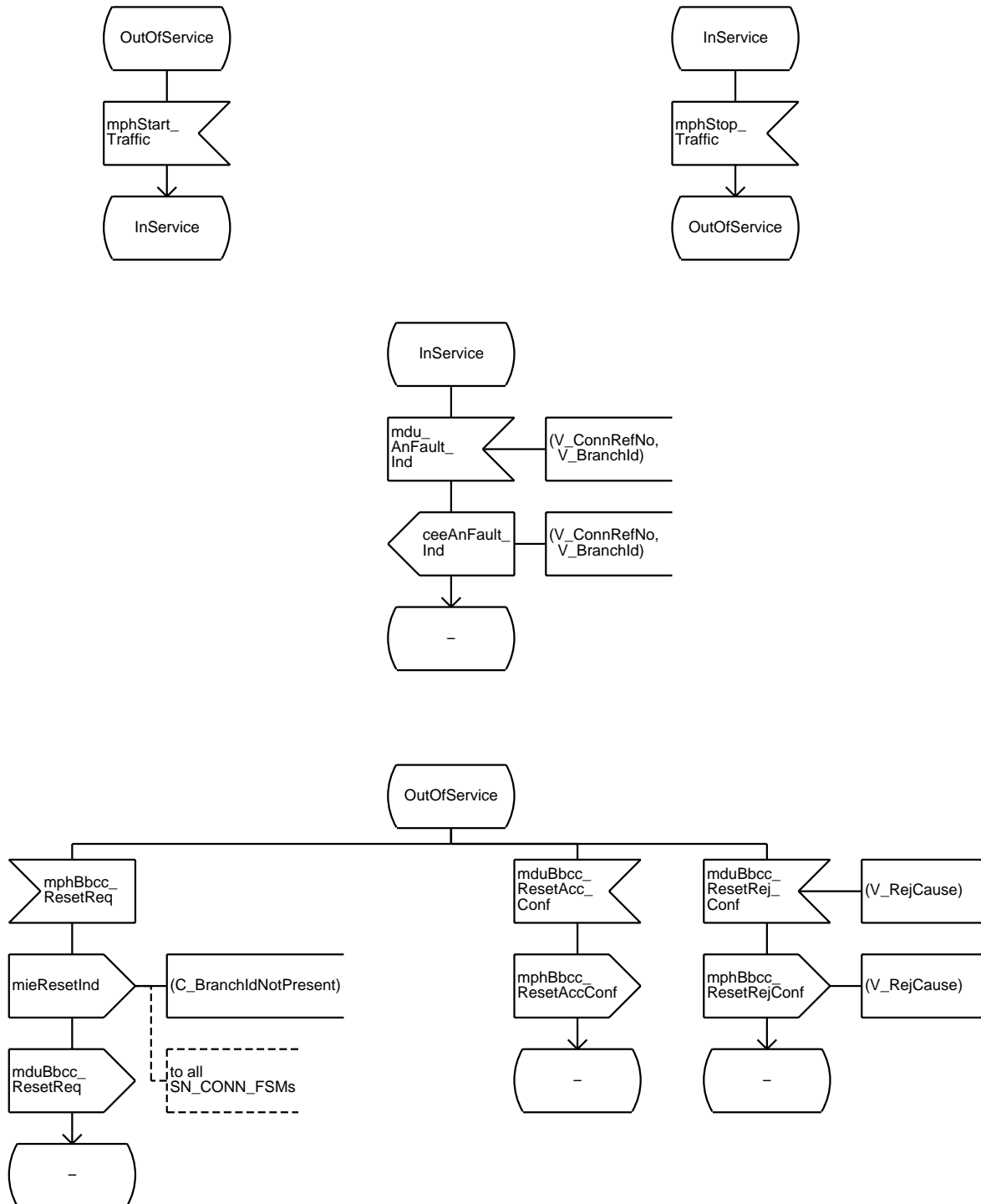
/* Process Data Declarations */
/**/
/* List of connection FSMs */
DCL
A_ConnPIdList AT_ConnPIdList;
/**/
/* Index of A_ConnList */
DCL
V_Element INTEGER;

```



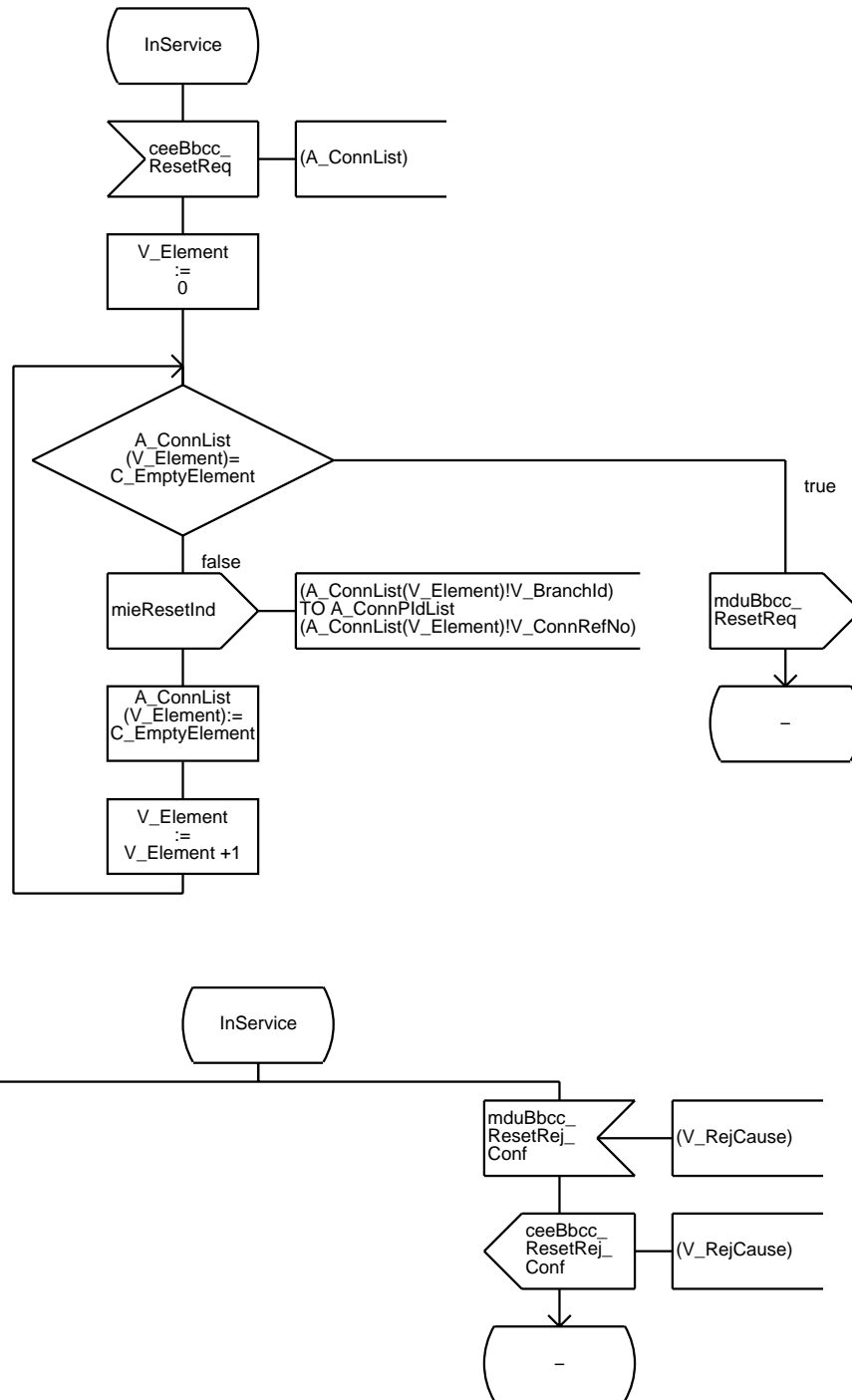
Process SN_CONN_MGR

2(13)



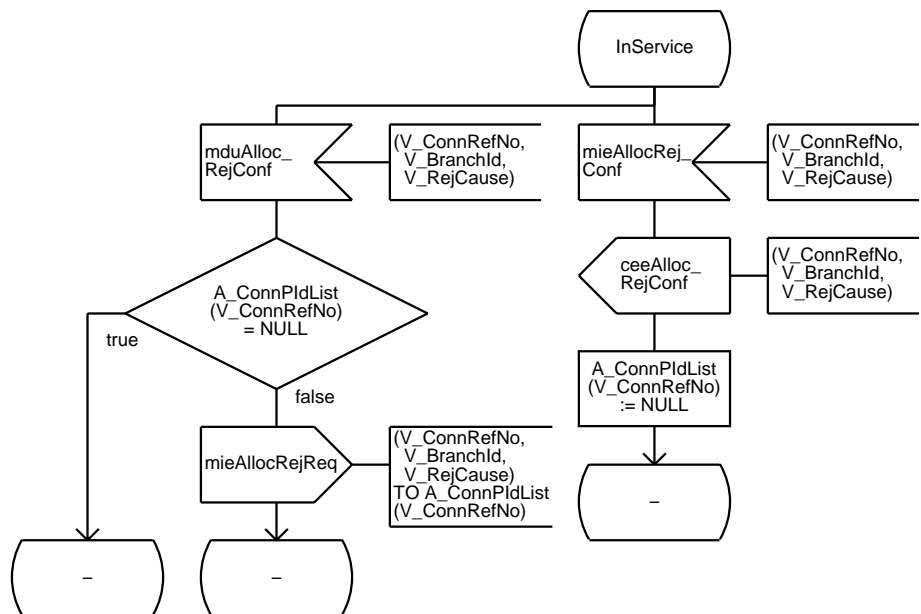
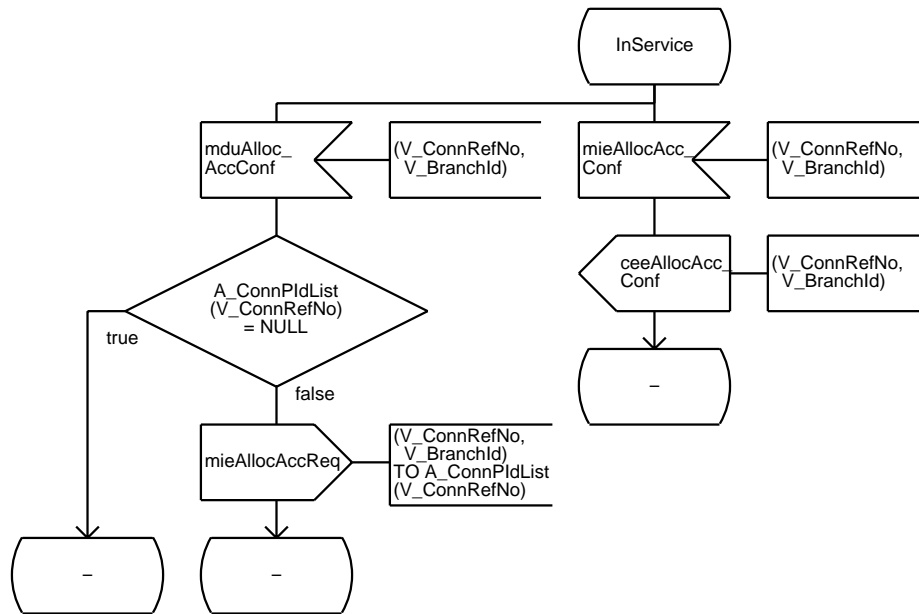
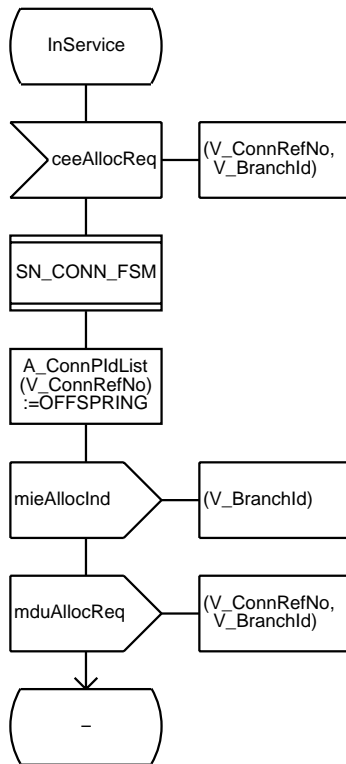
Process SN_CONN_MGR

3(13)



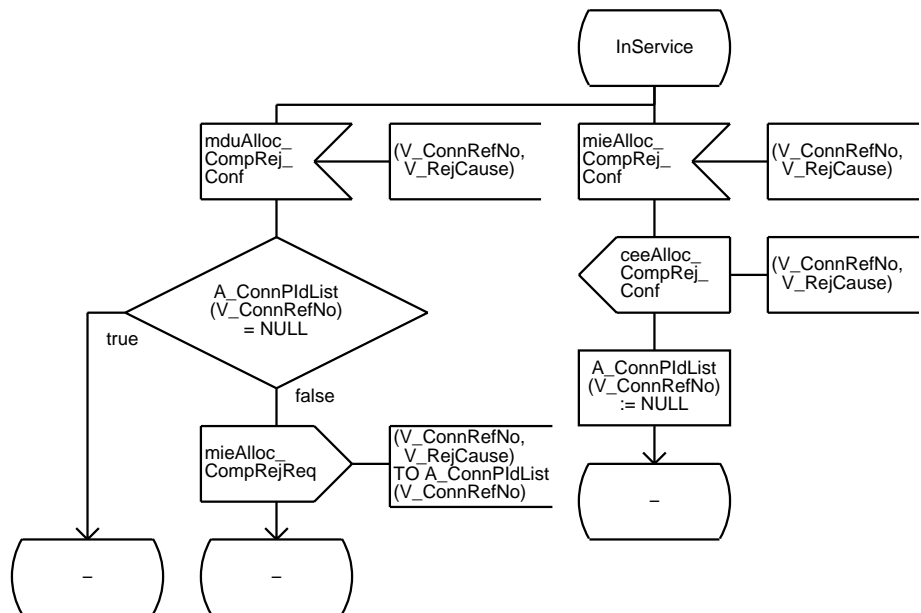
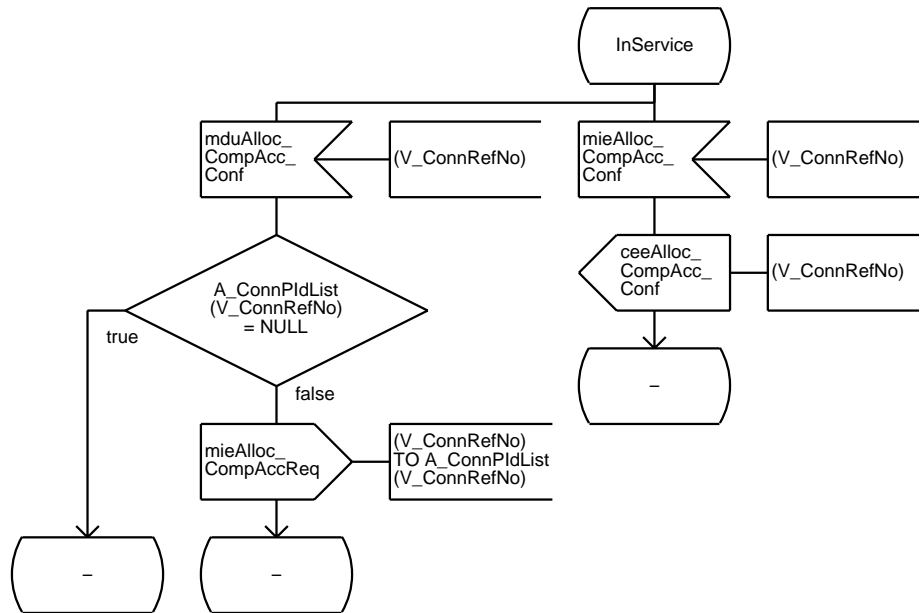
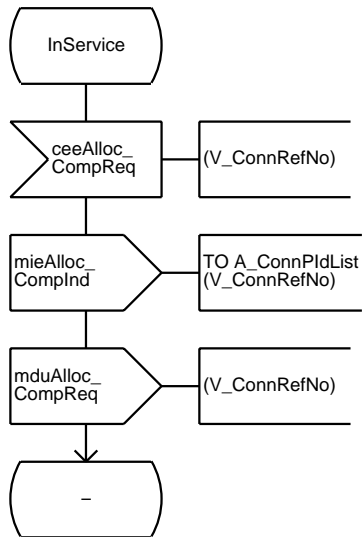
Process SN_CONN_MGR

4(13)



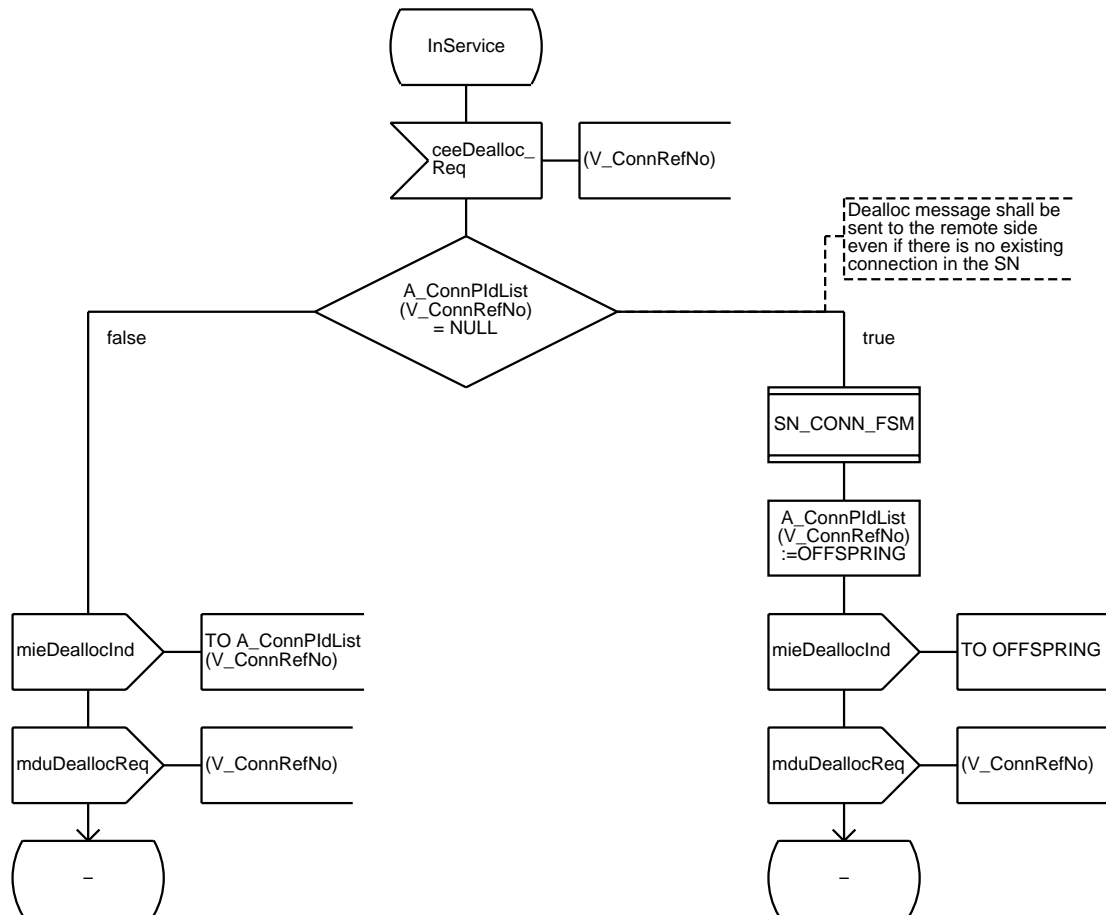
Process SN_CONN_MGR

5(13)



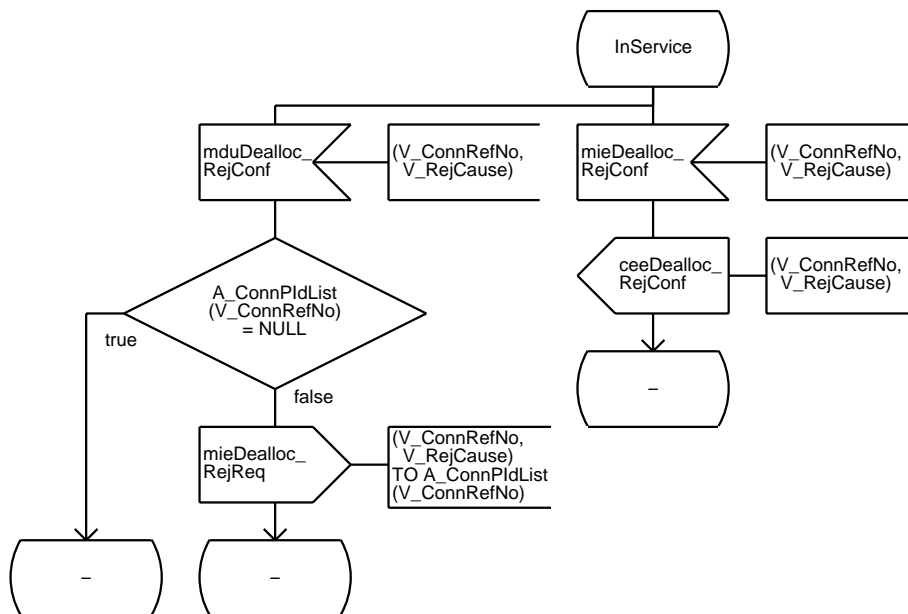
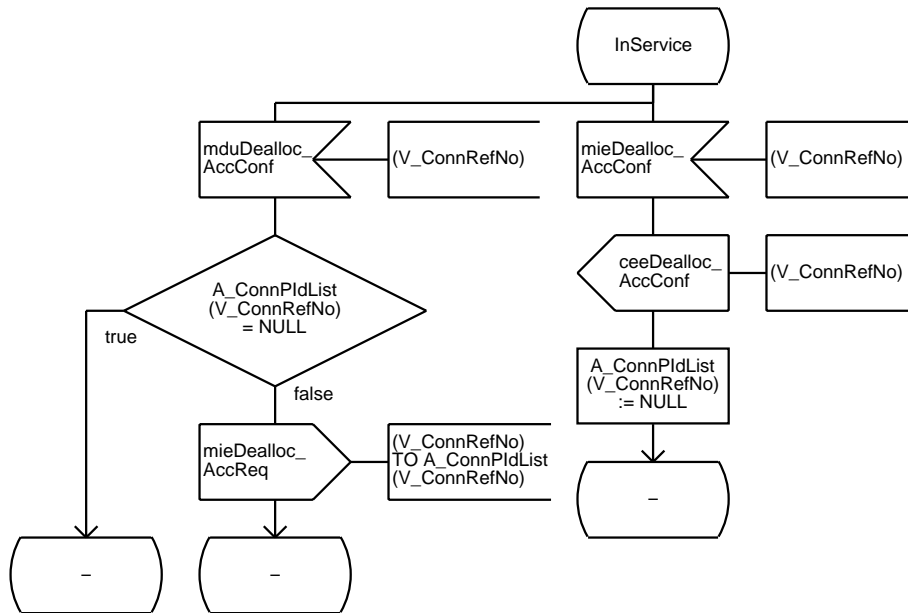
Process SN_CONN_MGR

6(13)



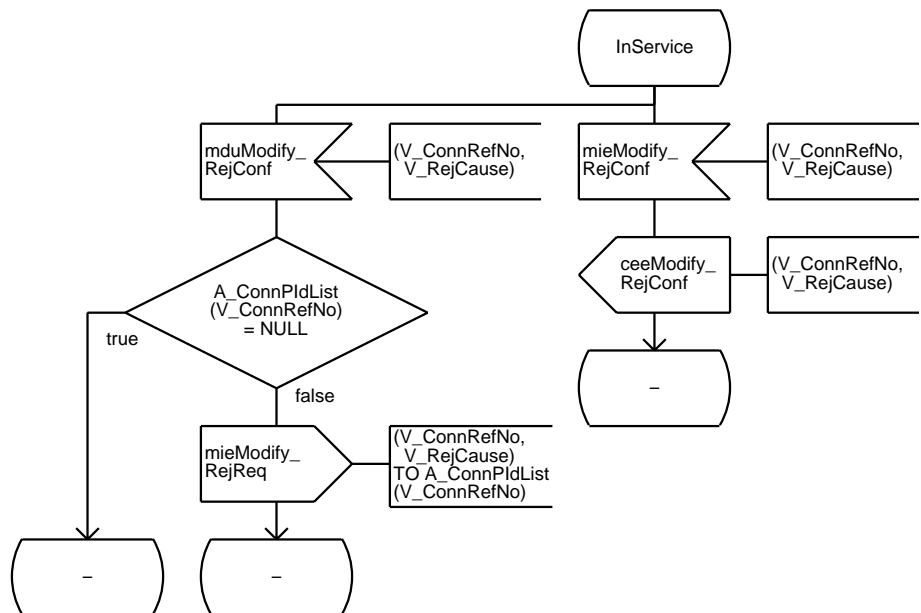
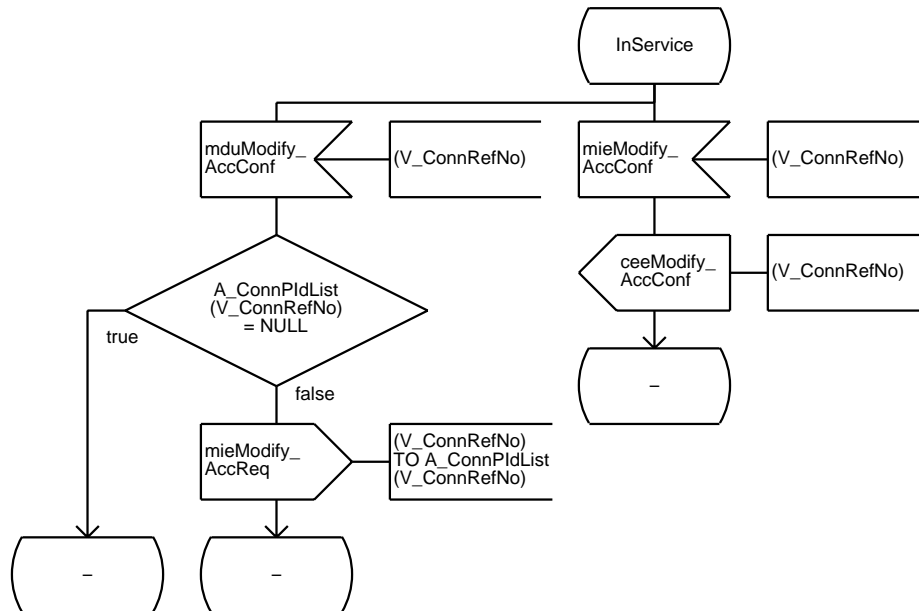
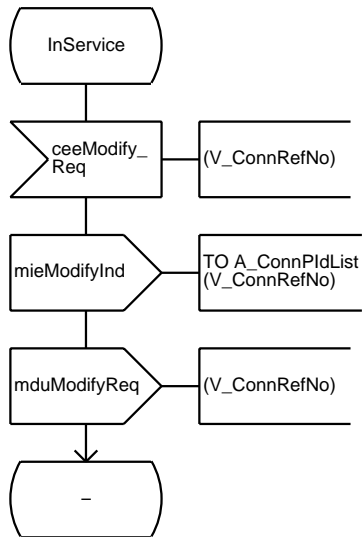
Process SN_CONN_MGR

7(13)



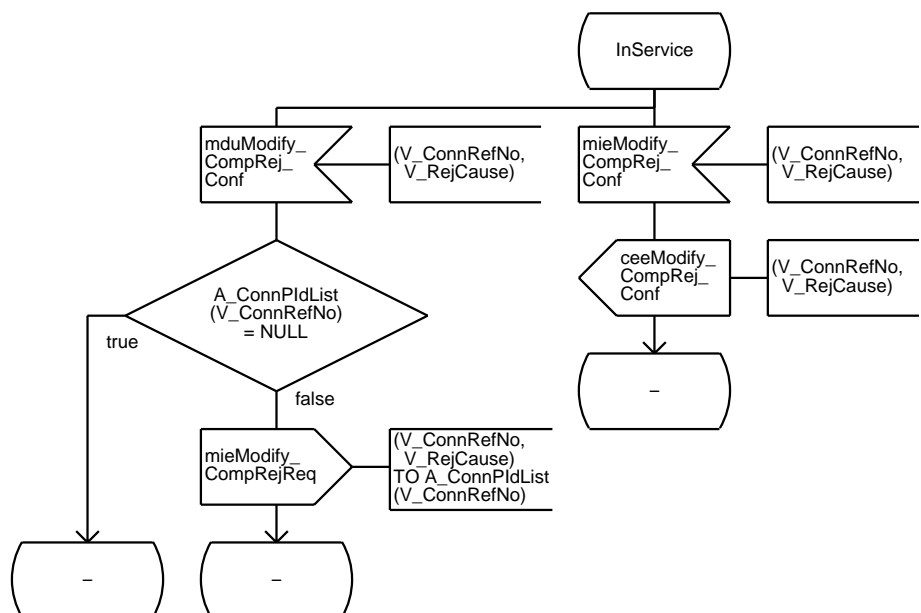
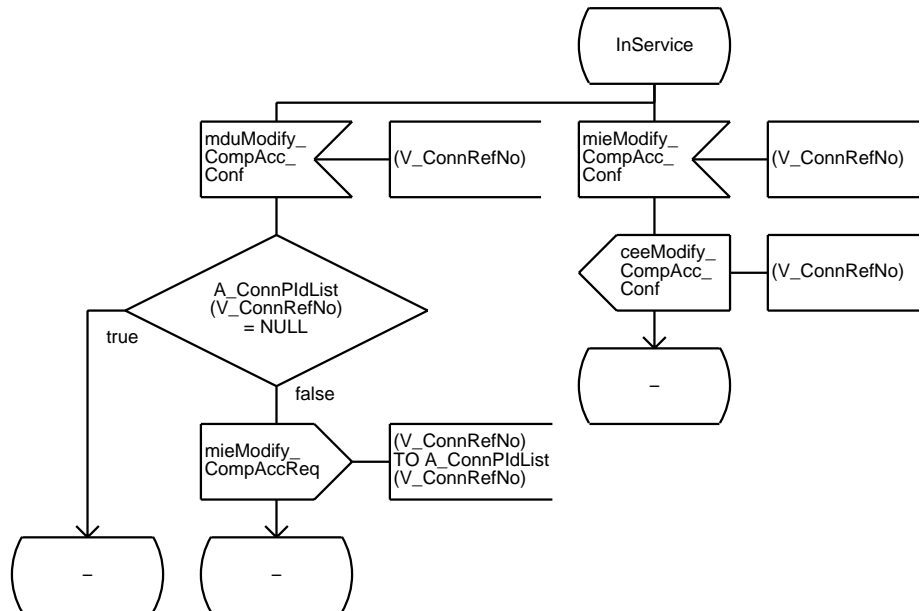
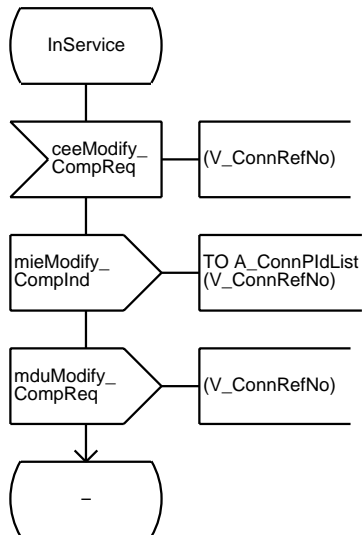
Process SN_CONN_MGR

8(13)



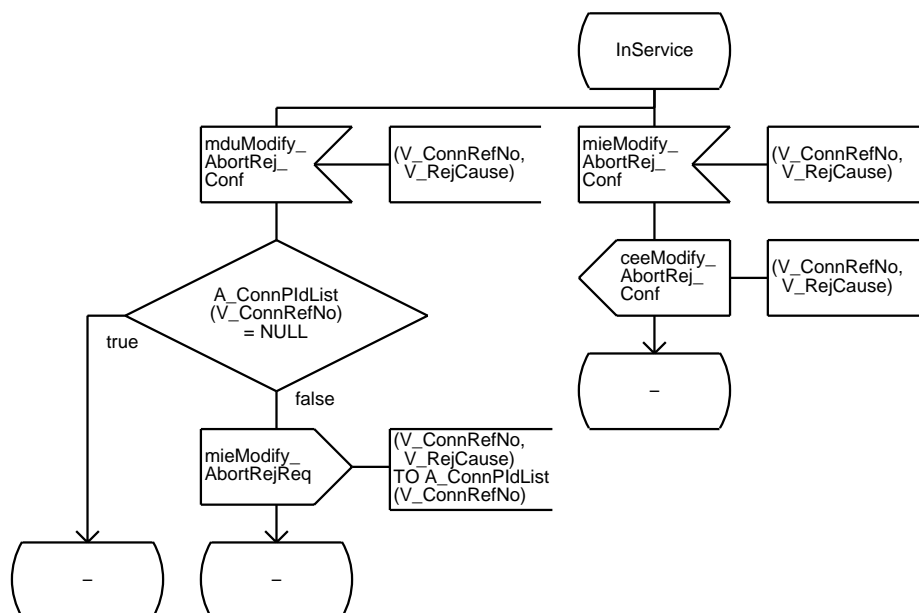
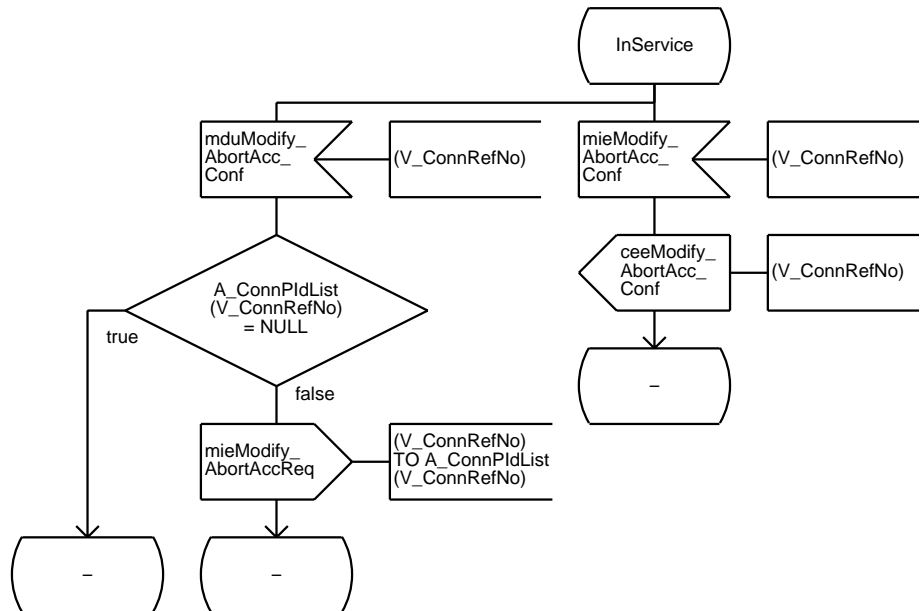
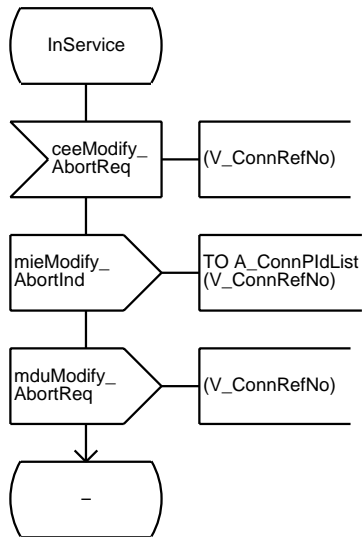
Process SN_CONN_MGR

9(13)



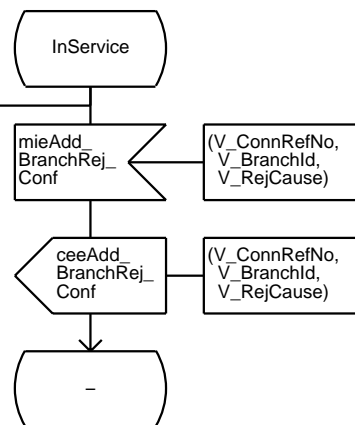
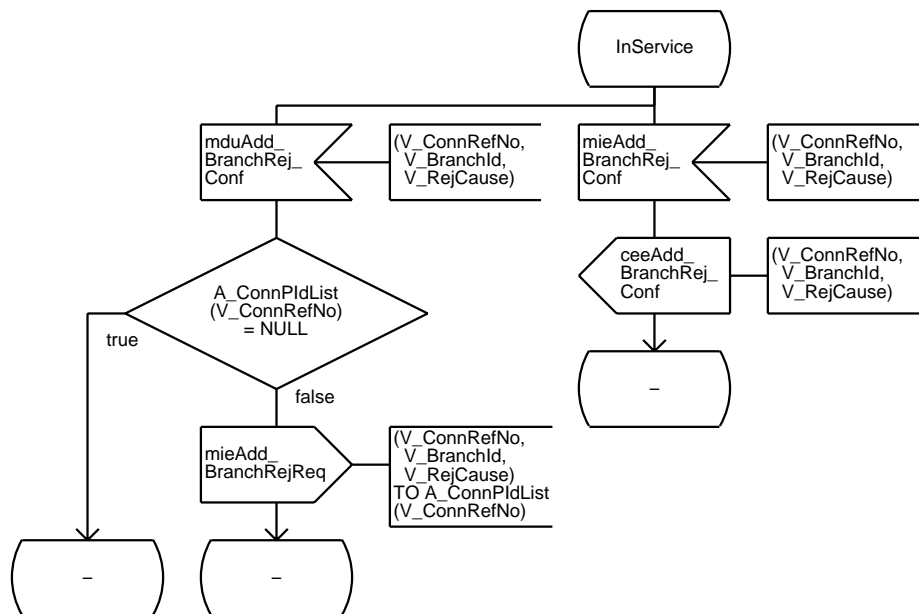
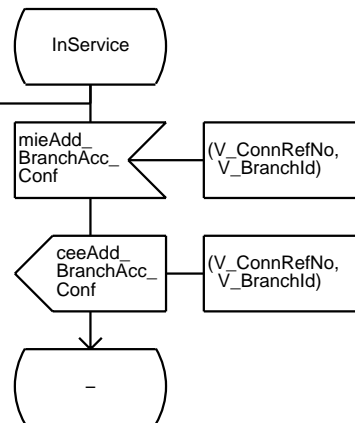
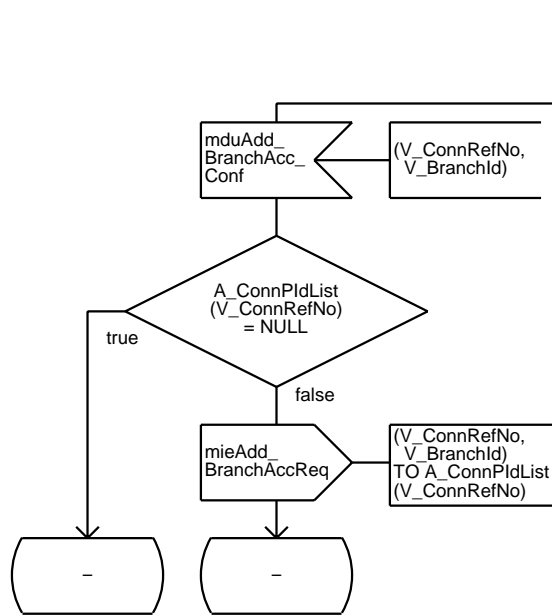
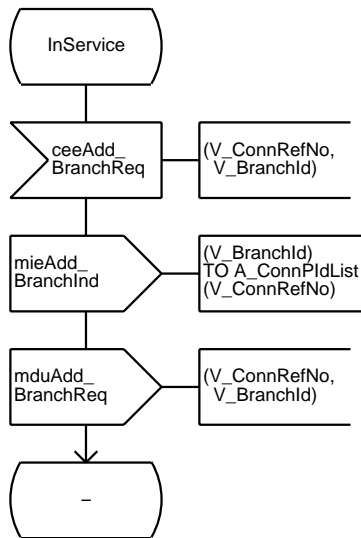
Process SN_CONN_MGR

10(13)



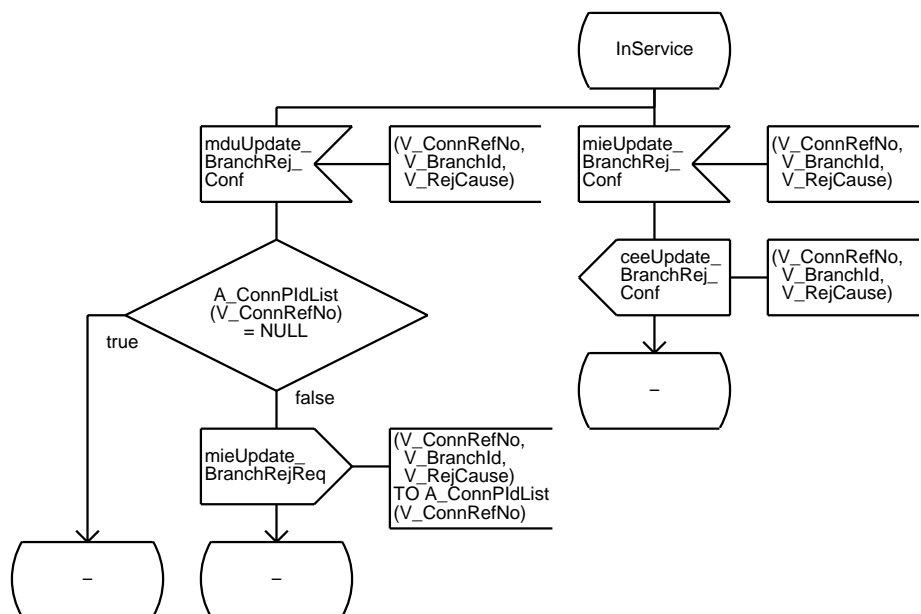
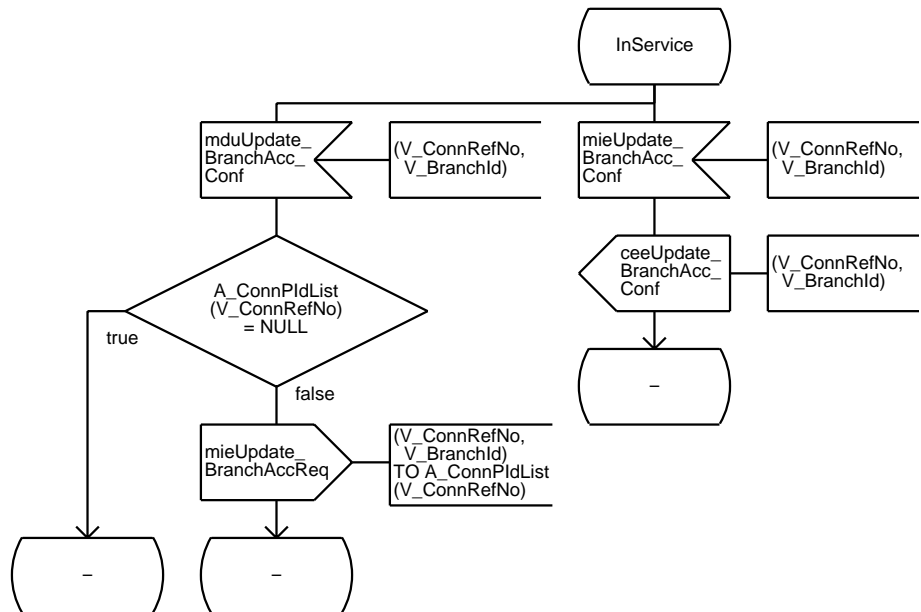
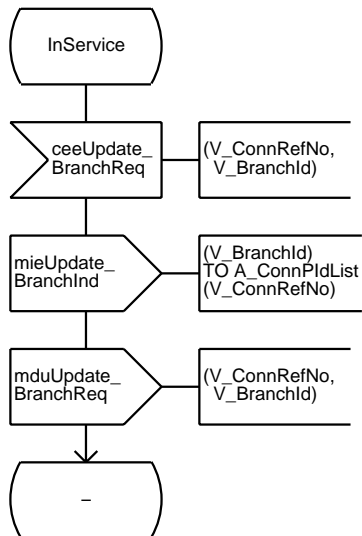
Process SN_CONN_MGR

11(13)



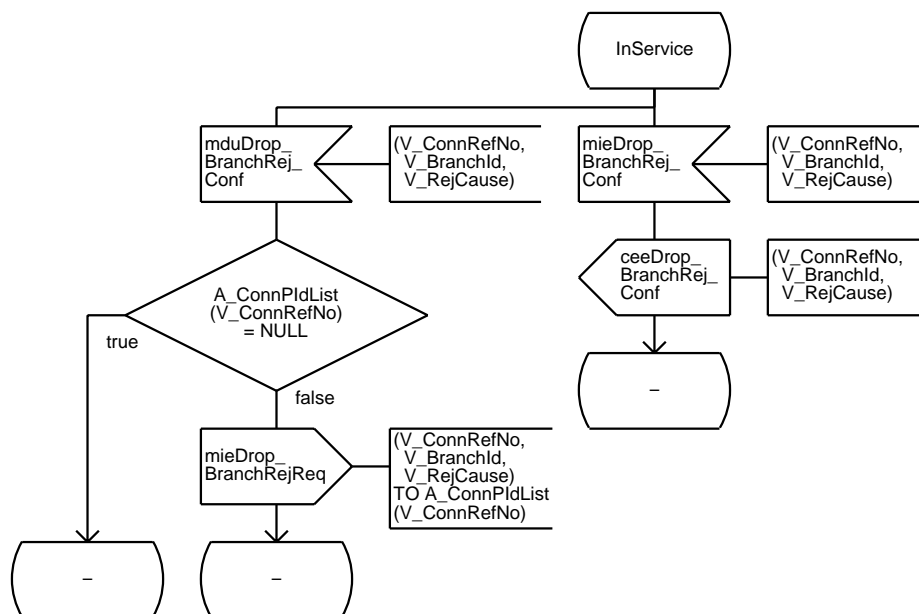
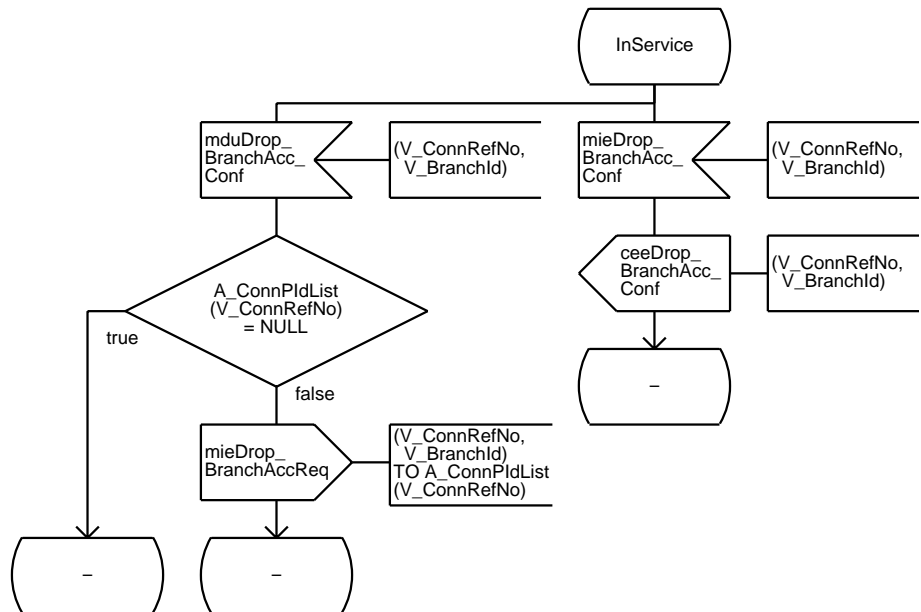
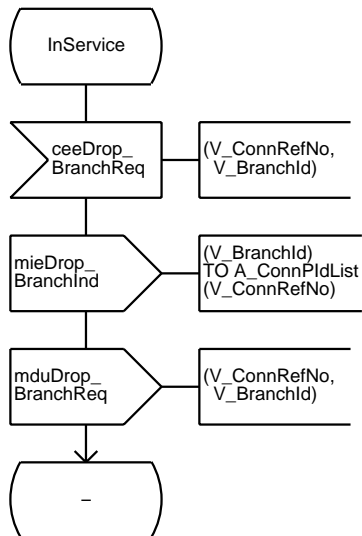
Process SN_CONN_MGR

12(13)



Process SN_CONN_MGR

13(13)



Process SN_CONN_FSM

1(10)



```

/* State descriptions
Init:      The process is waiting for the first message from the manager
           indicating a ptp or ptm connection
InService: The process is not waiting for a confirmation of a message
AllocEval: An alloc is sent but not confirmed yet
AllocCompEval: An alloc complete is sent but not confirmed yet
ModifyEval: An modify is sent but not confirmed yet
ModifyCompEval: An modify complete is sent but not confirmed yet
ModifyAbortEval: An modify abort is sent but not confirmed yet
DeallocEval: An dealloc is sent but not confirmed yet
*/

```

```

/* Data type definitions internal to the process */
/**
/* This data type is used for translation of the branch identifier to the process identifier (PID)
of the branch FSM */
NEWTTYPE AT_BranchList ARRAY
(IT_BranchId,PID)
ENDNEWTTYPE AT_BranchList;

```

```

/*Primitive Data Declarations*/
/**
/* Connection identifier */
DCL
V_ConnRefNo IT_ConnRefNo;
/**
/* Branch identifier */
DCL
V_BranchId IT_BranchId;
/**
/* Reject Cause */
DCL
V_RejCause IT_RejCause;

```

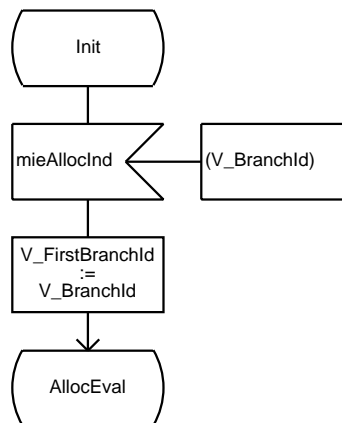
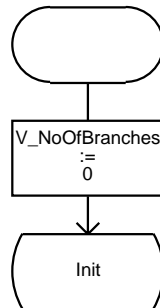
```

/* Process Data Declarations */
/**
/* List of connection FSMs */
DCL
A_BranchList AT_BranchList;
/**
/* Number of branches */
DCL
V_NoOfBranches INTEGER;
/**
/* First branch */
DCL
V_FirstBranchId IT_BranchId;

```

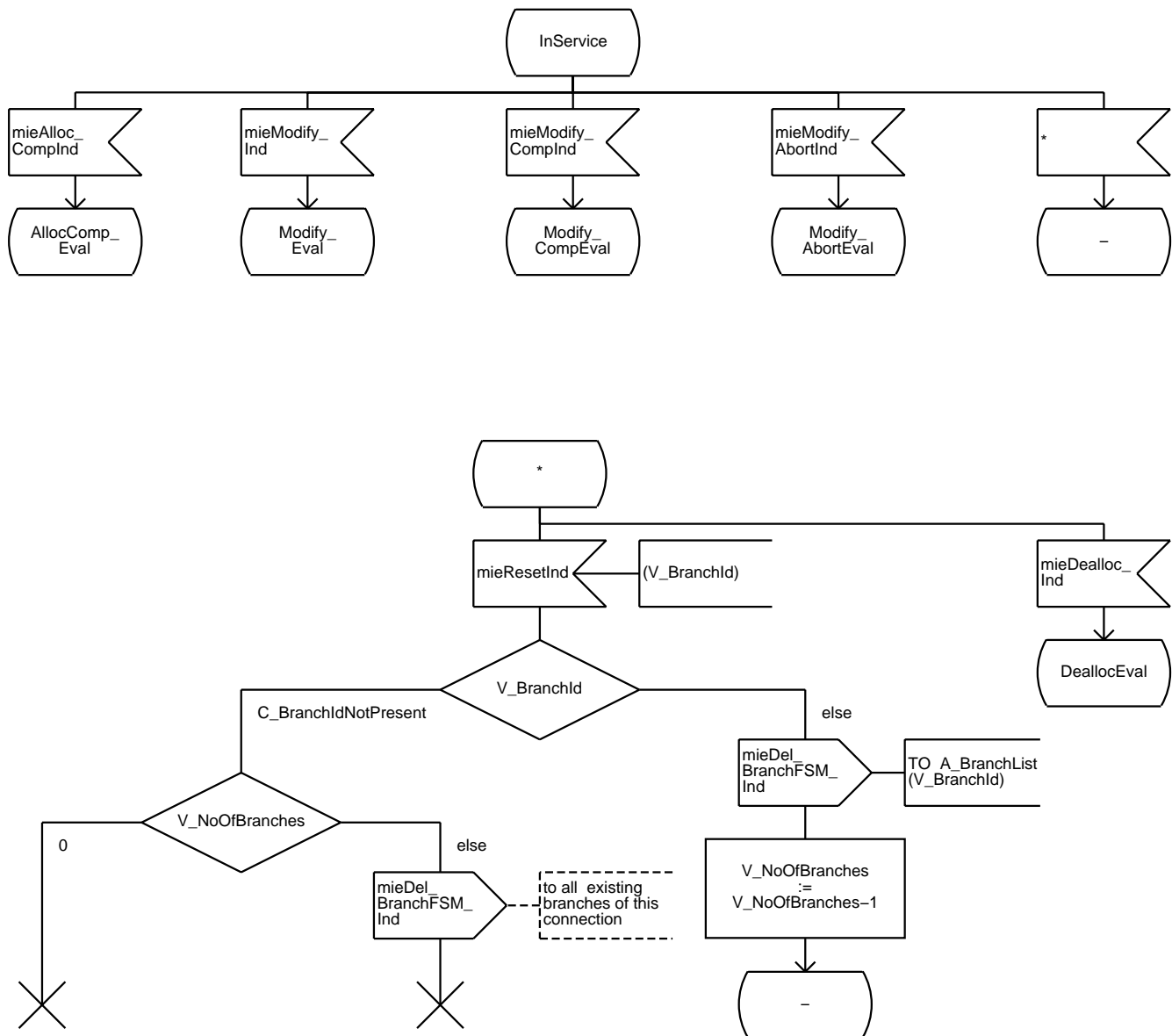
Process SN_CONN_FSM

2(10)



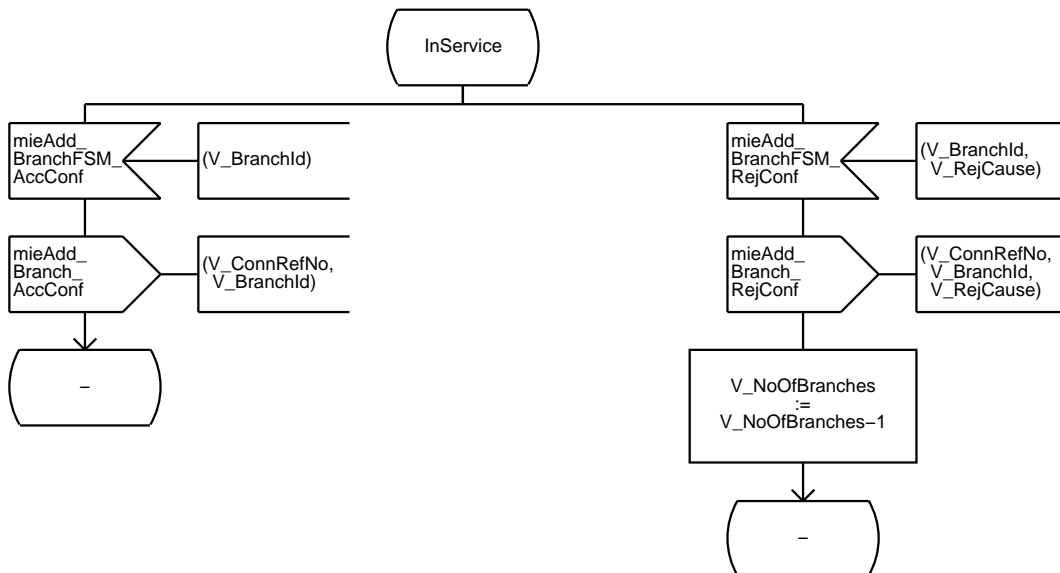
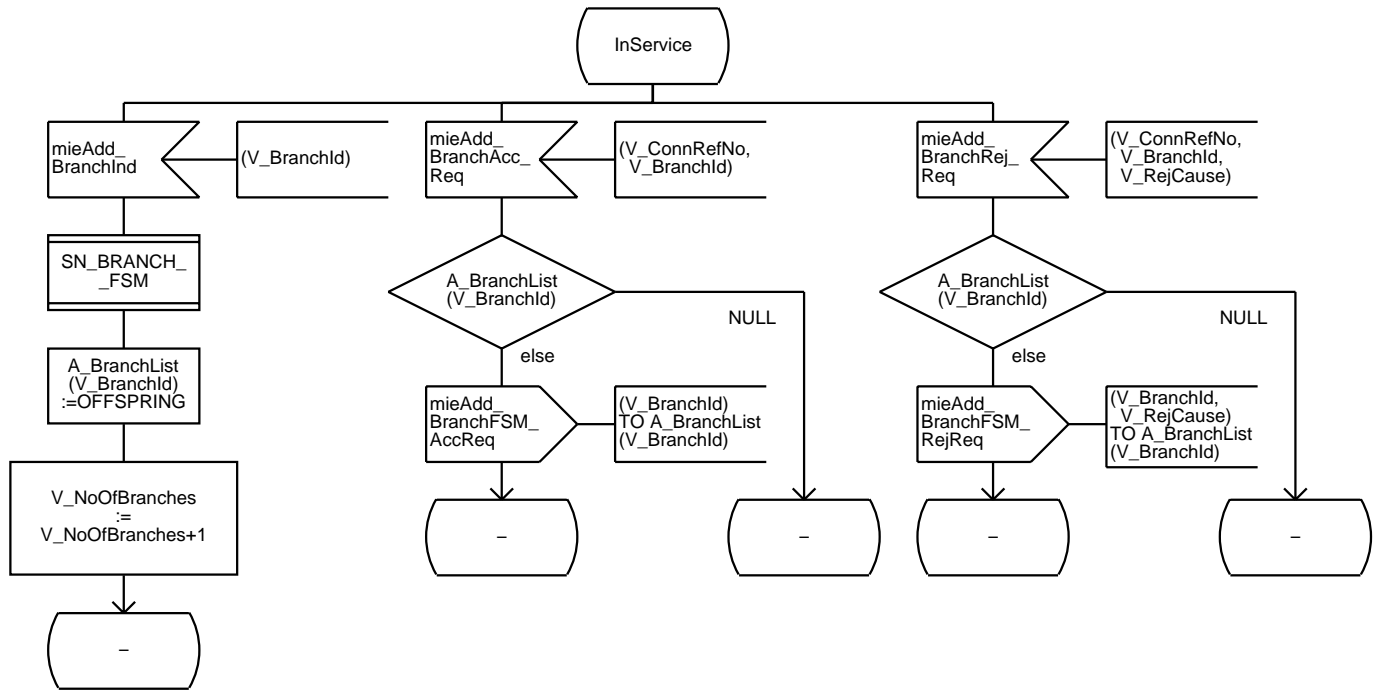
Process SN_CONN_FSM

3(10)



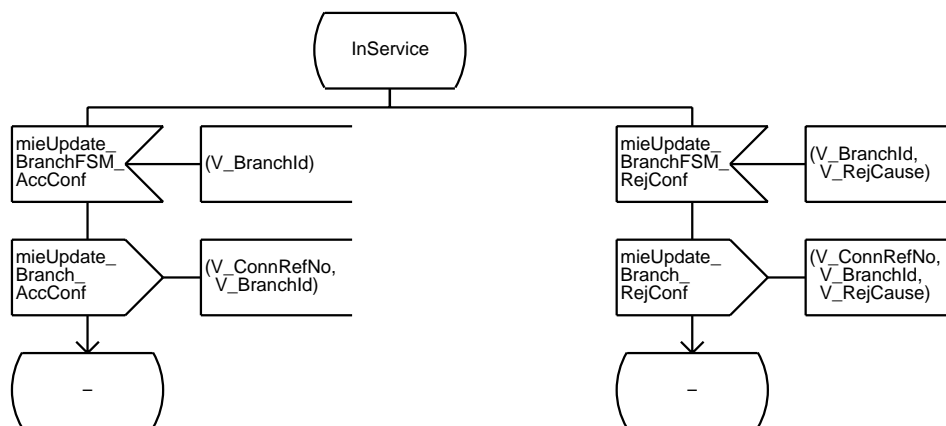
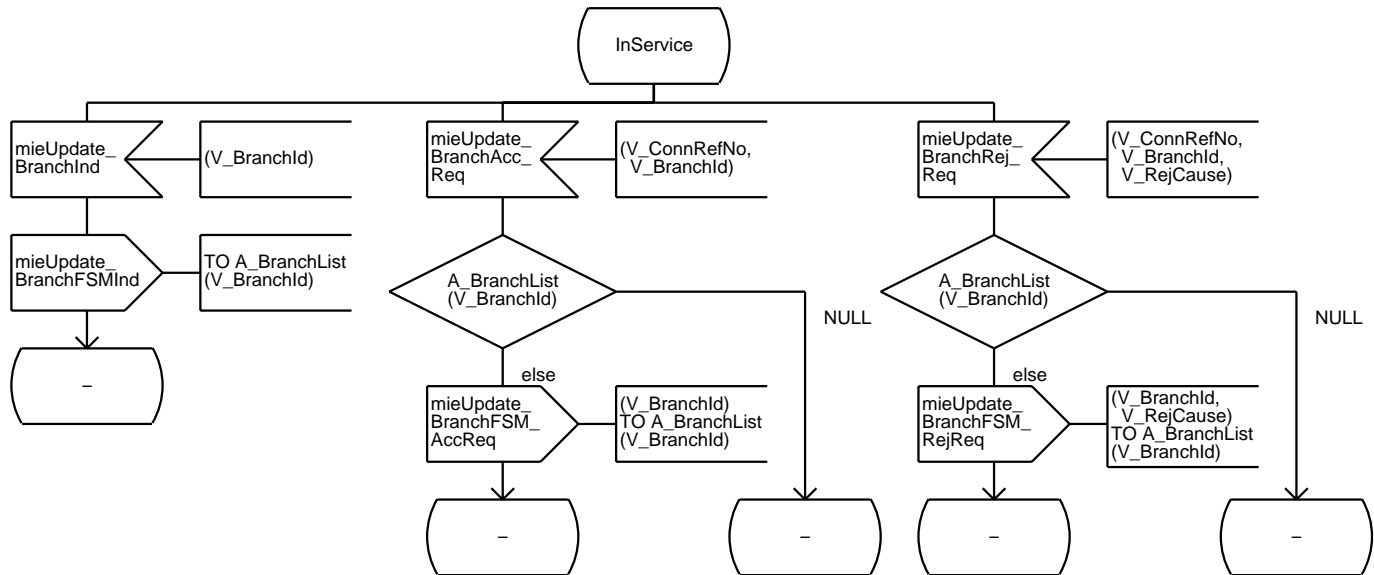
Process SN_CONN_FSM

4(10)



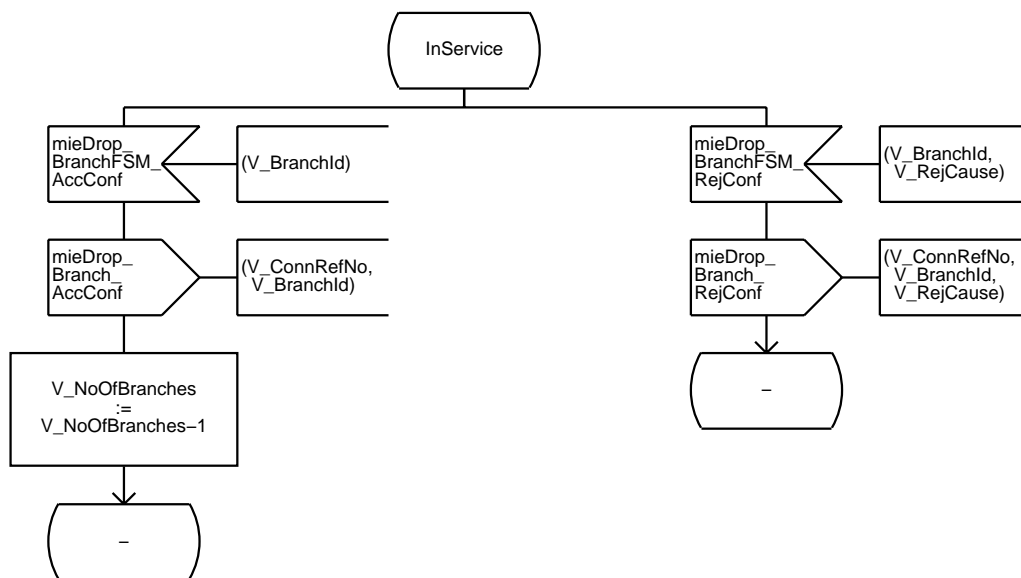
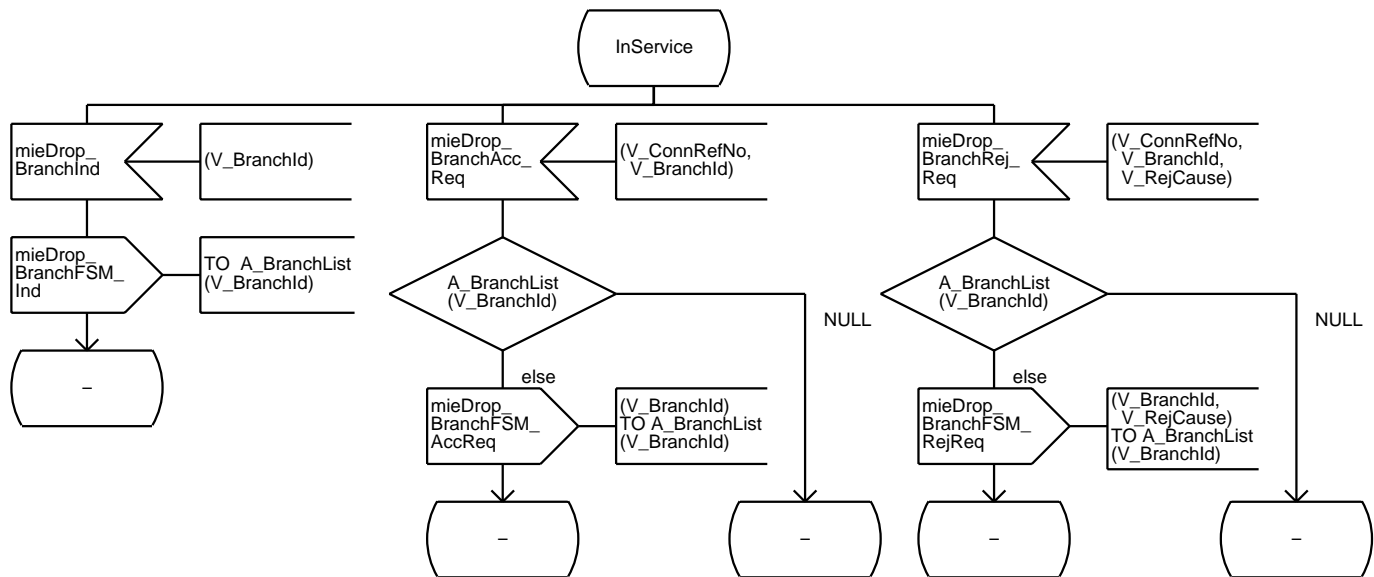
Process SN_CONN_FSM

5(10)



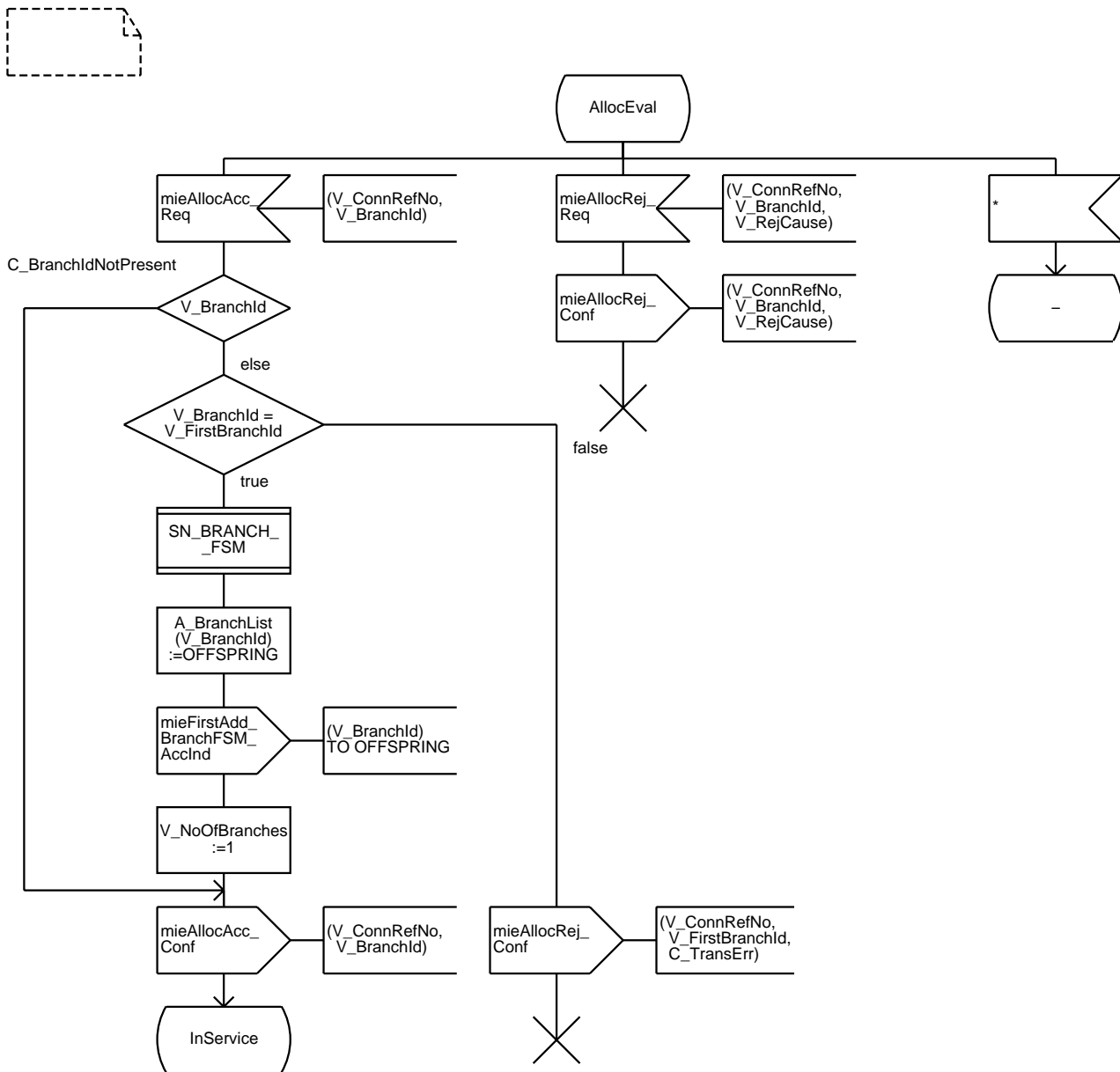
Process SN_CONN_FSM

6(10)



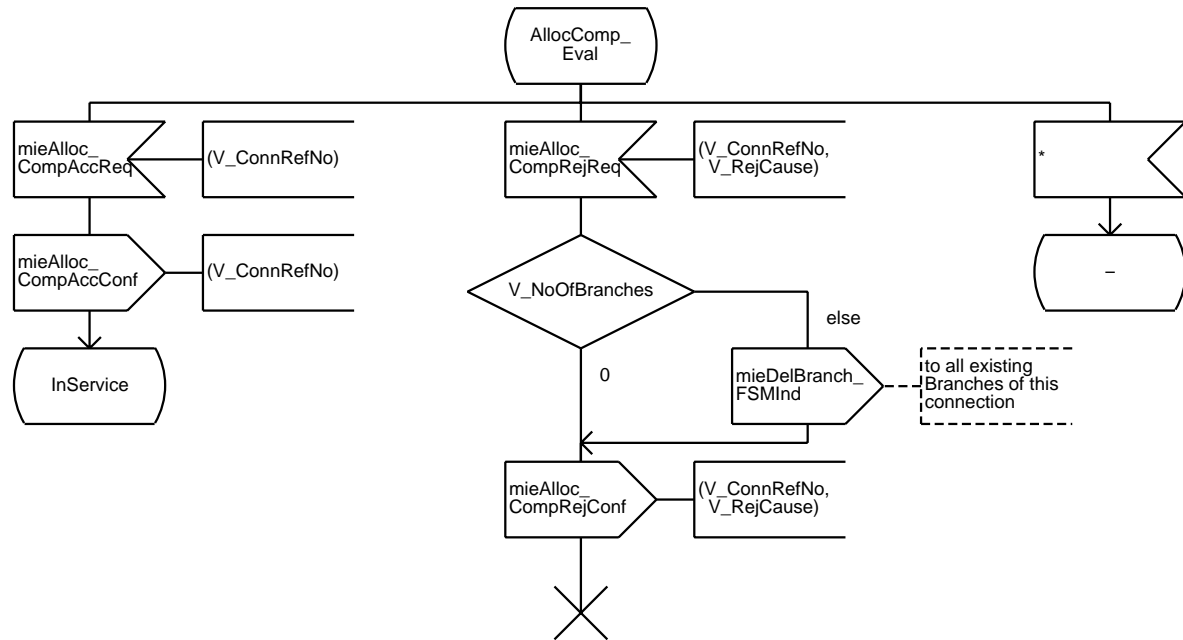
Process SN_CONN_FSM

7(10)



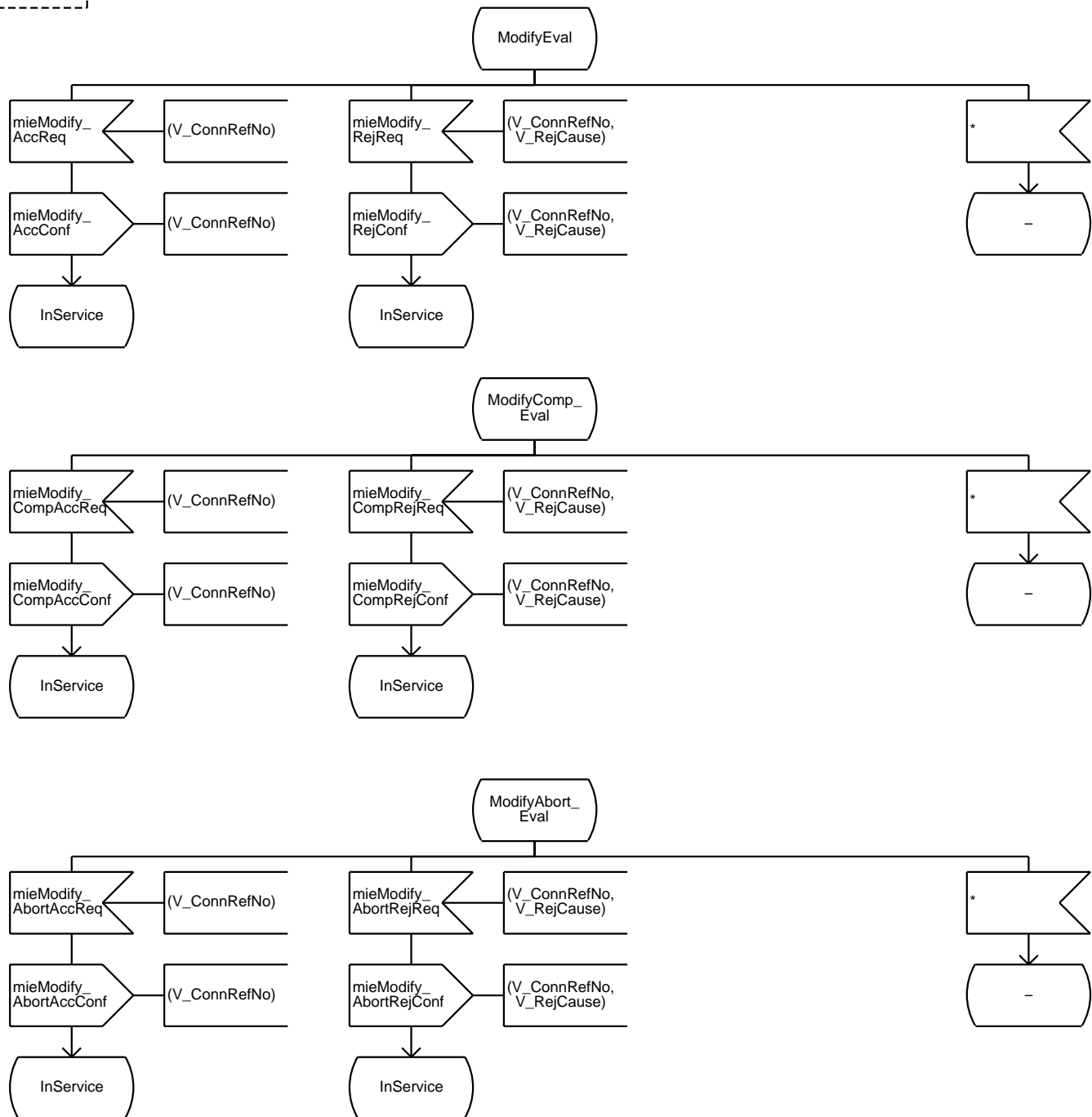
Process SN_CONN_FSM

8(10)



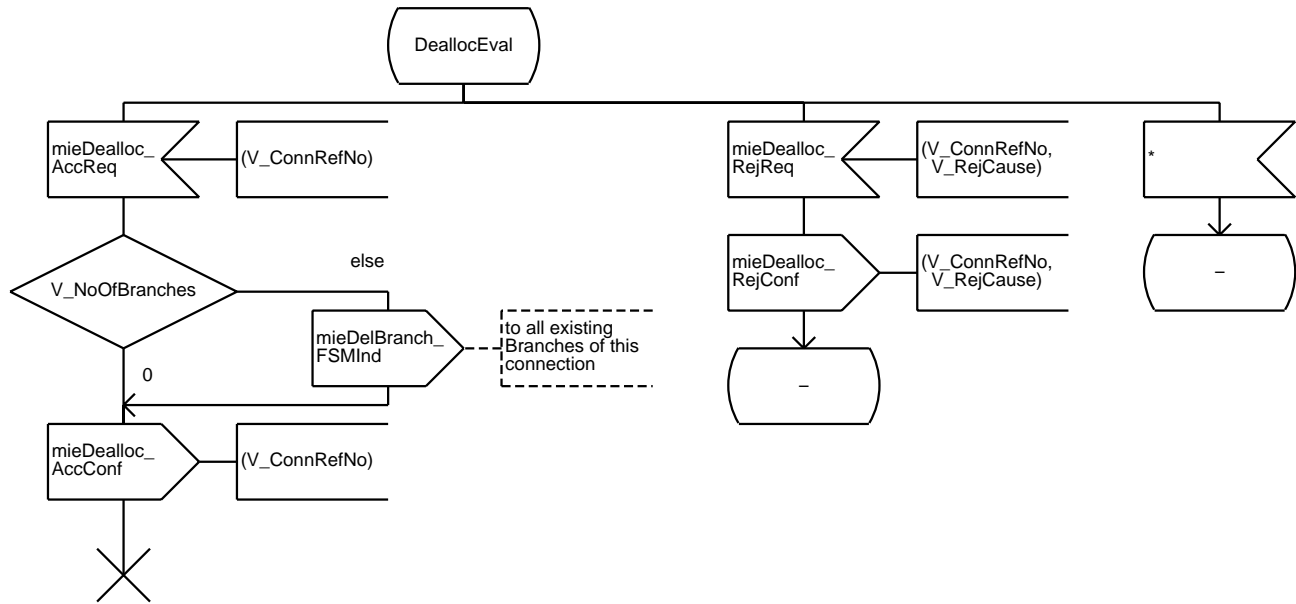
Process SN_CONN_FSM

9(10)



Process SN_CONN_FSM

10(10)



Process SN_BRANCH_FSM

1(2)



```

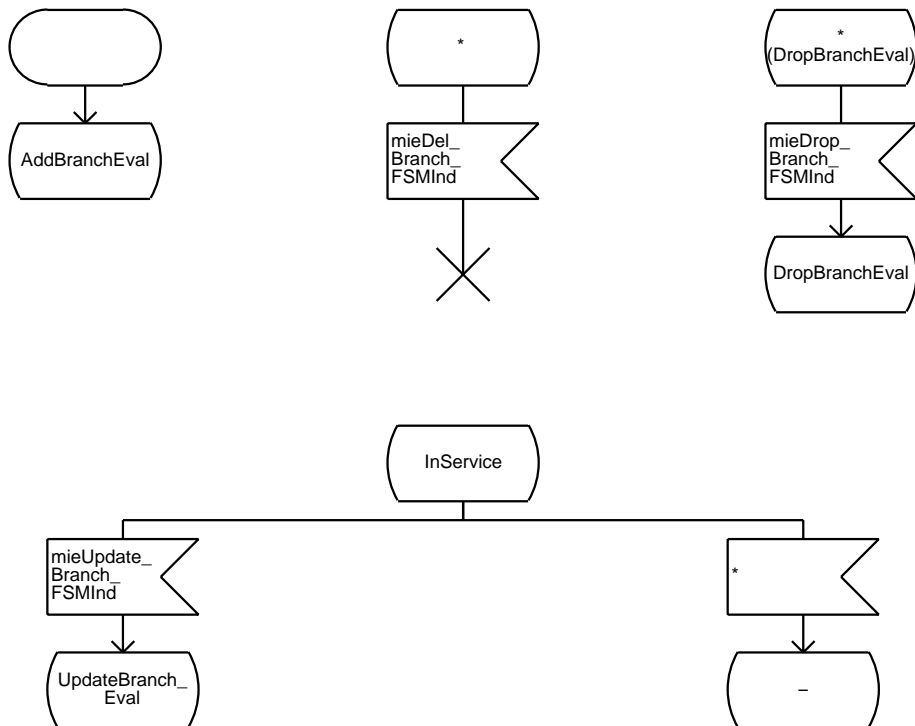
/* State descriptions
InService:   The process is not waiting for a confirmation of a message
AddBranchEval: An add branch is sent but not confirmed yet
UpdateBranchEval: An update branch is sent but not confirmed yet
DropBranchEval: An drop branch is sent but not confirmed yet
*/

```

```

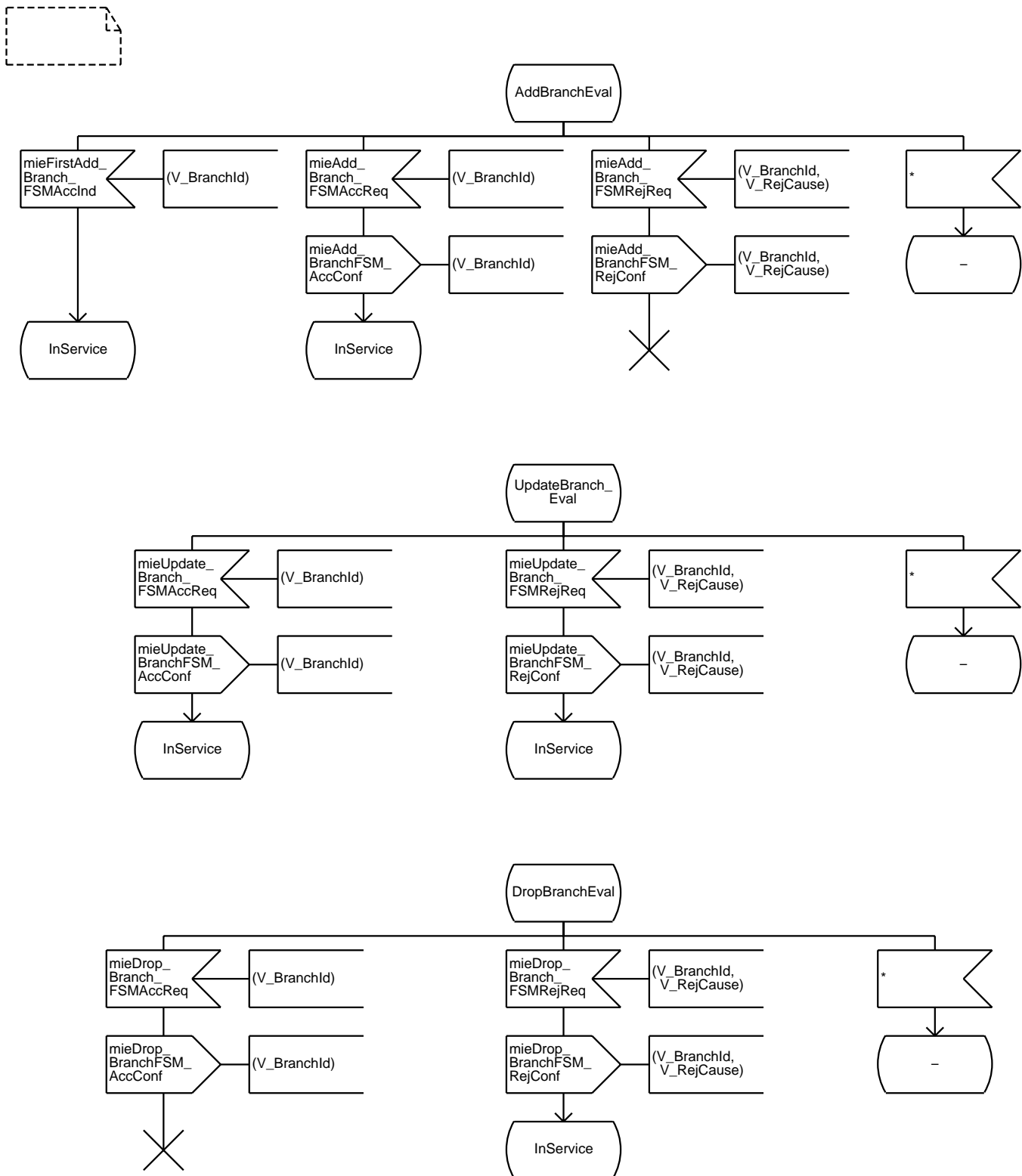
/*Primitive Data Declarations*/
/**
/* Branch identifier */
DCL
V_BranchId IT_BranchId;
/**
/* Reject Cause */
DCL
V_RejCause IT_RejCause;

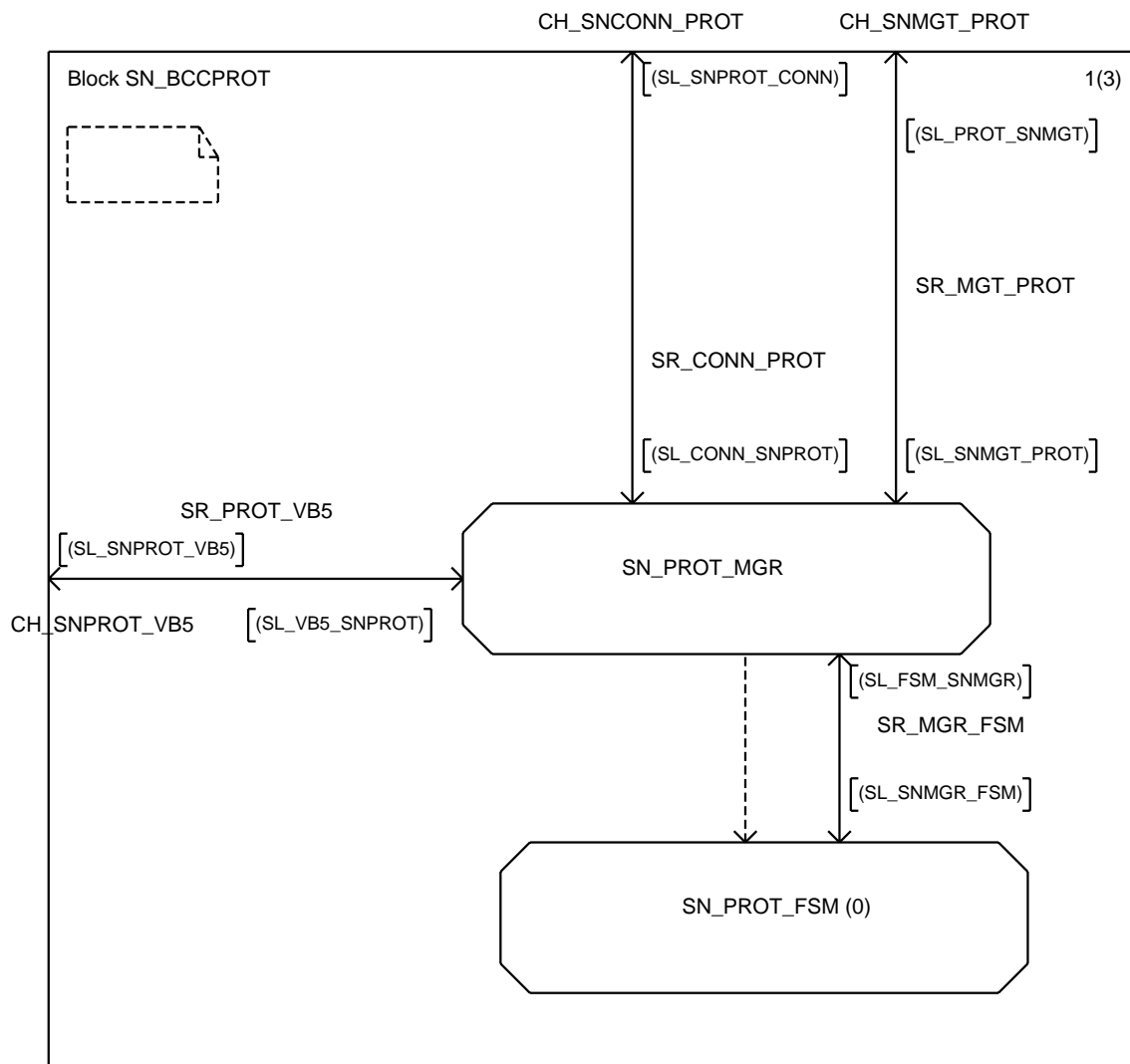
```



Process SN_BRANCH_FSM

2(2)





Block SN_BCCPROT

2(3)



/* Declaration of primitives internal
to block SN_BCCPROT */

```

signal
mieAllocReq      (IT_TransId, IT_ConnRefNo, IT_BranchId),
mieAllocAccConf  (IT_ConnRefNo, IT_BranchId),
mieAllocRejConf  (IT_ConnRefNo, IT_BranchId, IT_RejCause),
mieAllocCompReq  (IT_TransId, IT_ConnRefNo),
mieAllocCompAccConf (IT_ConnRefNo),
mieAllocCompRejConf (IT_ConnRefNo, IT_RejCause),
mieBbccResetReq  (IT_TransId),
mieBbccResetAccConf (IT_ConnRefNo),
mieBbccResetRejConf (IT_RejCause),
mieBbccPresyncReq (IT_TransId),
mieBbccPresyncAccConf (IT_ConnRefNo),
mieBbccPresyncRejConf (IT_RejCause),
mieDeallocReq    (IT_TransId, IT_ConnRefNo),
mieDeallocAccConf (IT_ConnRefNo),
mieDeallocRejConf (IT_ConnRefNo, IT_RejCause),
mieModifyReq     (IT_TransId, IT_ConnRefNo),
mieModifyAccConf (IT_ConnRefNo),
mieModifyRejConf (IT_ConnRefNo, IT_RejCause),
mieModifyCompReq (IT_TransId, IT_ConnRefNo),
mieModifyCompAccConf (IT_ConnRefNo),
mieModifyCompRejConf (IT_ConnRefNo, IT_RejCause),
mieModifyAbortReq (IT_TransId, IT_ConnRefNo),
mieModifyAbortAccConf (IT_ConnRefNo),
mieModifyAbortRejConf (IT_ConnRefNo, IT_RejCause),
mieAddBranchReq   (IT_TransId, IT_ConnRefNo, IT_BranchId),
mieAddBranchAccConf (IT_ConnRefNo, IT_BranchId),
mieAddBranchRejConf (IT_ConnRefNo, IT_BranchId, IT_RejCause),
mieUpdateBranchReq (IT_TransId, IT_ConnRefNo, IT_BranchId),
mieUpdateBranchAccConf (IT_ConnRefNo, IT_BranchId),
mieUpdateBranchRejConf (IT_ConnRefNo, IT_BranchId, IT_RejCause),
mieDropBranchReq   (IT_TransId, IT_ConnRefNo, IT_BranchId),
mieDropBranchAccConf (IT_ConnRefNo, IT_BranchId),
mieDropBranchRejConf (IT_ConnRefNo, IT_BranchId, IT_RejCause),
mieAnFaultInd      (IT_ConnRefNo, IT_BranchId);

```


Block SN_BCCPROT

3(3)



```
/* Declaration of signallists
   internal to block SN_BCCPROT */
```

```
signallist SL_FSM_SNMGR =
bccAlloc,
bccAllocComp,
bccBbccReset,
bccPresync,
bccDealloc,
bccModify,
bccModifyComp,
bccModifyAbort,
bccAddBranch,
bccUpdateBranch,
bccDropBranch,
bccAnFaultAcc,
mieAllocAccConf,
mieAllocRejConf,
mieAllocCompAccConf,
mieAllocCompRejConf,
mieBbccResetAccConf,
mieBbccResetRejConf,
mieBbccPresyncAccConf,
mieBbccPresyncRejConf,
mieDeallocAccConf,
mieDeallocRejConf,
mieModifyAccConf,
mieModifyRejConf,
mieModifyCompAccConf,
mieModifyCompRejConf,
mieModifyAbortAccConf,
mieModifyAbortRejConf,
mieAddBranchAccConf,
mieAddBranchRejConf,
mieUpdateBranchAccConf,
mieUpdateBranchRejConf,
mieDropBranchAccConf,
mieDropBranchRejConf,
mieAnFaultInd,
mieProtocolError,
mieErrorInd;
```

```
signallist SL_SNMGR_FSM =
mieAllocReq,
mieAllocCompReq,
mieBbccResetReq,
mieBbccPresyncReq,
mieDeallocReq,
mieModifyReq,
mieModifyCompReq,
mieModifyAbortReq,
mieAddBranchReq,
mieUpdateBranchReq,
mieDropBranchReq,
bccAllocAcc,
bccAllocRej,
bccAllocCompAcc,
bccAllocCompRej,
bccBbccResetAcc,
bccBbccResetRej,
bccPresyncAcc,
bccPresyncRej,
bccDeallocAcc,
bccModifyAcc,
bccModifyRej,
bccModifyCompAcc,
bccModifyCompRej,
bccModifyAbortAcc,
bccModifyAbortRej,
bccAddBranchAcc,
bccAddBranchRej,
bccUpdateBranchAcc,
bccUpdateBranchRej,
bccDropBranchAcc,
bccDropBranchRej,
bccAnFault,
bccProtocolError,
mieAbortReq;
```

Process SN_PROT_MGR

1(14)

/*SN BCC protocol management entity
controls instantiation of
protocol FSMs (process SN_PROT_FSM) */

/* The generation of the transaction identifier, i.e. the increment by one
for each invoke message, is for simulation purposes only. In an
implementation, any other method which is appropriate may be applied */

/* Data type definitions internal to the process */
/**/
/* This data type is used for translation of the remote transaction identifier to the process identifier (PID)
of the process handling the corresponding transaction in the SN */
NEWTYPE AT_RemTransList ARRAY
(IT_TransId, PID)
ENDNEWTYPE AT_RemTransList;
/**/
/* This data type is used for translation of the local transaction identifier to the process identifier (PID)
of the process handling the corresponding transaction in the SN */
NEWTYPE AT_LocTransList ARRAY
(IT_TransId, PID)
ENDNEWTYPE AT_LocTransList;

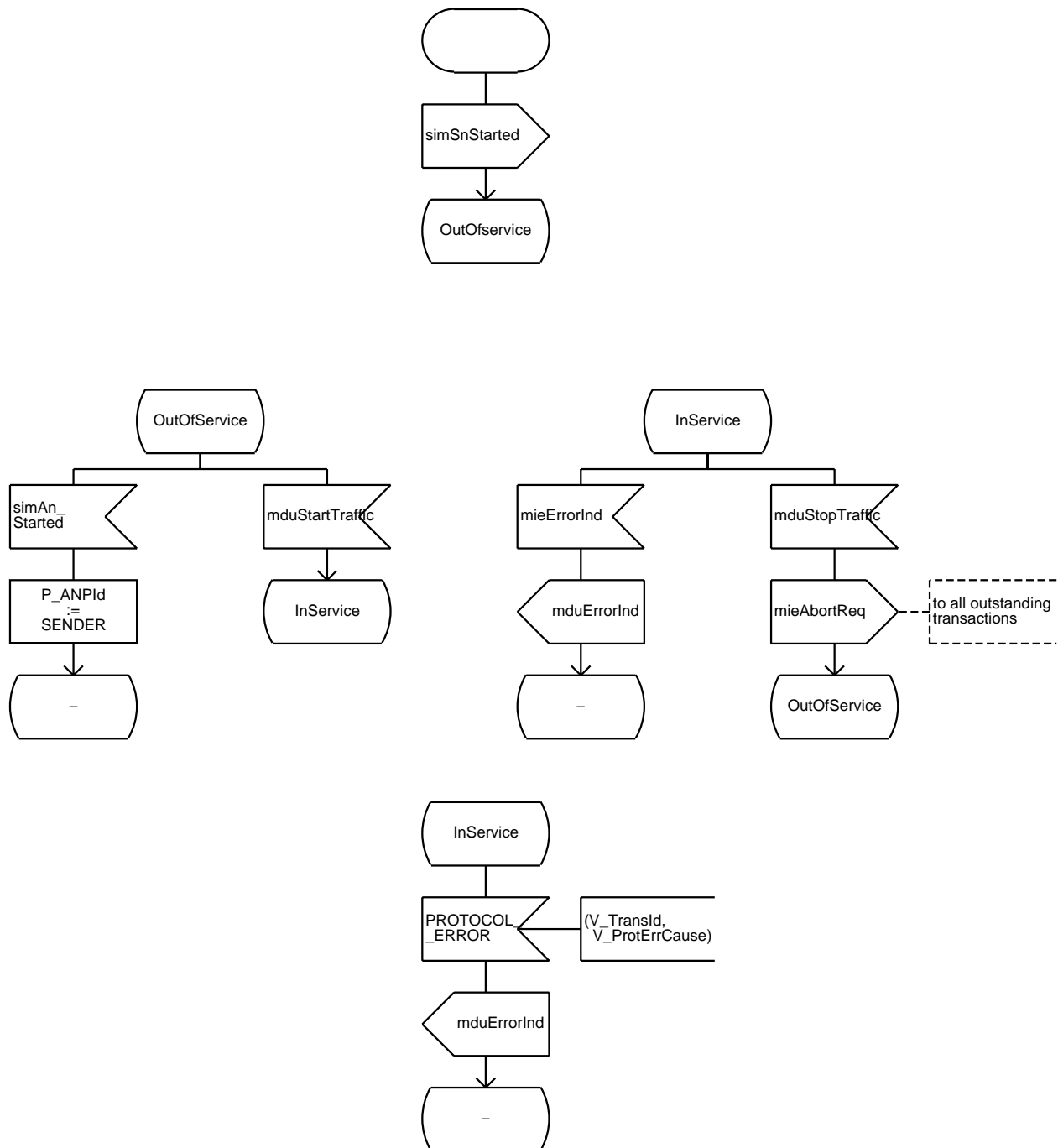
/*Primitive Data Declarations*/
/**/
/* Protocol error cause */
DCL
V_ProtErrCause IT_ProtErrCause;
/**/
/* Transaction identifier */
DCL
V_TransId IT_TransId;
/**/
/* Connection reference number */
DCL
V_ConnRefNo IT_ConnRefNo;
/**/
/* Branch identifier */
DCL
V_BranchId IT_BranchId;
/**/
/* Reject cause */
DCL
V_RejCause IT_RejCause;

/* Process Data Declarations */
/**/
/* List of SN_PROT_FSMs */
DCL
A_RemTransList AT_RemTransList,
A_LocTransList AT_LocTransList;
/**/
/* PID of the receiver process in the AN */
DCL
P_ANPid PID;
/* Counter for transaction identifier */
DCL
V_NewTransId IT_TransId := 1;

Process SN_PROT_MGR

2(14)

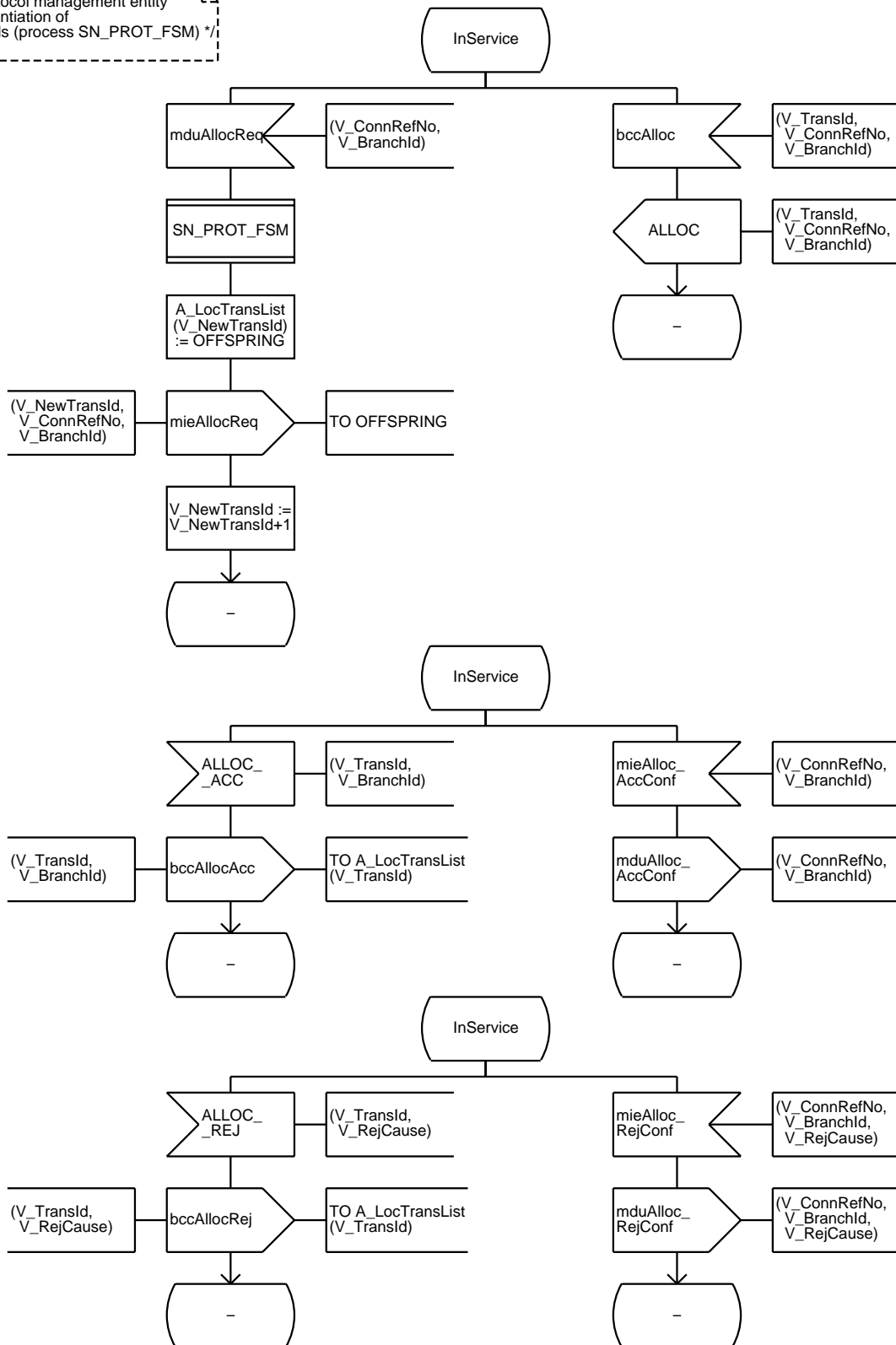
/*SN BCC protocol management entity controls instantiation of protocol FSMs (process SN_PROT_FSM) */



Process SN_PROT_MGR

3(14)

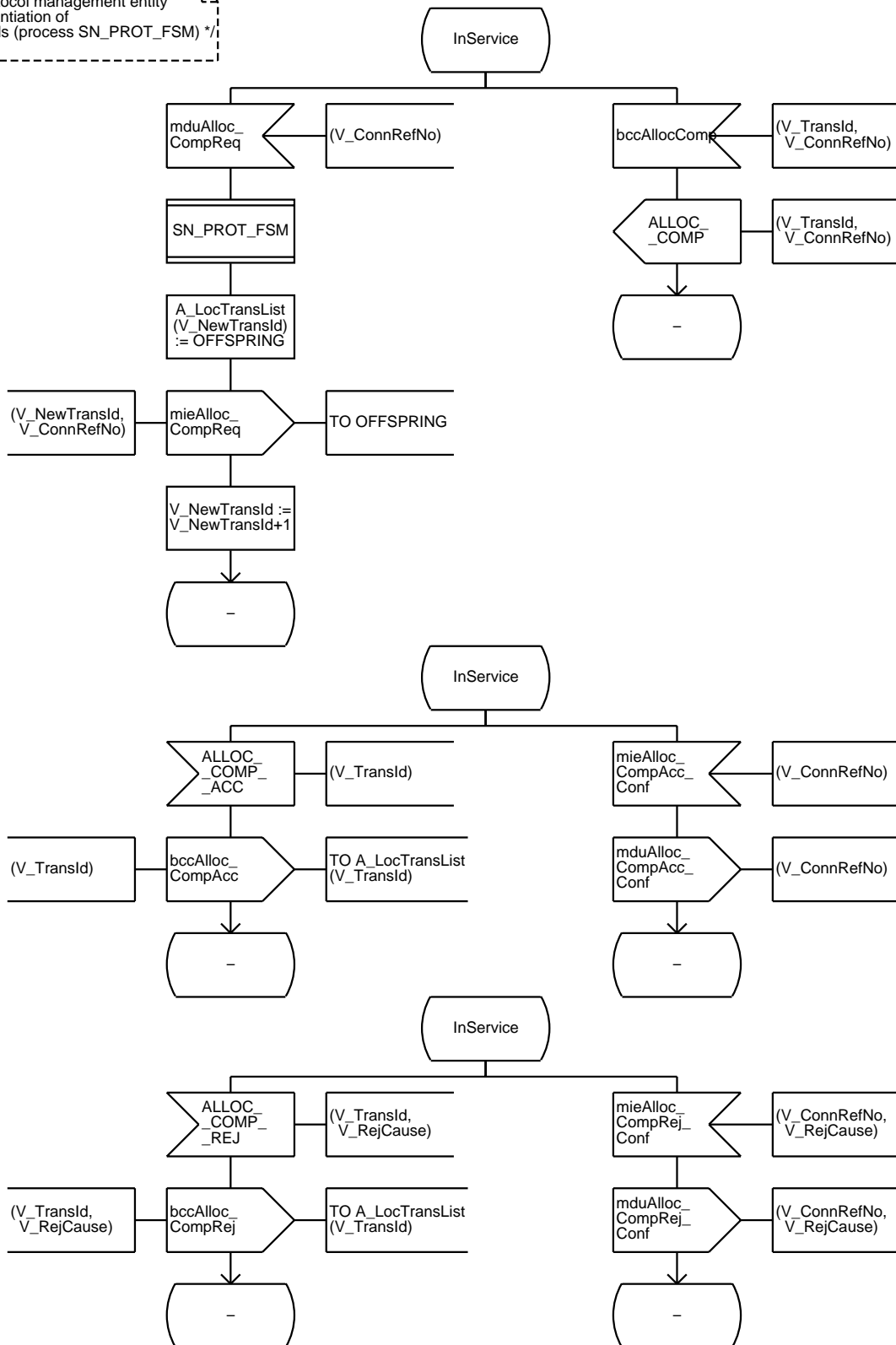
/*SN BCC protocol management entity
controls instantiation of
protocol FSMs (process SN_PROT_FSM) */



Process SN_PROT_MGR

4(14)

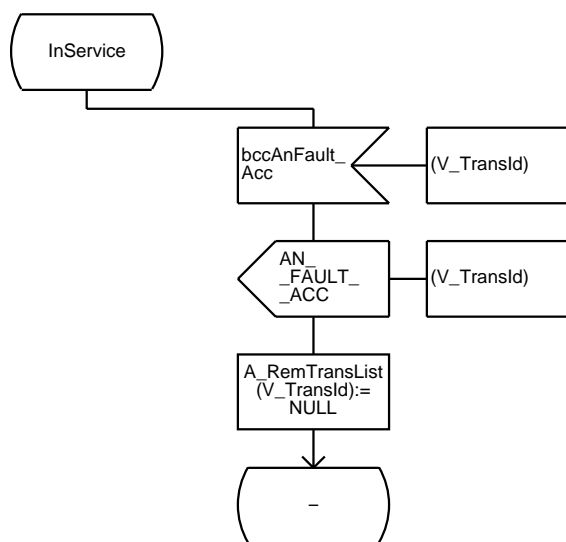
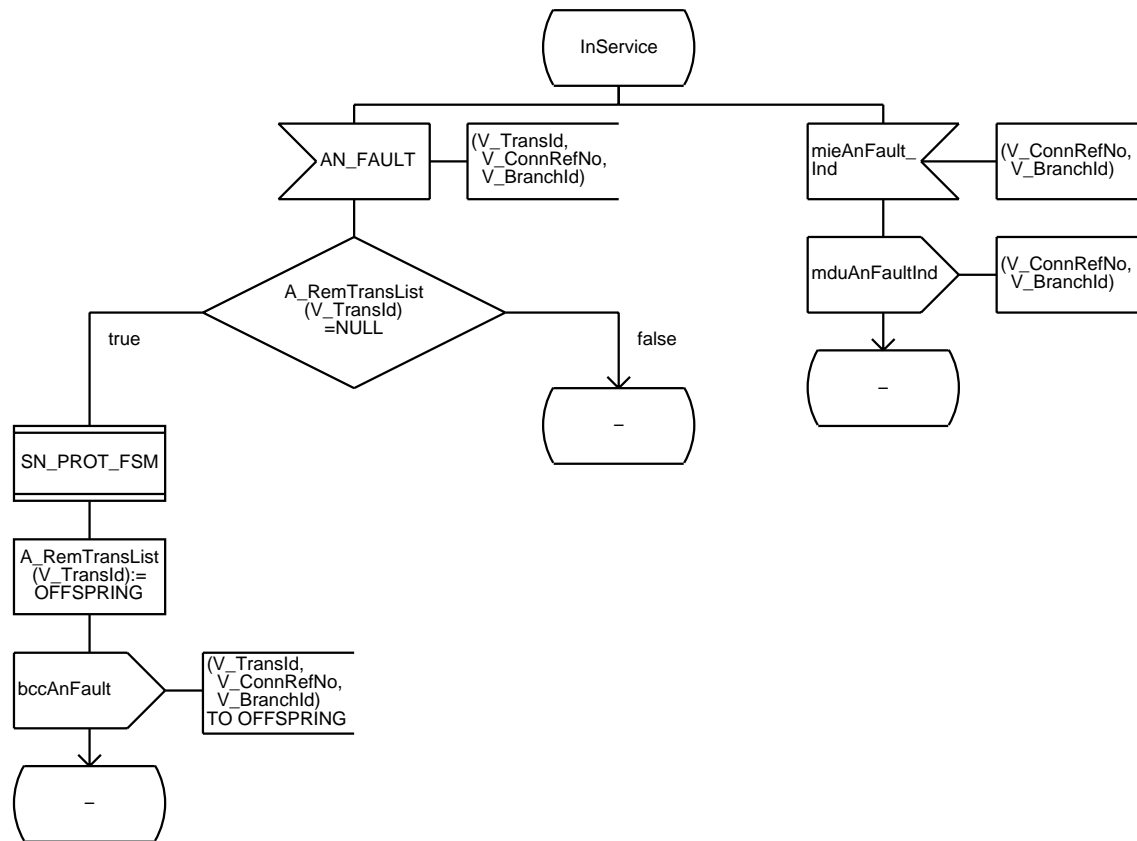
/*SN BCC protocol management entity
controls instantiation of
protocol FSMs (process SN_PROT_FSM) */



Process SN_PROT_MGR

5(14)

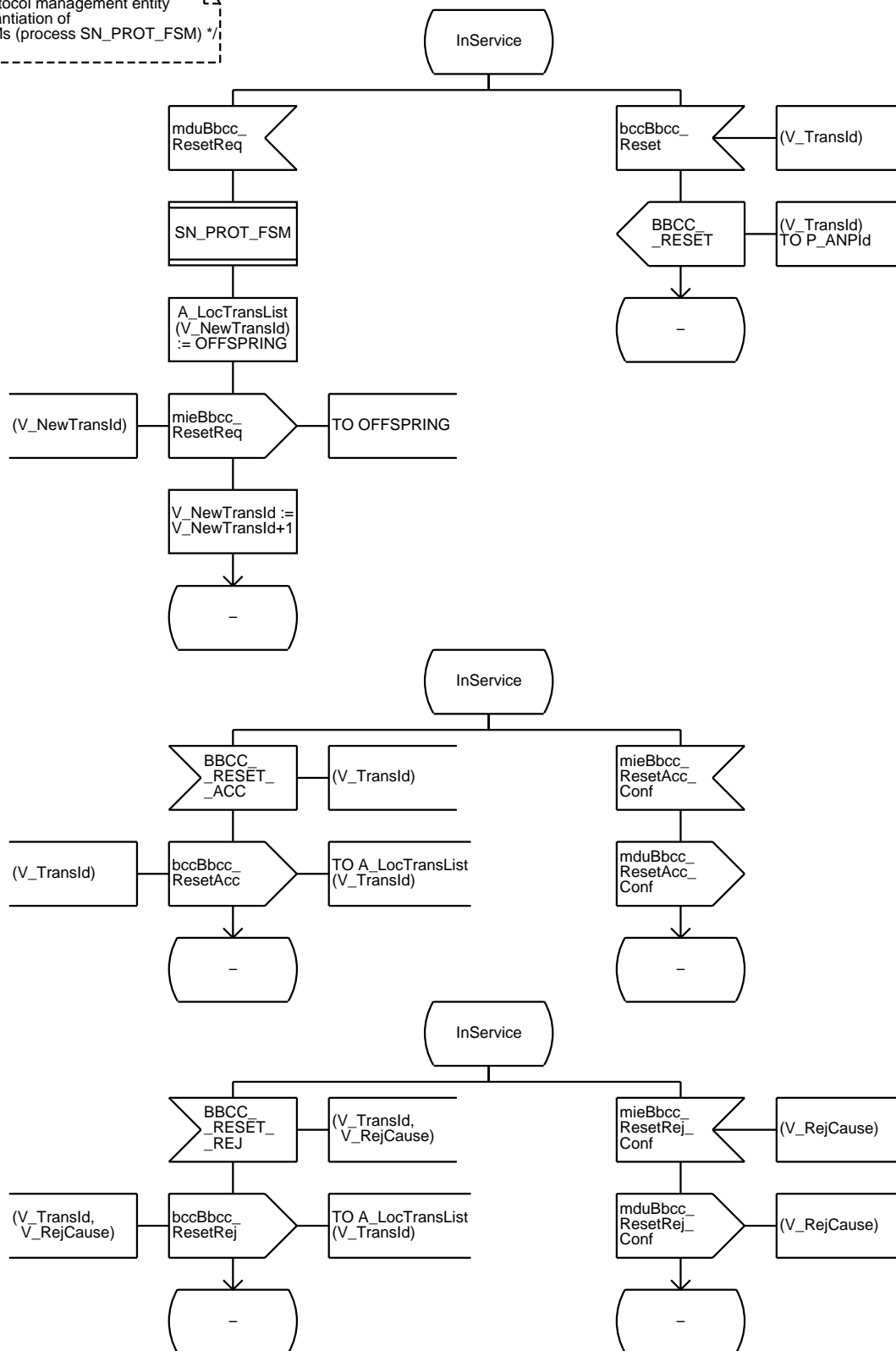
/*SN BCC protocol management entity
controls instantiation of
protocol FSMs (process SN_PROT_FSM) */



Process SN_PROT_MGR

6(14)

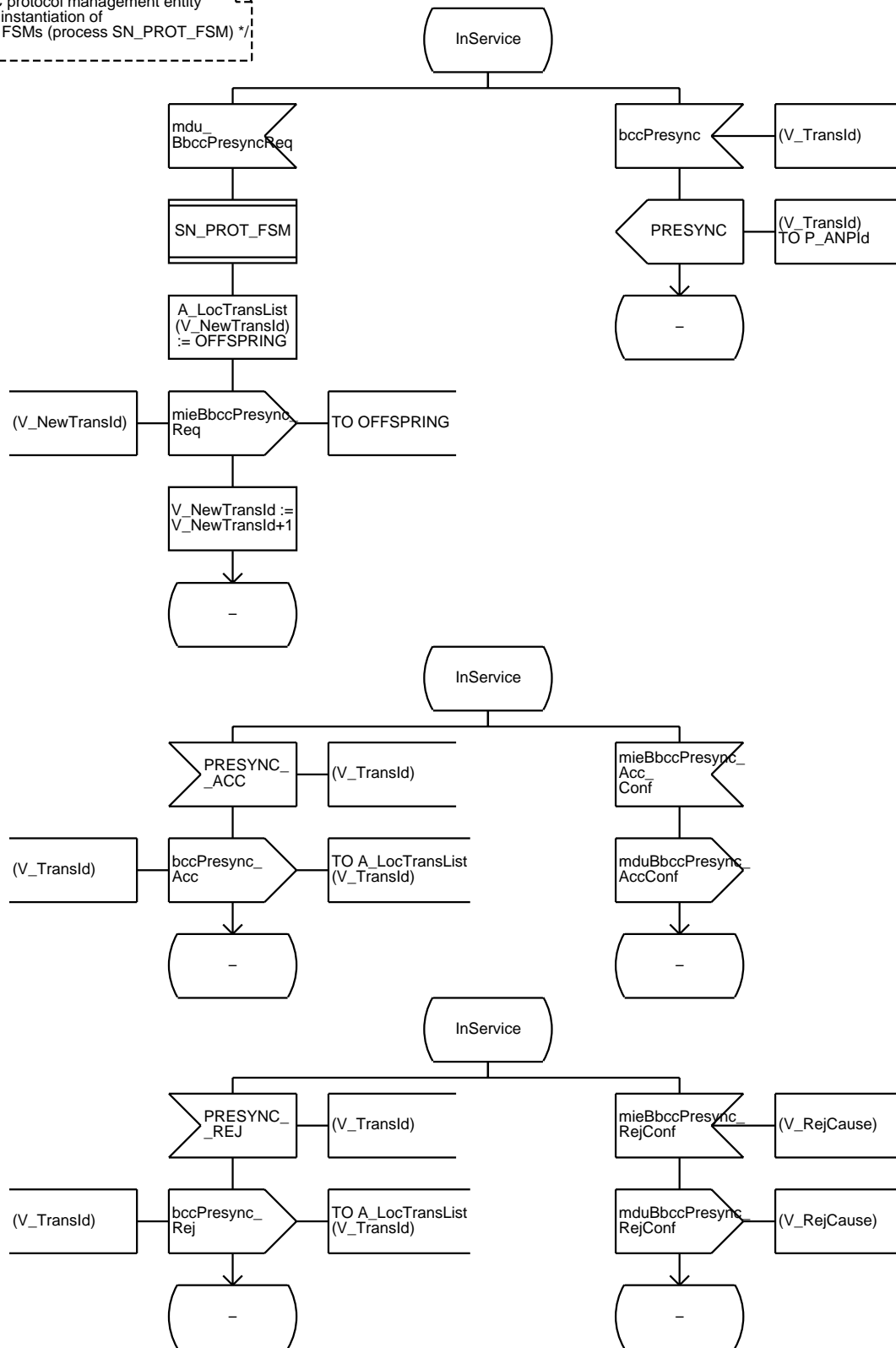
/*SN BCC protocol management entity
controls instantiation of
protocol FSMs (process SN_PROT_FSM) */



Process SN_PROT_MGR

7(14)

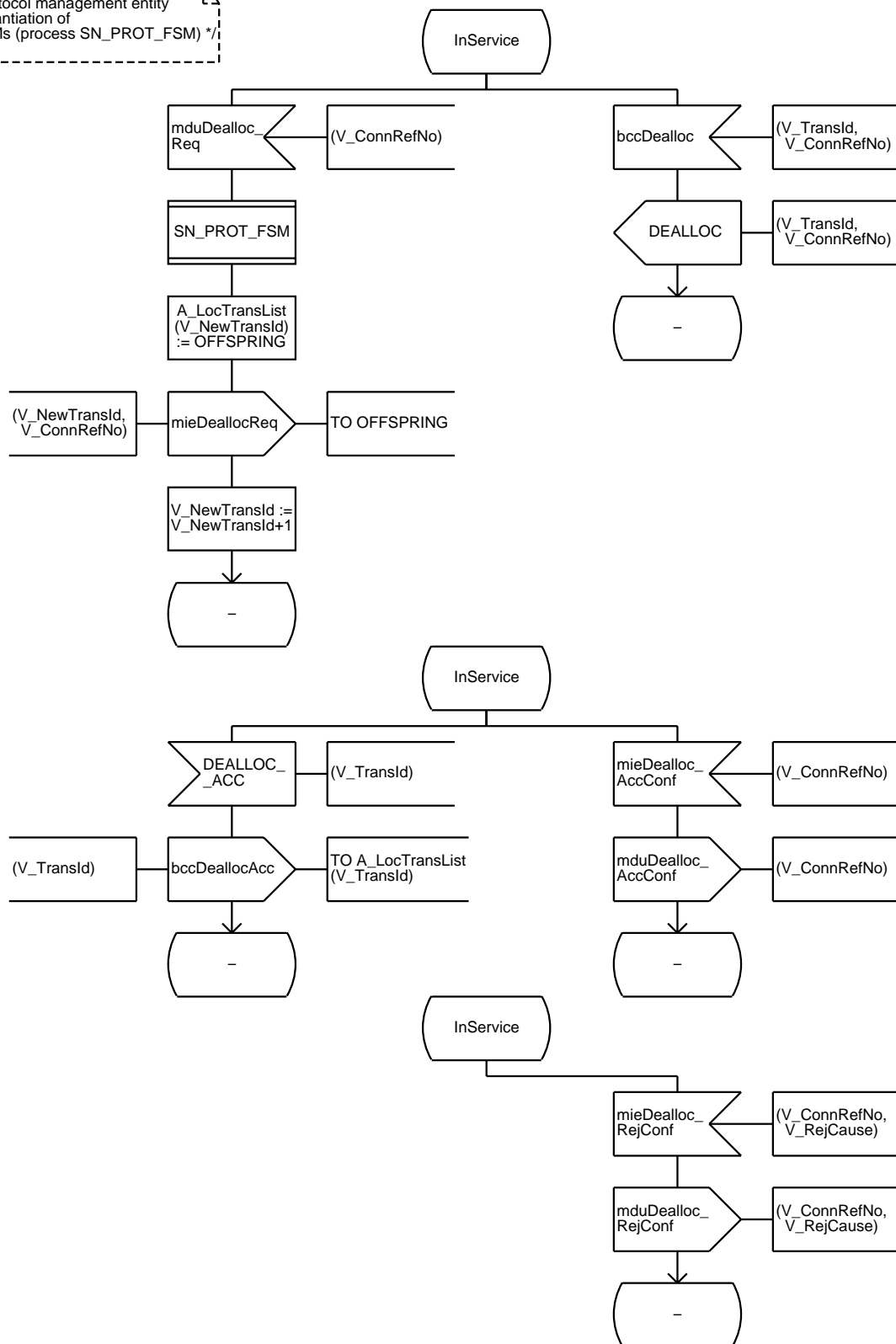
/*SN BCC protocol management entity
controls instantiation of
protocol FSMs (process SN_PROT_FSM) */



Process SN_PROT_MGR

8(14)

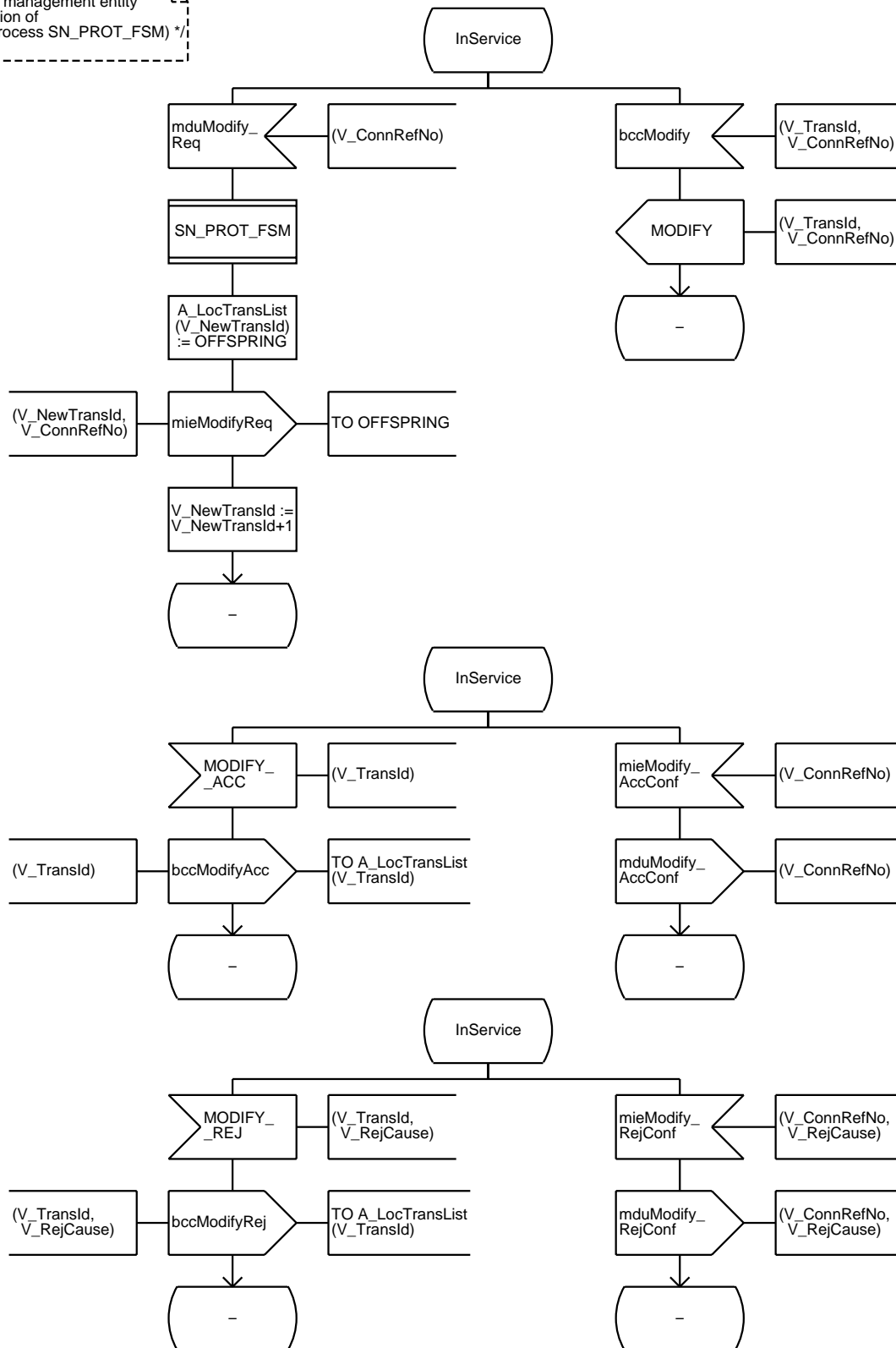
/*SN BCC protocol management entity
controls instantiation of
protocol FSMs (process SN_PROT_FSM) */



Process SN_PROT_MGR

9(14)

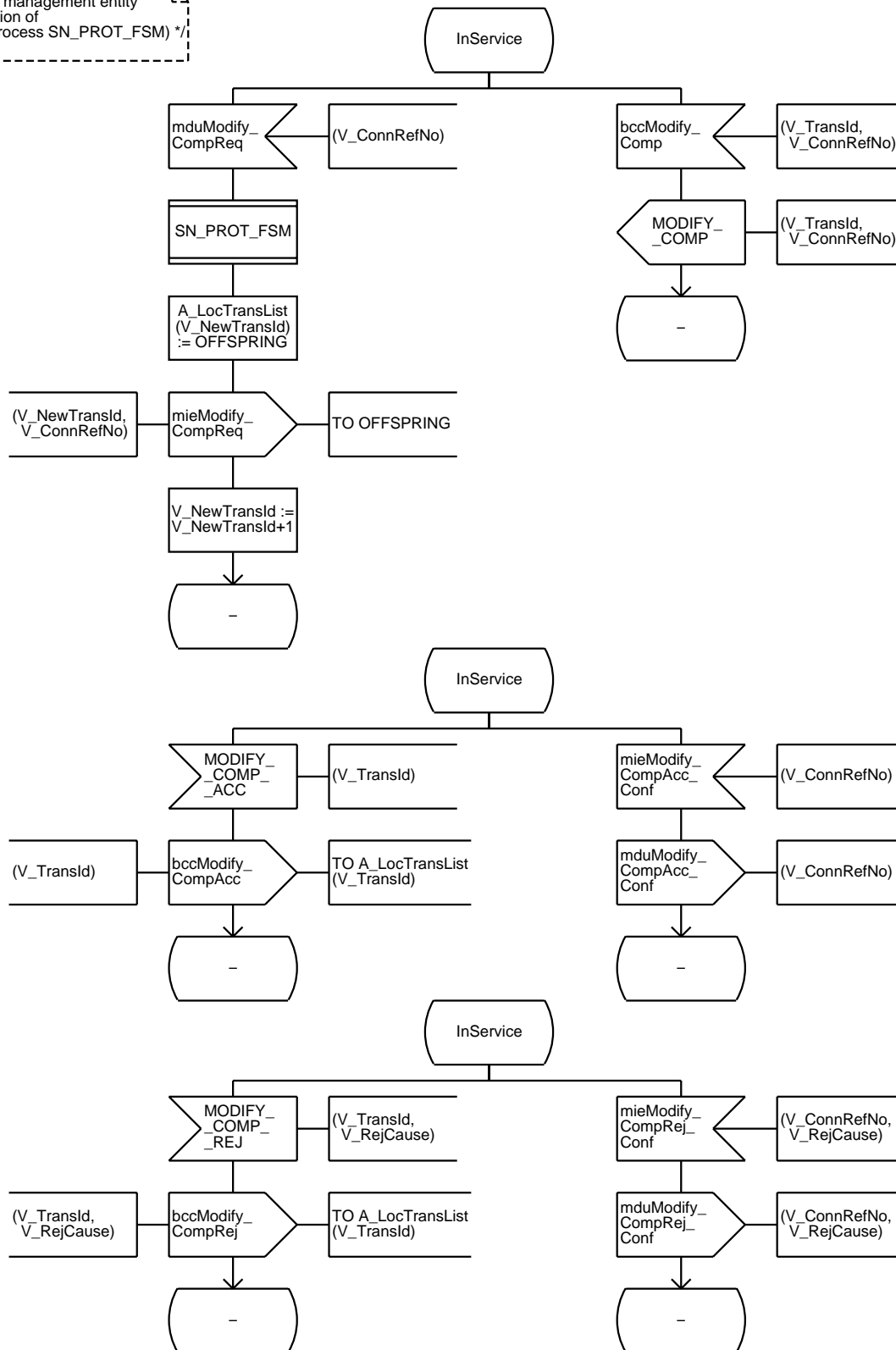
/*SN BCC protocol management entity
controls instantiation of
protocol FSMs (process SN_PROT_FSM)*/



Process SN_PROT_MGR

10(14)

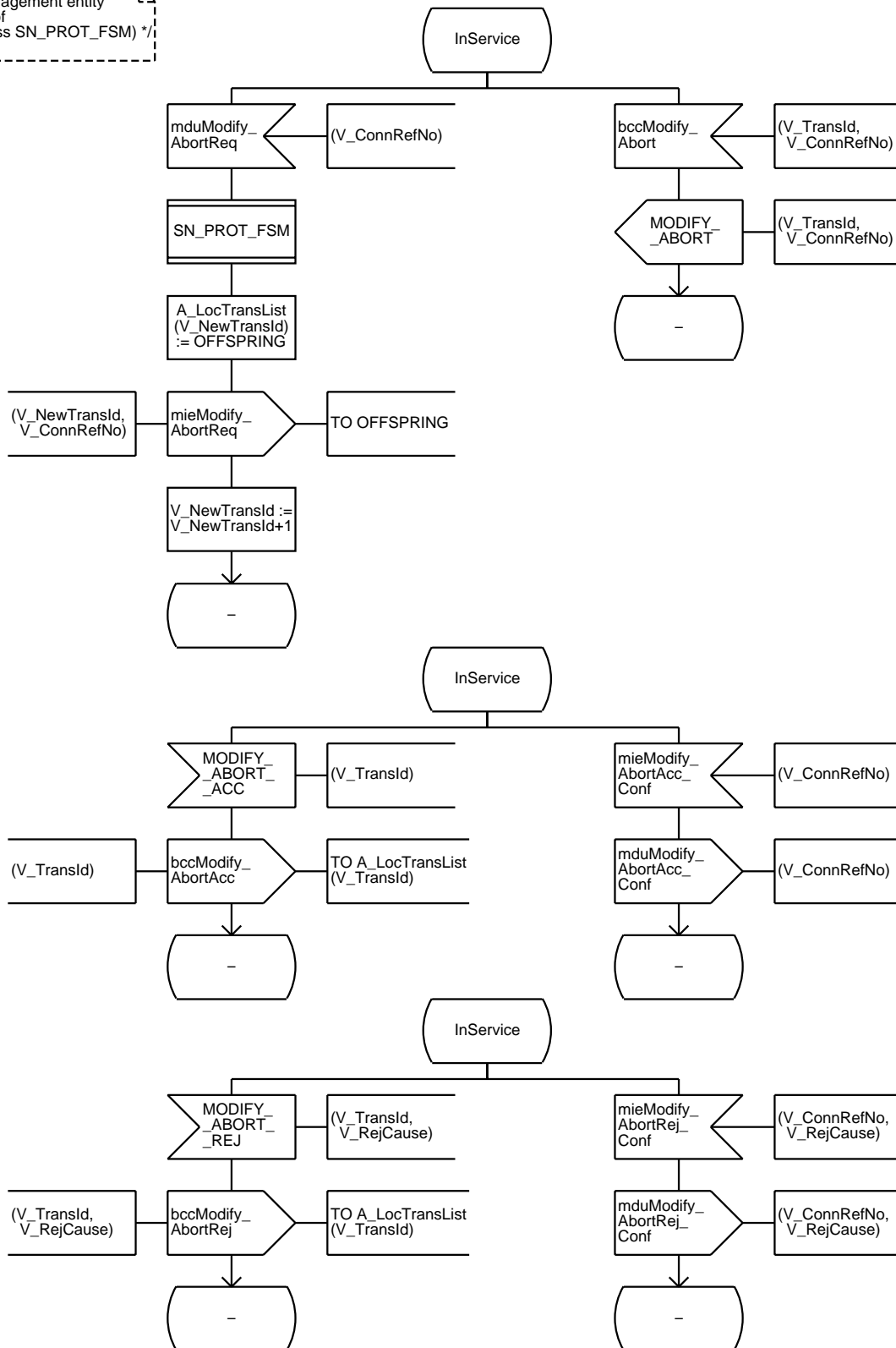
/*SN BCC protocol management entity
controls instantiation of
protocol FSMs (process SN_PROT_FSM)*/



Process SN_PROT_MGR

11(14)

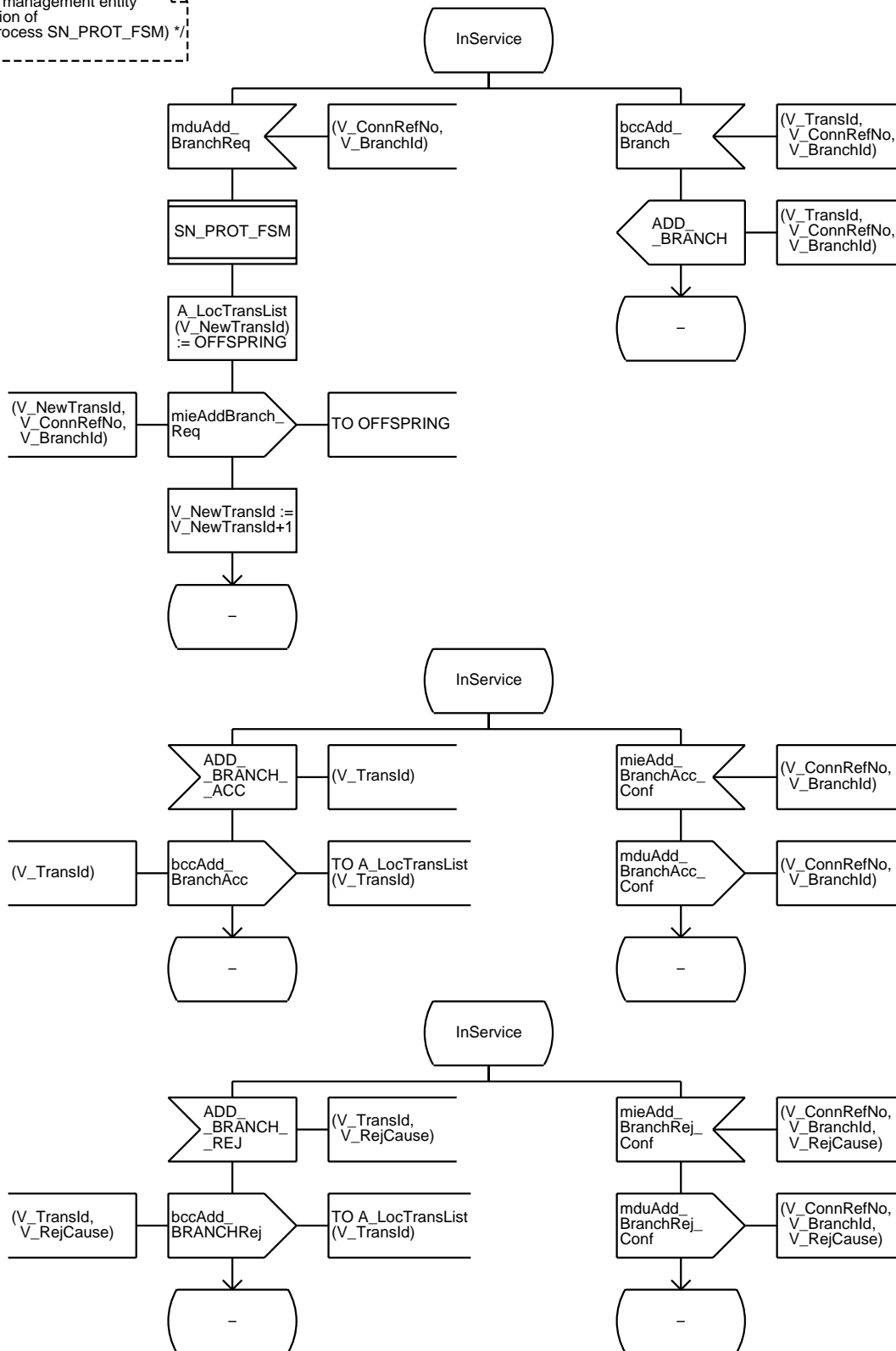
/*SN BCC protocol management entity
controls instantiation of
protocol FSMs (process SN_PROT_FSM) */



Process SN_PROT_MGR

12(14)

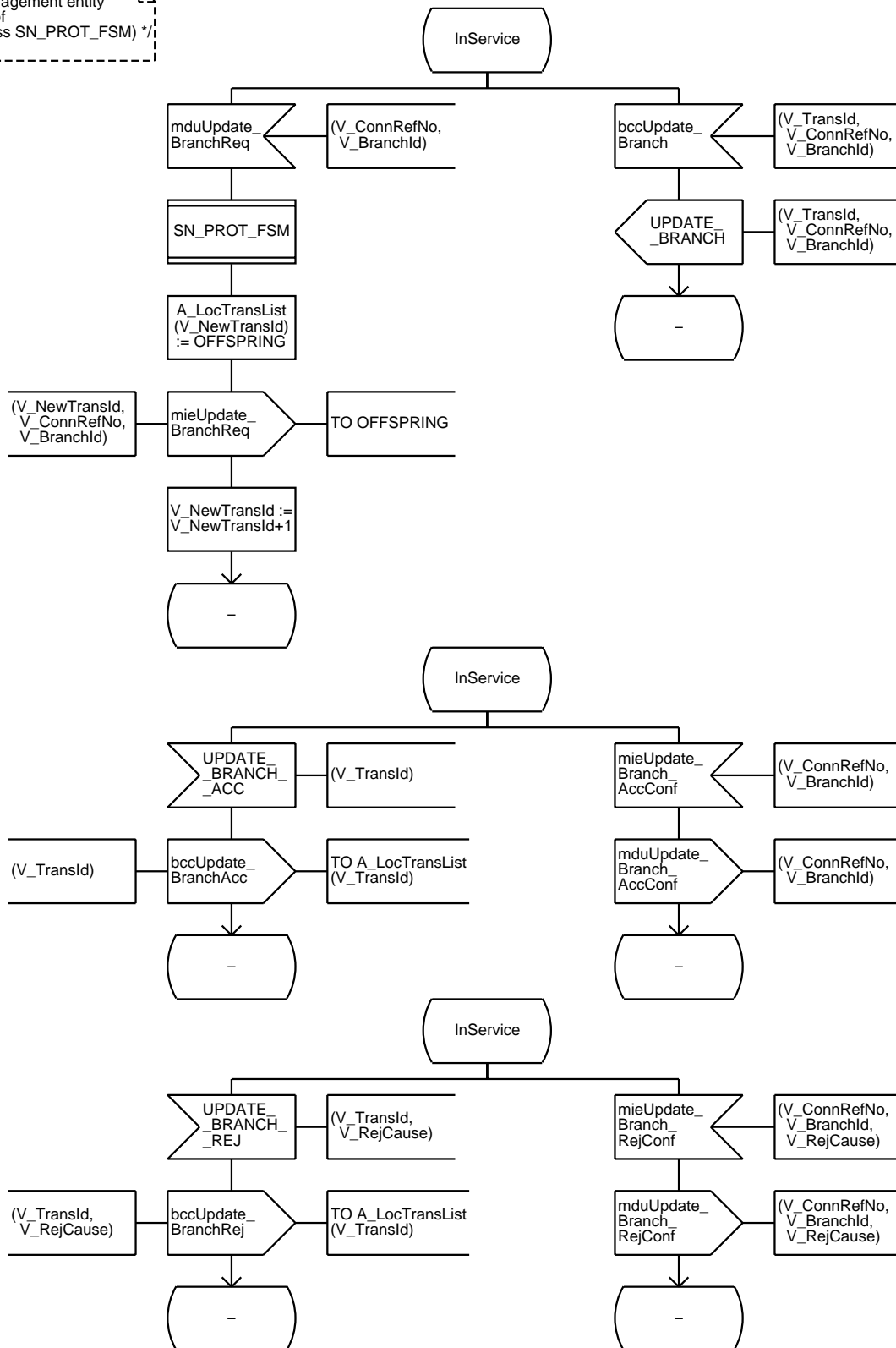
/*SN BCC protocol management entity
controls instantiation of
protocol FSMs (process SN_PROT_FSM)*/



Process SN_PROT_MGR

13(14)

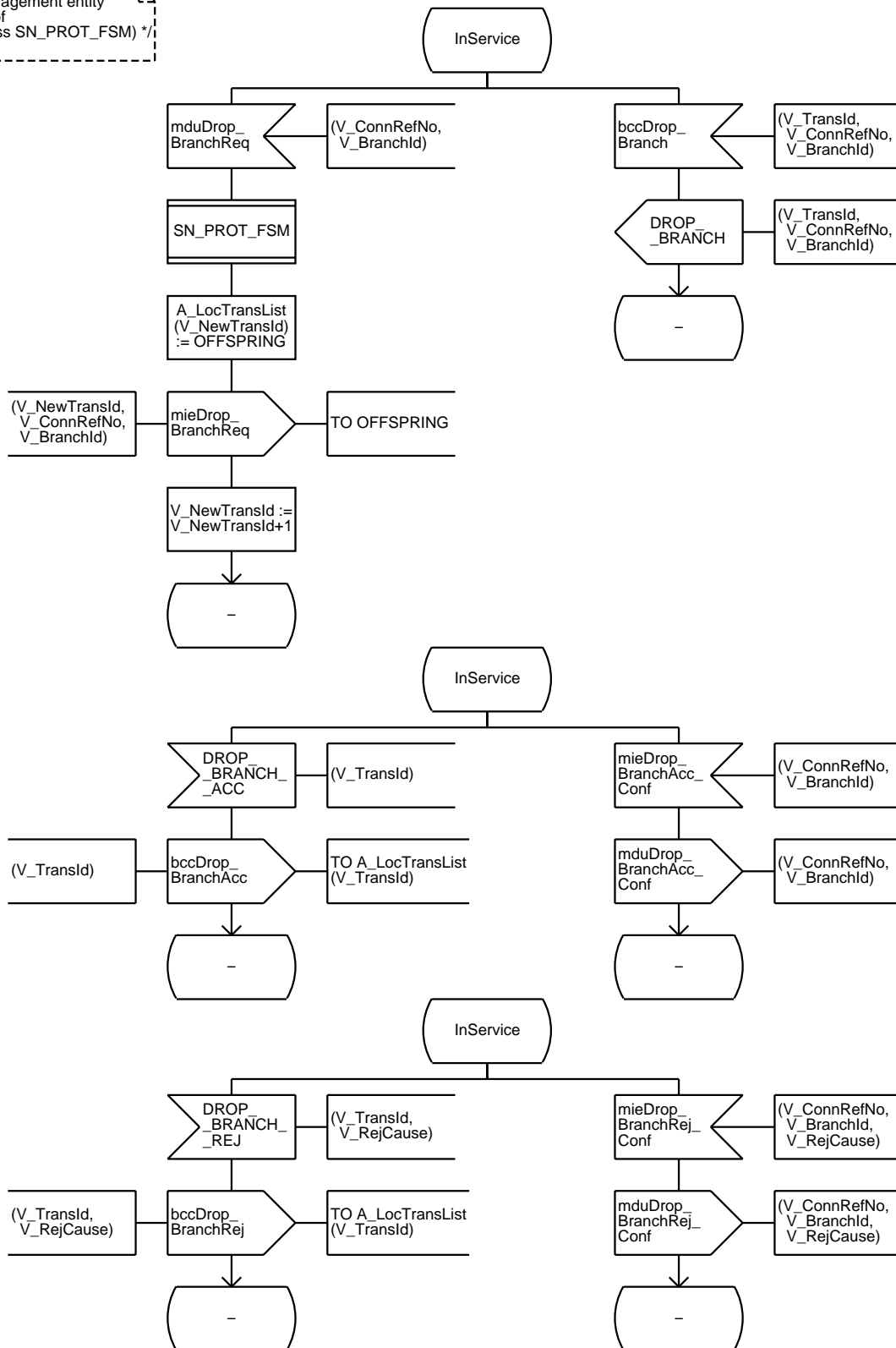
/*SN BCC protocol management entity
controls instantiation of
protocol FSMs (process SN_PROT_FSM) */



Process SN_PROT_MGR

14(14)

/*SN BCC protocol management entity
controls instantiation of
protocol FSMs (process SN_PROT_FSM) */



Process SN_PROT_FSM

1(19)

```

/*SN_PROT_FSM is responsible for sending
receiving BCC messages*/

```

```

/* Definitions and declarations for
SN_PROT_FSM */

```

```

/* State descriptions

```

```

InService: Init State of SN_PROT_FSM, process is waiting for initial SN_SYSMGT primitives or AN messages
AwaitLocAck: SN_PROT_FSM has passed an AN request to SN_SYSMGT and is now awaiting the SN_SYSMGT response
AwaitRemAck: SN_PROT_FSM has sent a VB5 message to the AN and is now awaiting the AN response
*/

```

```

/* Timerdefinitions */

```

```

TIMER
T_Alloc      := 15, /* Default value for ALLOC supervision timer is 15 sec, Tolerance +/- 10% */
T_AllocComp  := 15, /* Default value for ALLOC_COMP supervision timer is 15 sec, Tolerance +/- 10% */
T_BbccReset  := 30, /* Default value for BBCC_RESET supervision timer is 30 sec, Tolerance +/- 10% */
T_BbccPresync := 15, /* Default value for PRESYNC supervision timer is 15 sec, Tolerance +/- 10% */
T_Dealloc    := 30, /* Default value for DEALLOC supervision timer is 30 sec, Tolerance +/- 10% */
T_Modify     := 15, /* Default value for MODIFY supervision timer is 15 sec, Tolerance +/- 10% */
T_ModifyComp := 15, /* Default value for MODIFY_COMP supervision timer is 15 sec, Tolerance +/- 10% */
T_ModifyAbort := 15, /* Default value for MODIFY_ABORT supervision timer is 15 sec, Tolerance +/- 10% */
T_AddBranch  := 15, /* Default value for ADD_BRANCH supervision timer is 15 sec, Tolerance +/- 10% */
T_UpdateBranch := 15, /* Default value for UPDATE_BRANCH supervision timer is 15 sec, Tolerance +/- 10% */
T_DropBranch := 30, /* Default value for DROP_BRANCH supervision timer is 30 sec, Tolerance +/- 10% */

```

```

/* SN_PROT_FSM internal variables and constants*/

```

```

/**/
/* timer handling */
DCL V_ExpCount NATURAL; /*number of actual message repetitions*/
/*possible values*/
SYNONYM C_ExpMax INTEGER = 1; /*maximum number of message repetitions is 1*/
/**/
/*syntax check handling*/
DCL V_SynResult IT_SynResult; /*syntax check result, provided by procedure IE_CHECK*/

```

```

/*Signal data declarations*/

```

```

/**/
/* Transaction identifier towards AN */
DCL
V_ConnRefNo IT_ConnRefNo;
/**/
/* Transaction identifier */
DCL
V_TransId IT_TransId;
/**/
/* Branch identifier */
DCL
V_BranchId IT_BranchId;
/**/
/*Error Cause in case of reject*/
DCL
V_ProtErrCause IT_ProtErrCause;
/**/
/* Reject cause */
DCL
V_RejCause IT_RejCause;

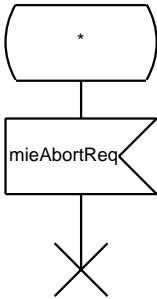
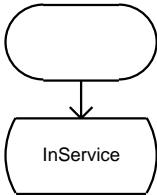
```


Process SN_PROT_FSM

2(19)

/*SN_PROT_FSM is responsible for sending/receiving BCC messages*/

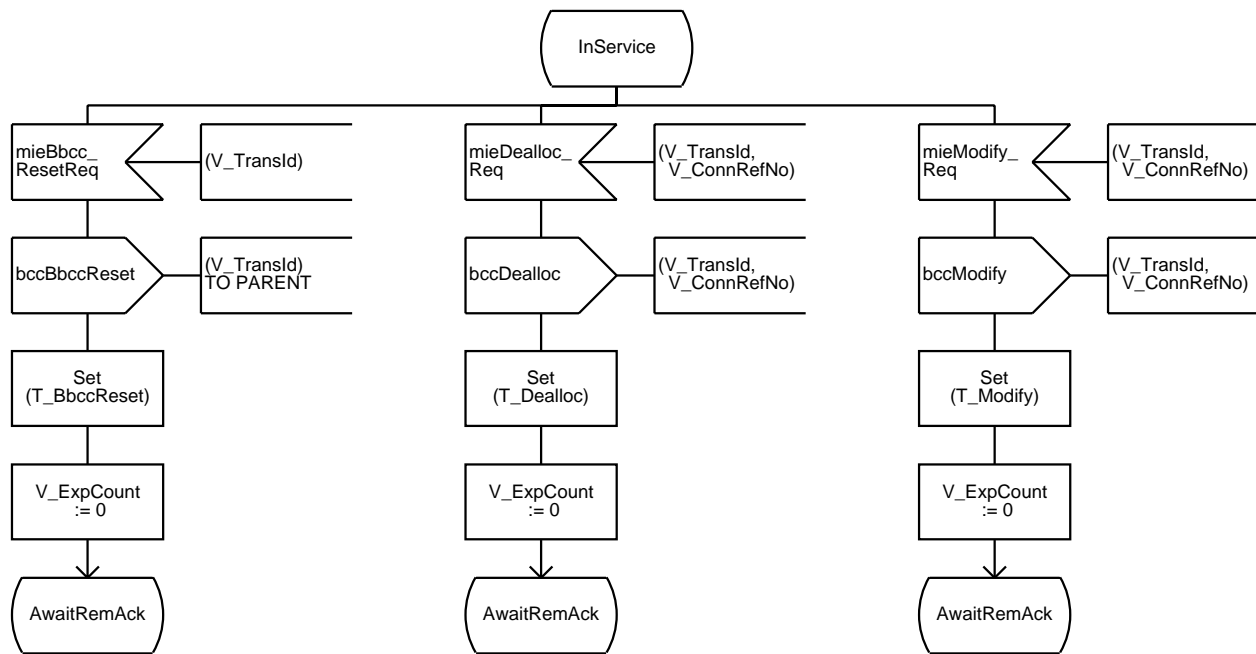
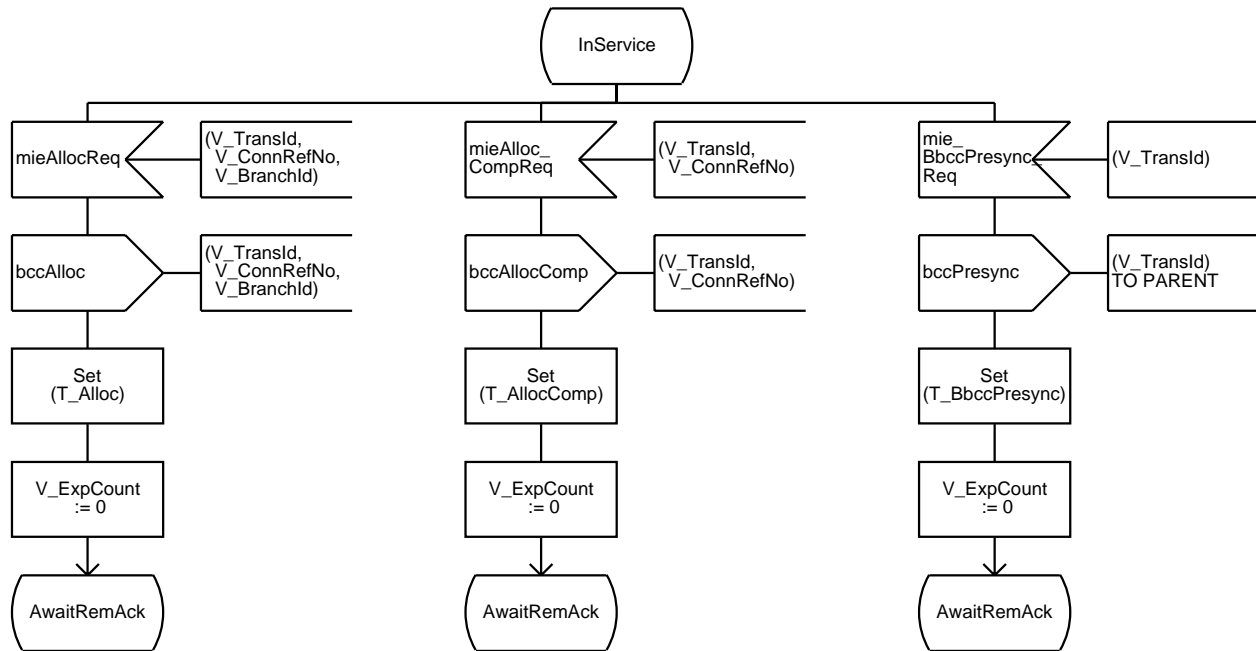
/*Handling of Requests from SN System Management */



Process SN_PROT_FSM

3(19)

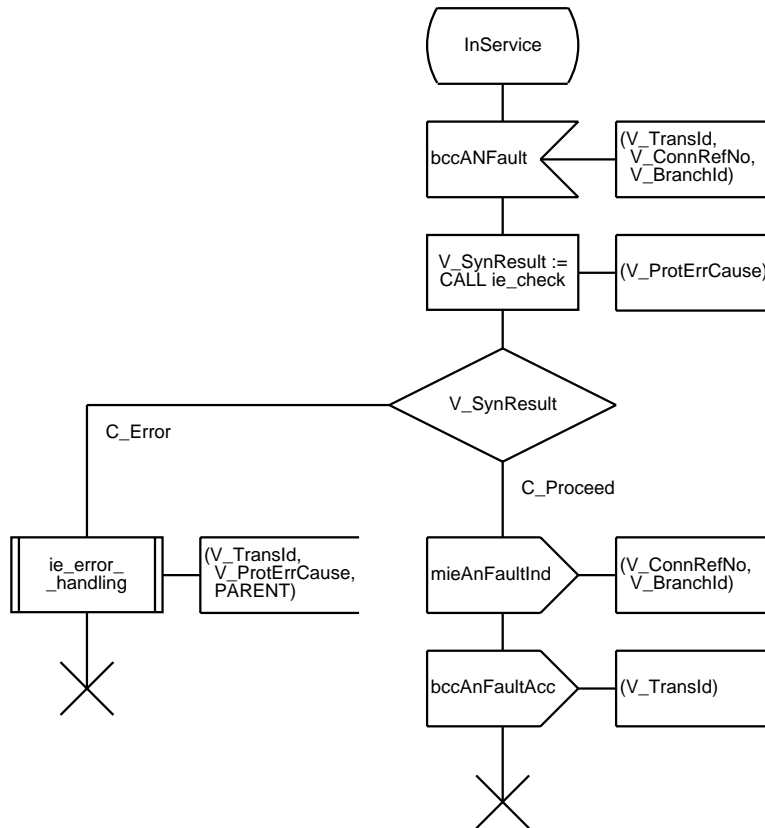
SN_PROT_FSM is responsible for sending/receiving BCC messages



Process SN_PROT_FSM

4(19)

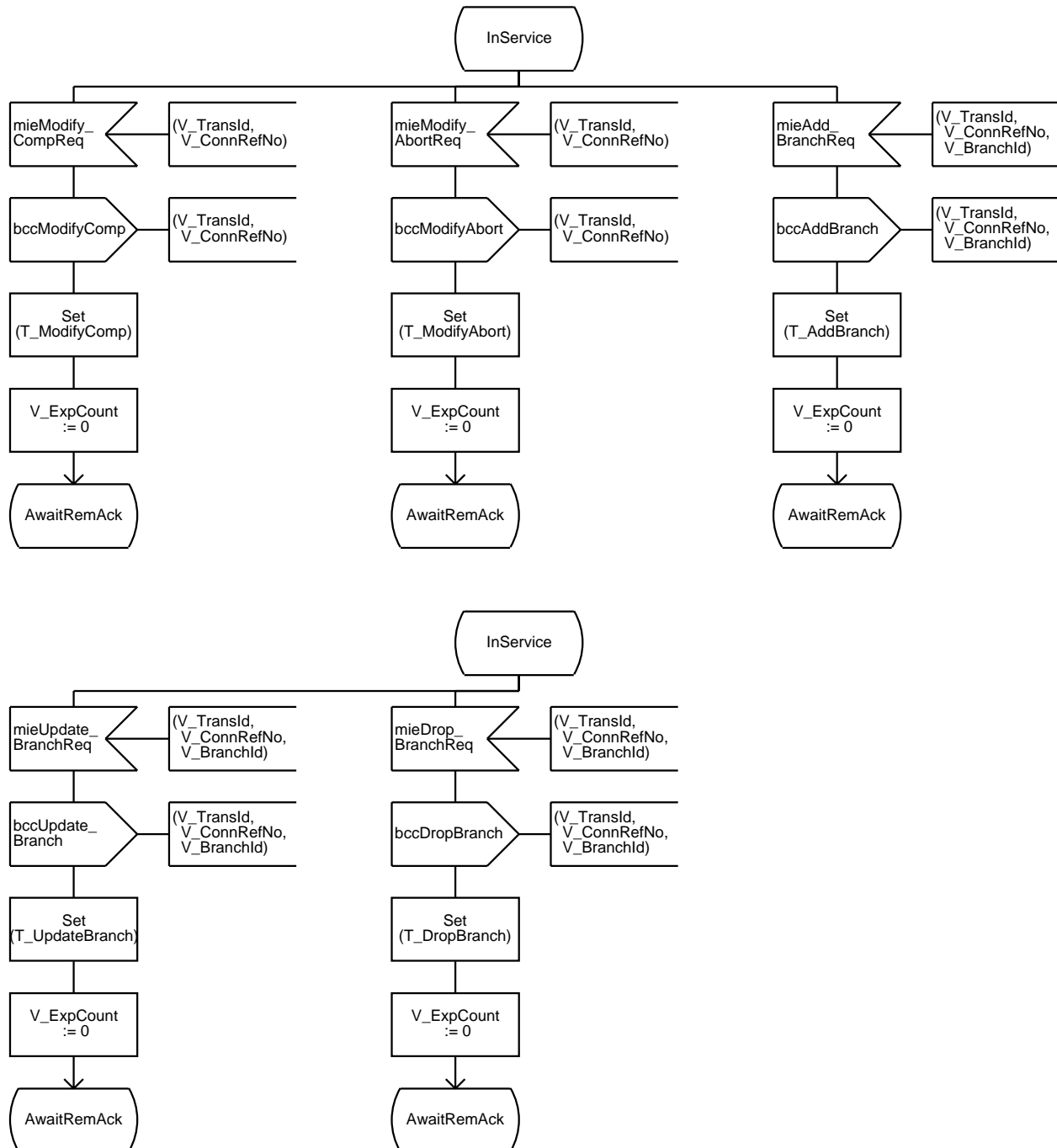
/*SN_PROT_FSM is responsible for sending/
receiving BCC messages*/



Process SN_PROT_FSM

5(19)

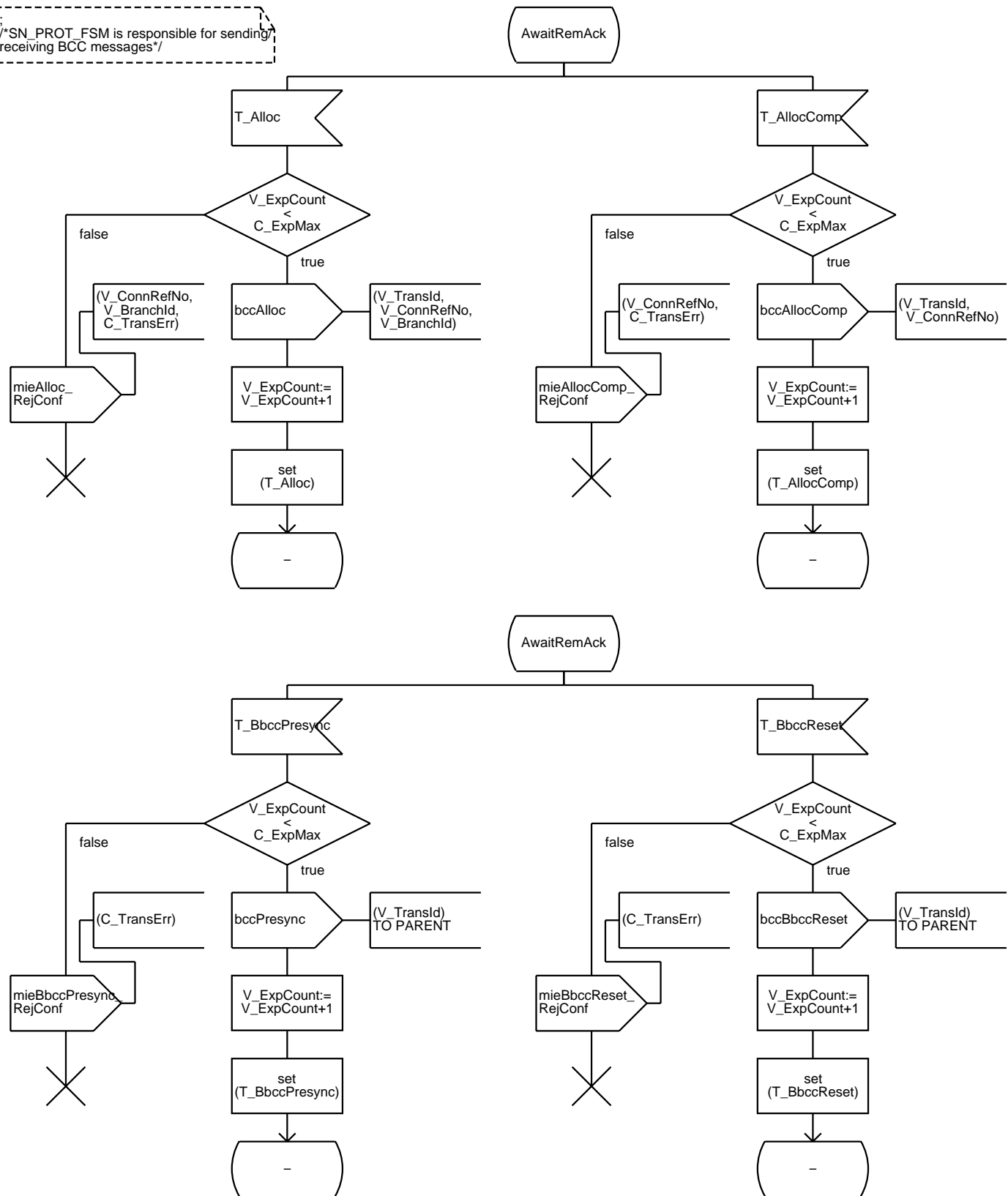
/*SN_PROT_FSM is responsible for sending/
receiving BCC messages*/



Process SN_PROT_FSM

6(19)

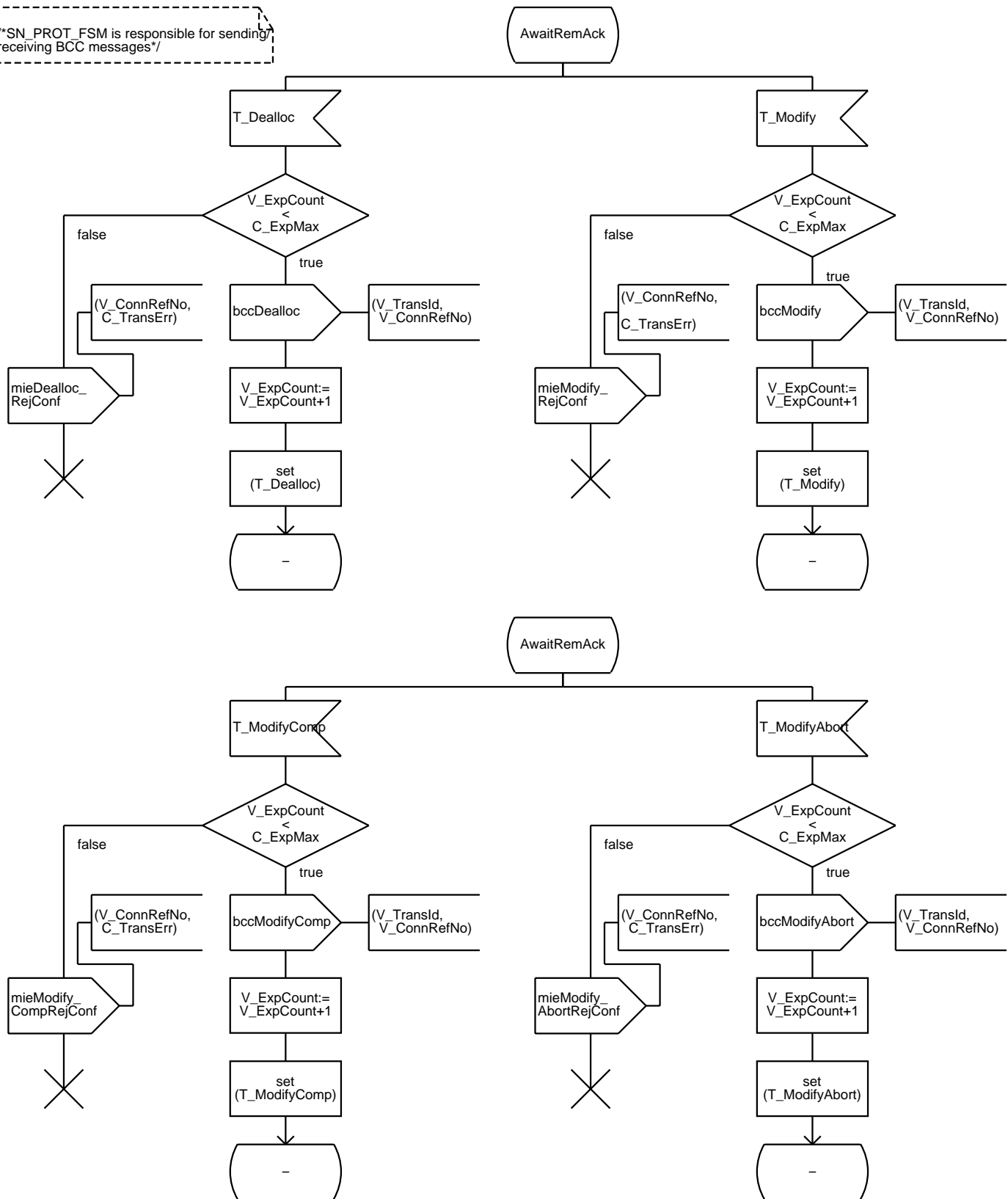
/*SN_PROT_FSM is responsible for sending/
receiving BCC messages*/



Process SN_PROT_FSM

7(19)

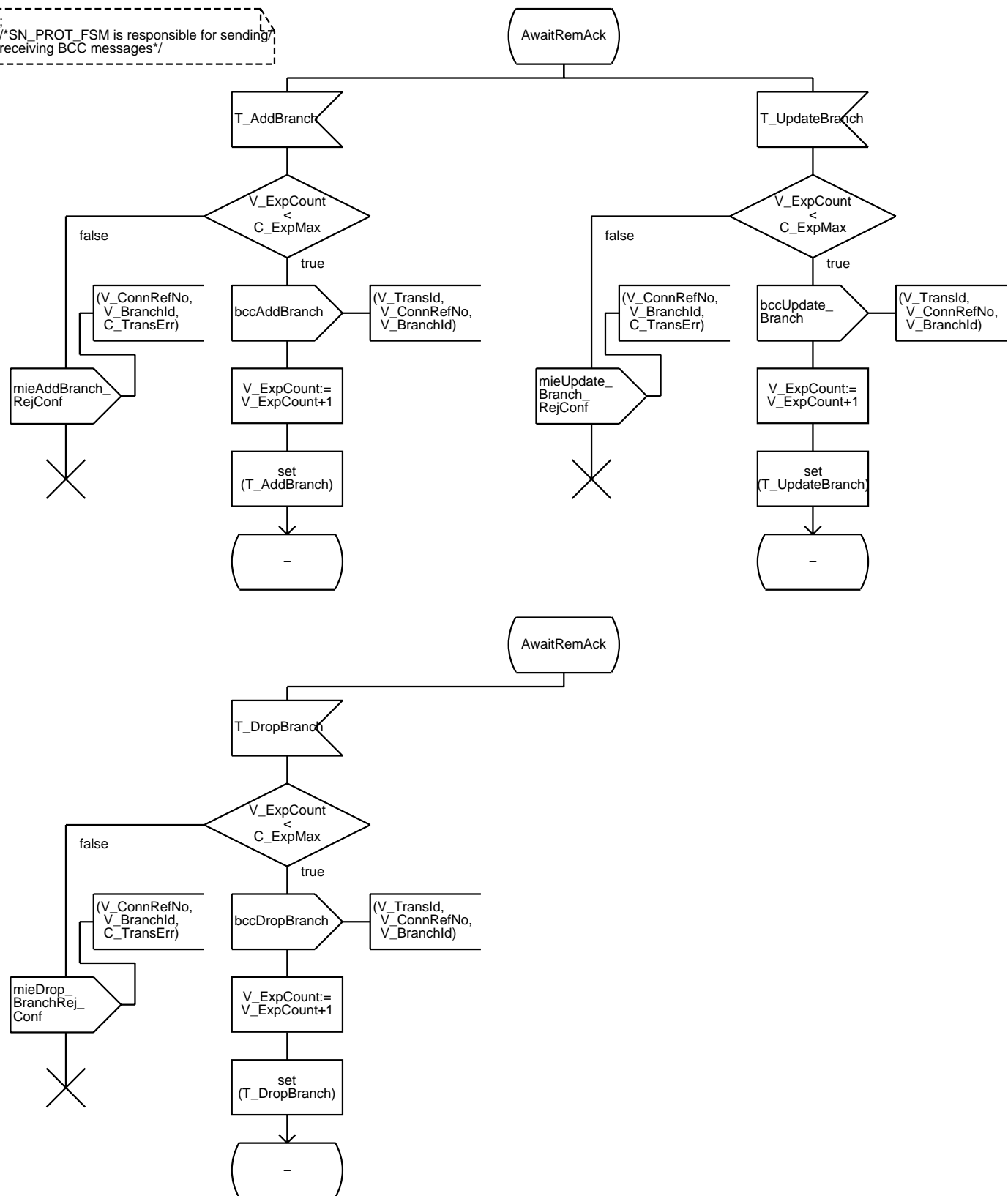
/*SN_PROT_FSM is responsible for sending/
receiving BCC messages*/



Process SN_PROT_FSM

8(19)

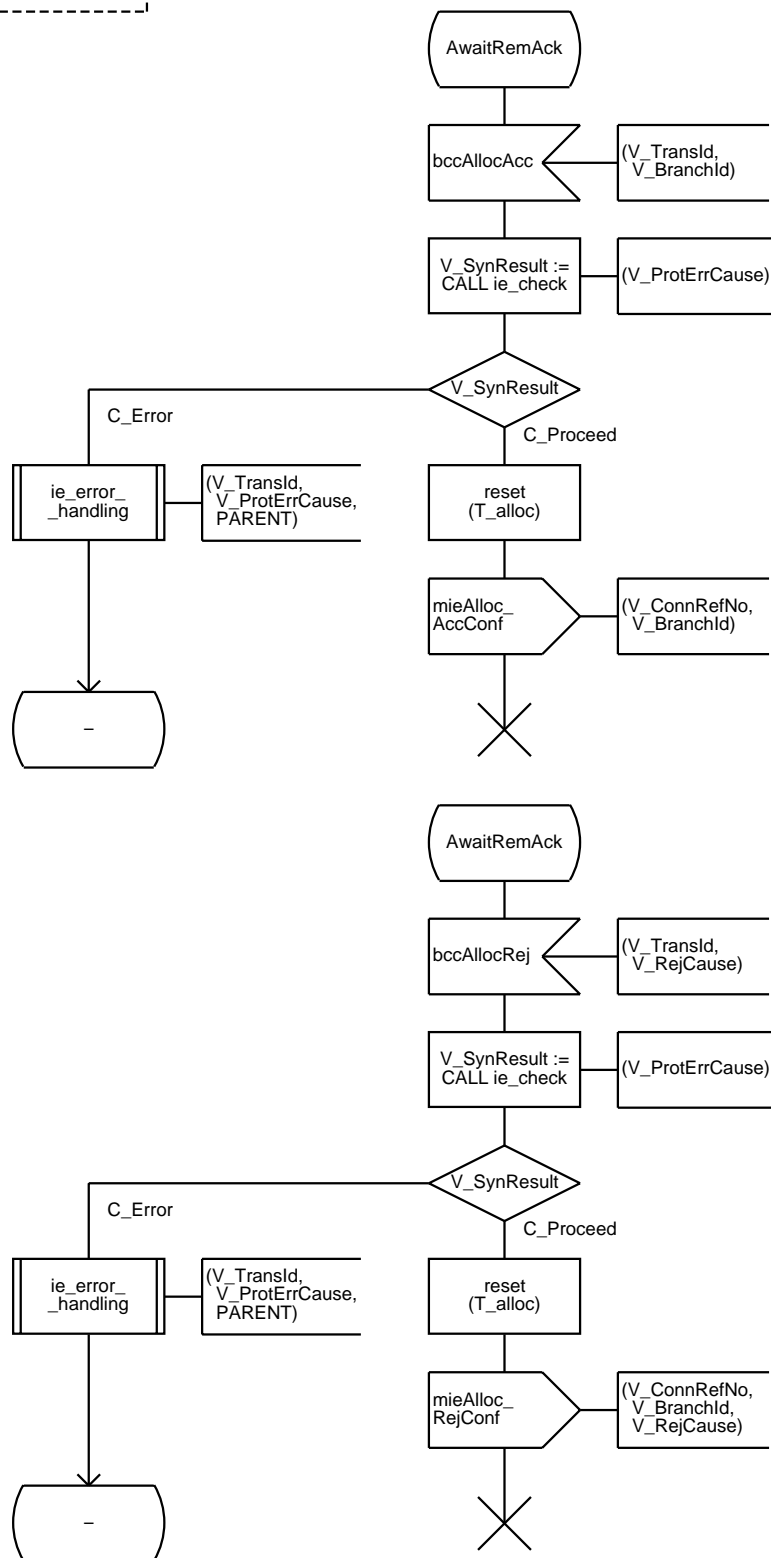
/*SN_PROT_FSM is responsible for sending/
receiving BCC messages*/



Process SN_PROT_FSM

9(19)

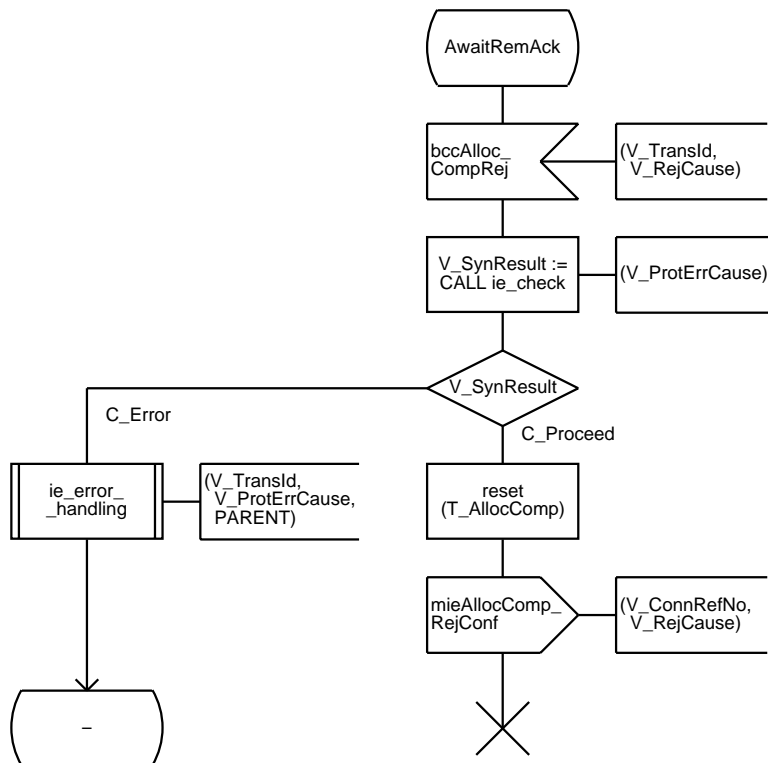
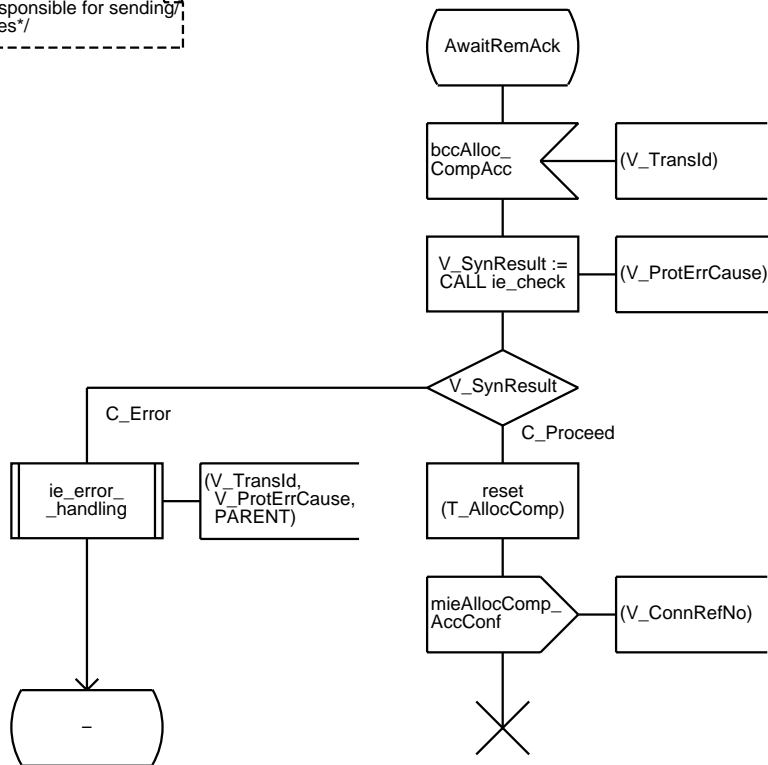
*SN_PROT_FSM is responsible for sending/
receiving BCC messages*



Process SN_PROT_FSM

10(19)

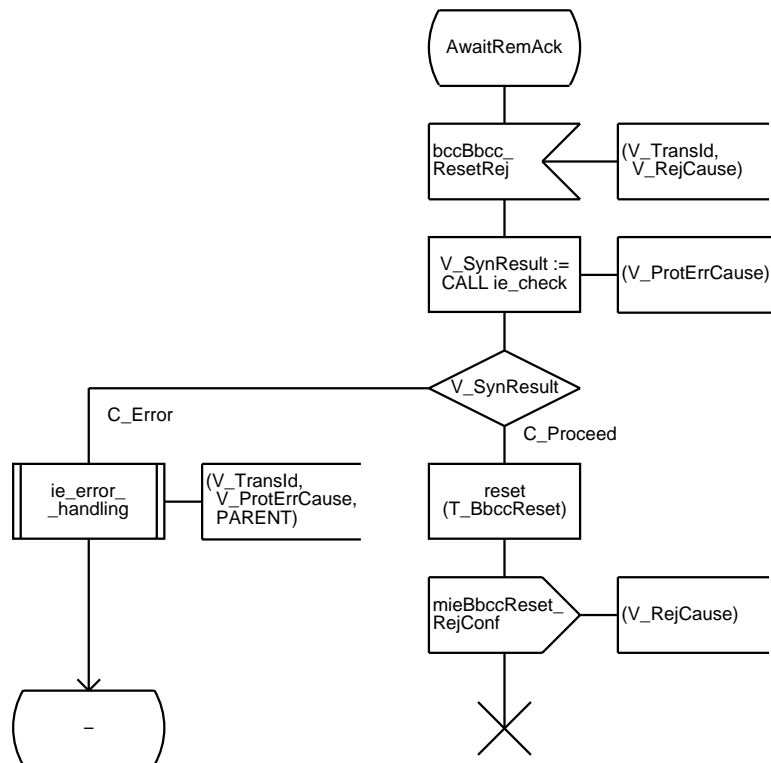
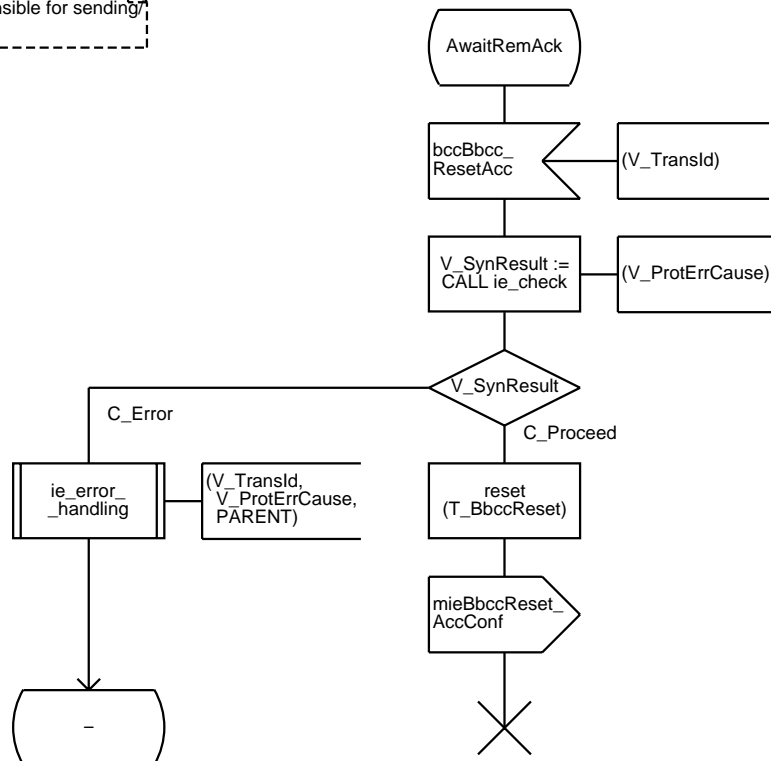
/*SN_PROT_FSM is responsible for sending/
receiving BCC messages*/



Process SN_PROT_FSM

11(19)

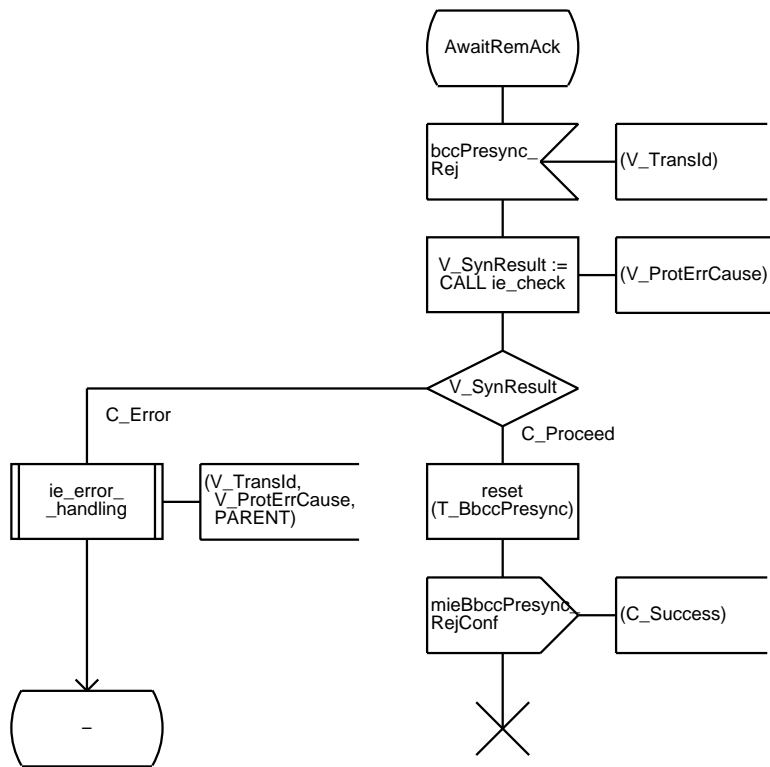
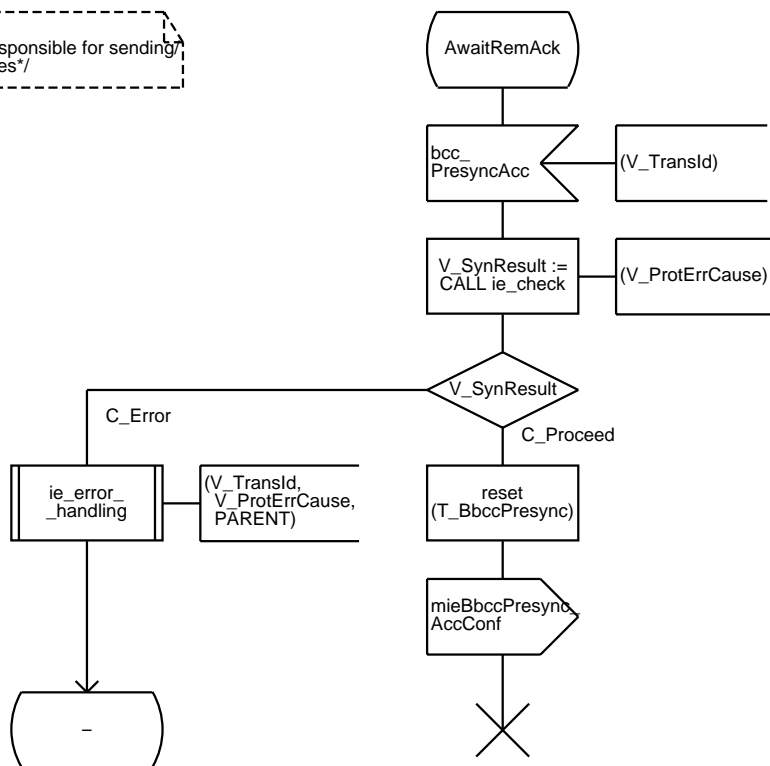
/*SN_PROT_FSM is responsible for sending/
receiving BCC messages*/



Process SN_PROT_FSM

12(19)

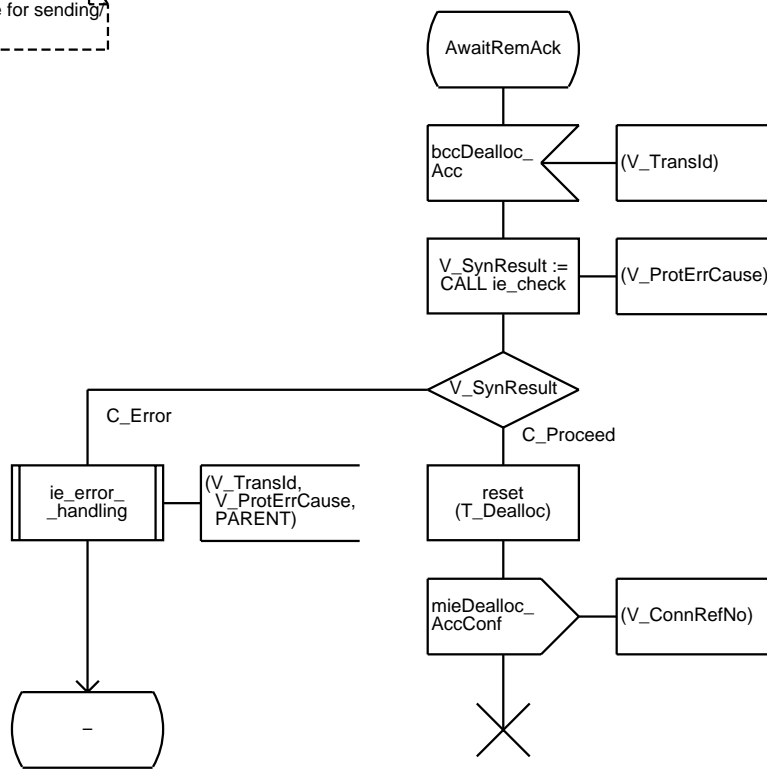
SN_PROT_FSM is responsible for sending/receiving BCC messages



Process SN_PROT_FSM

13(19)

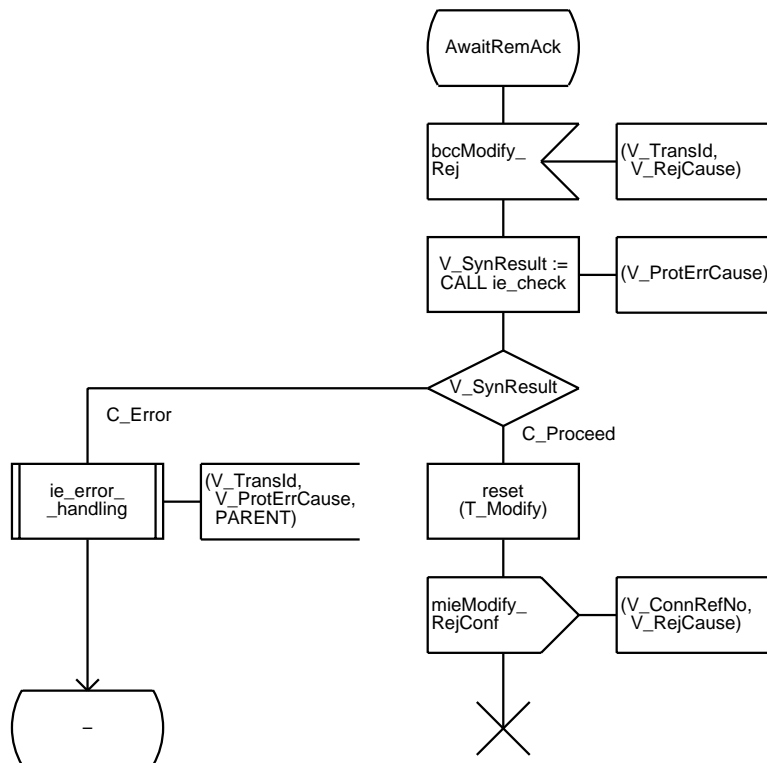
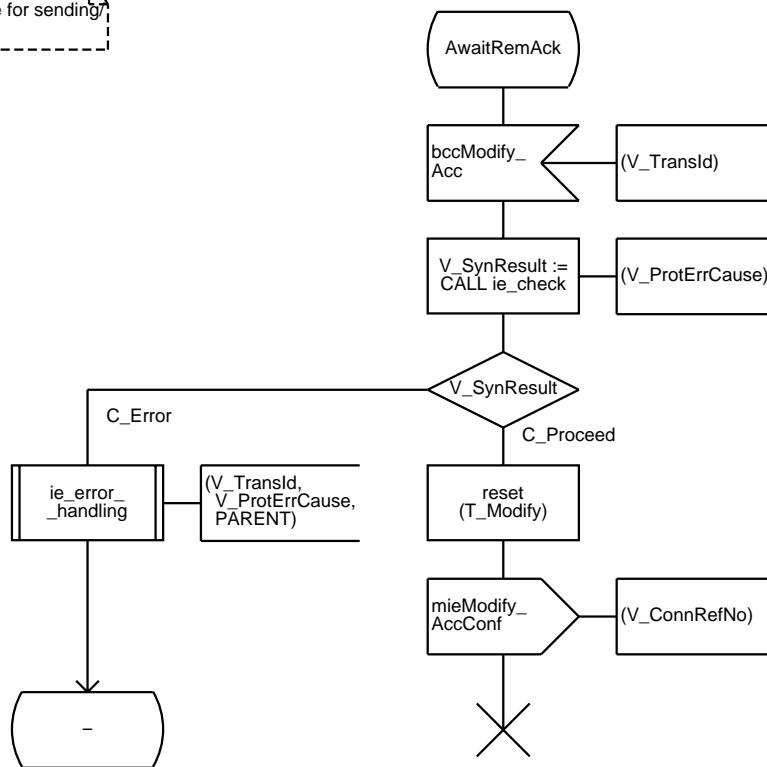
/*SN_PROT_FSM is responsible for sending/
receiving BCC messages*/



Process SN_PROT_FSM

14(19)

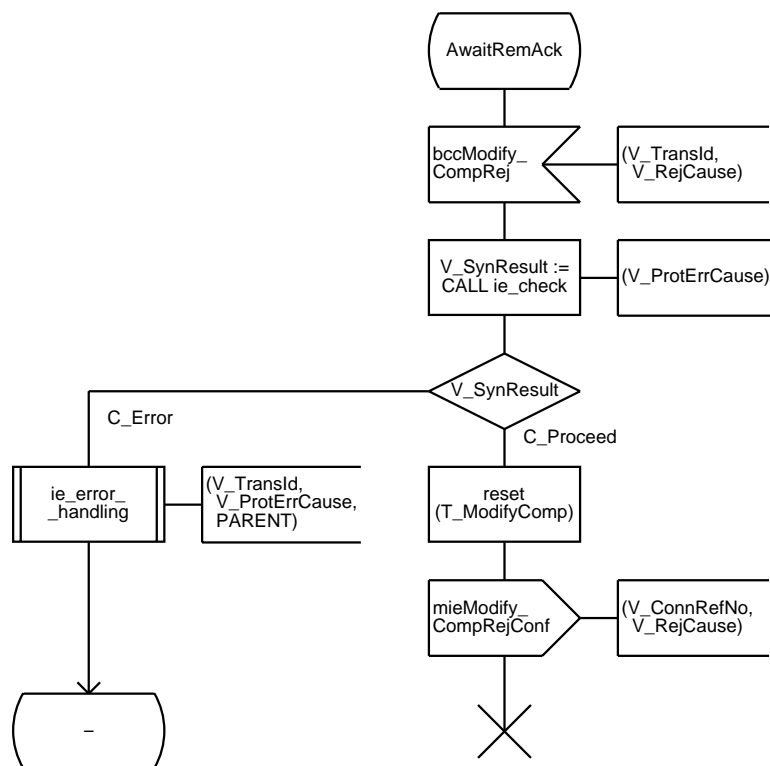
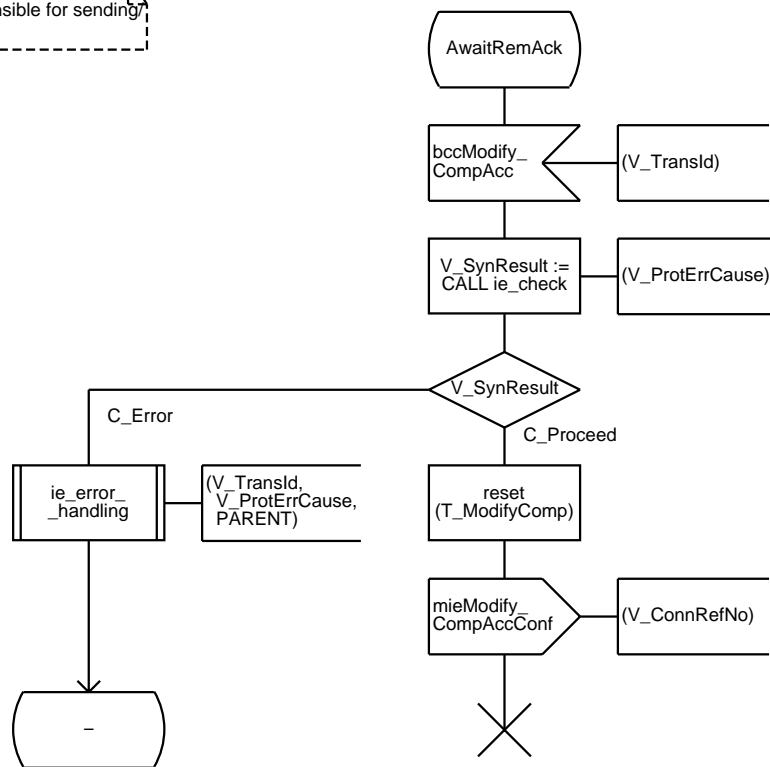
*SN_PROT_FSM is responsible for sending/
receiving BCC messages*/



Process SN_PROT_FSM

15(19)

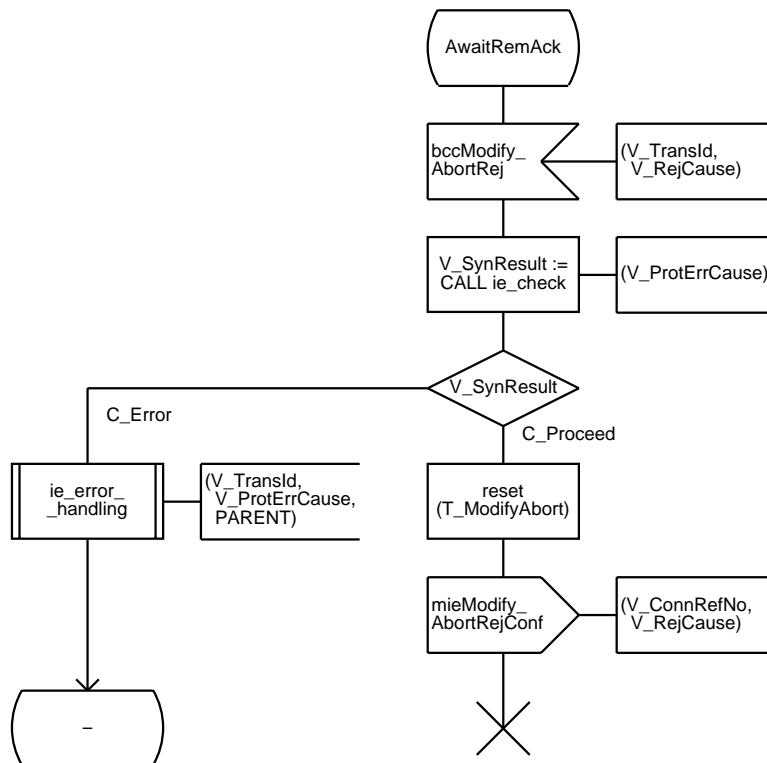
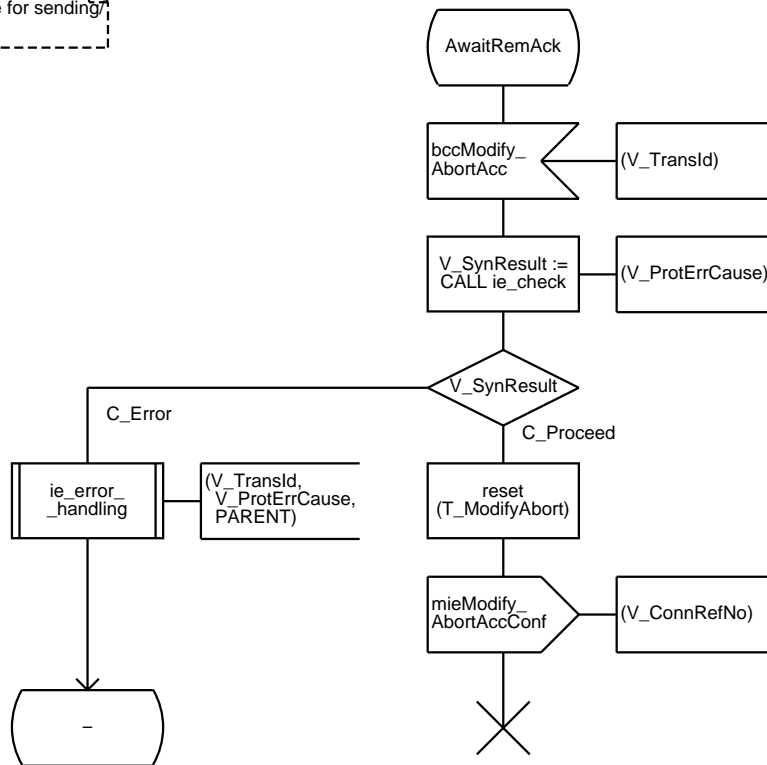
/*SN_PROT_FSM is responsible for sending/
receiving BCC messages*/



Process SN_PROT_FSM

16(19)

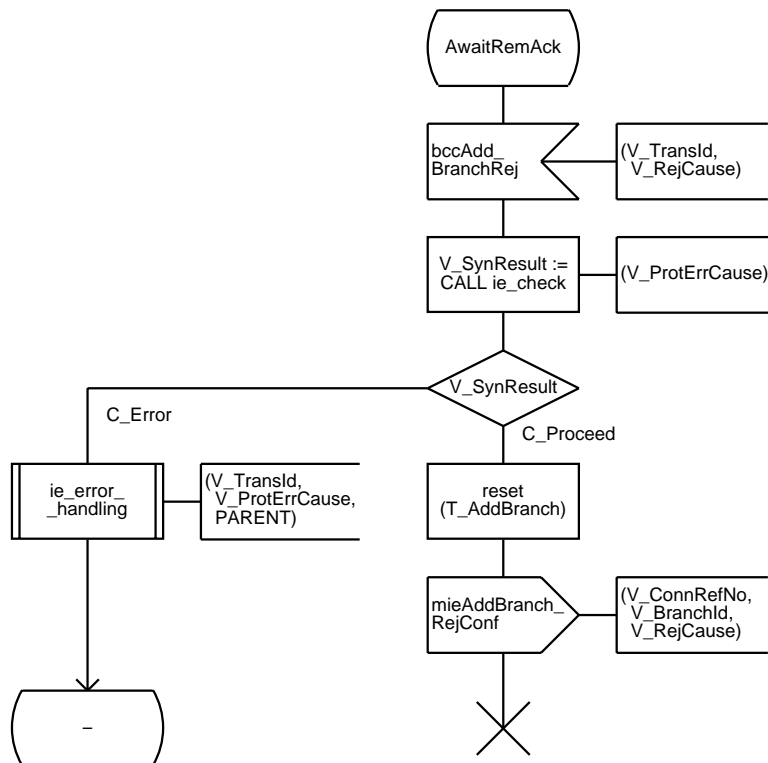
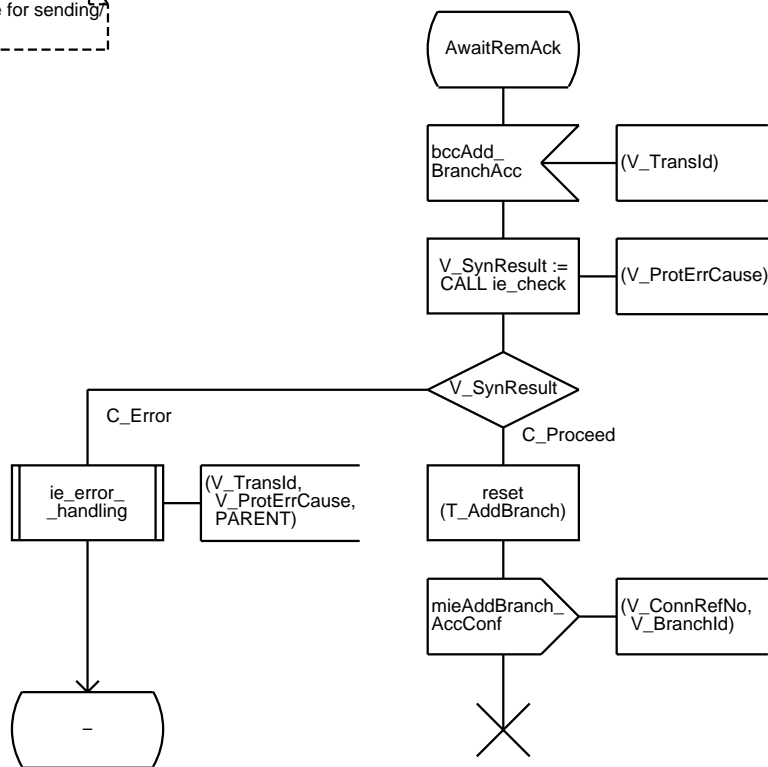
*SN_PROT_FSM is responsible for sending/
receiving BCC messages*/



Process SN_PROT_FSM

17(19)

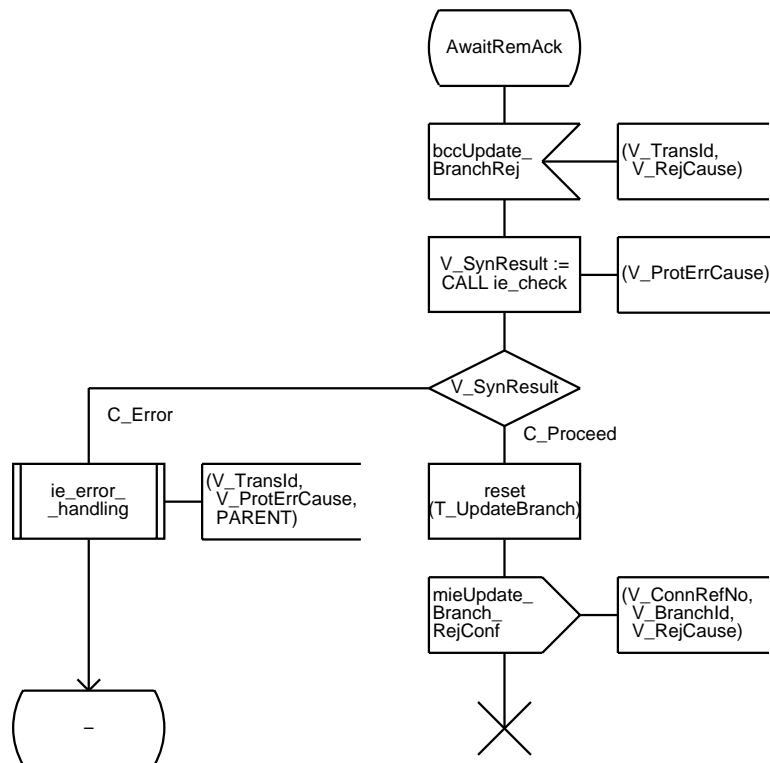
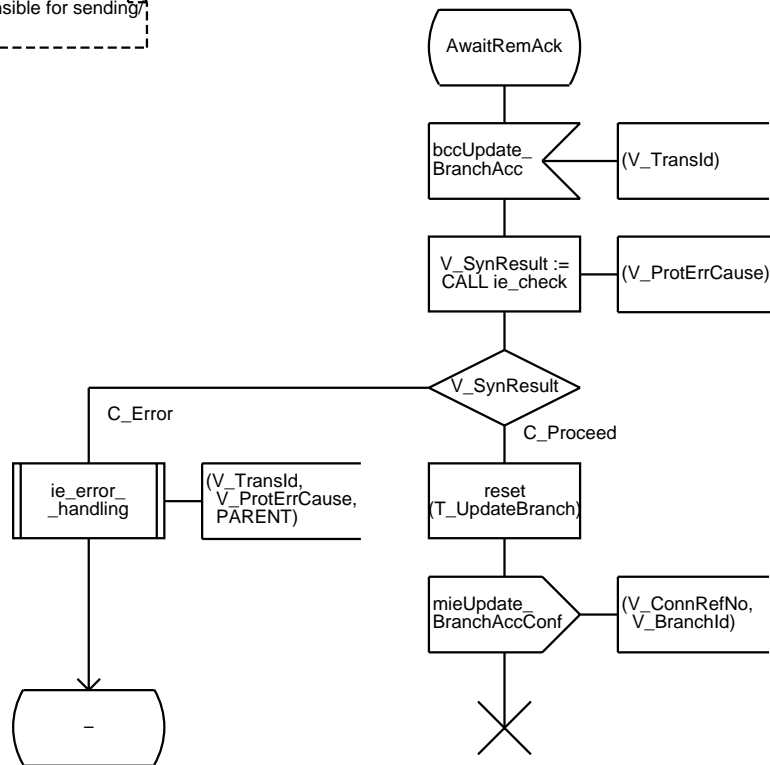
SN_PROT_FSM is responsible for sending/receiving BCC messages



Process SN_PROT_FSM

18(19)

/*SN_PROT_FSM is responsible for sending/
receiving BCC messages*/



Process SN_PROT_FSM

19(19)

/*SN_PROT_FSM is responsible for sending/
receiving BCC messages*/

