ECMA

Standard ECMA-212

3rd Edition - December 2001

Standardizing Information and Communication Systems

Private Integrated Services Network (PISN) -
Inter-Exchange Signalling Protocol -
Advice of Charge Supplementary Services

ECMA

Standardizing Information and Communication Systems

Private Integrated Services Network (PISN) -
Inter-Exchange Signalling Protocol -
Advice of Charge Supplementary Services

(QSIG-AOC)

Brief History

This Standard is one of a series of ECMA Standards defining services and signalling protocols applicable to Private Integrated Services Networks (PISNs). The series uses ISDN concepts as developed by ITU-T and conforms to the framework of International Standards for Open Systems Interconnection as defined by ISO/IEC.

This particular Standard specifies the signalling protocol for use at the Q reference point in support of the Advice Of Charge supplementary services. The protocol defined in this Standard forms part of the PSS1 protocol (informally known as QSIG).

This Standard is based upon the practical experience of ECMA member companies and the results of their active and continuous participation in the work of ISO/IEC JTC1, ITU-T, ETSI and other international and national standardization bodies. It represents a pragmatic and widely based consensus.

Compared to the 1st Edition of Standard ECMA-212 (published by ECMA in December 1994), the 2nd Edition incorporated changes in order to achieve complete alignment with International Standard ISO/IEC 15050:1997(E) published by ISO/IEC in May 1997.

Compared to the 2nd Edition of Standard ECMA-212 (published by ECMA in June 1997), this 3rd Edition incorporates migration to ASN.1 version 1997 as well as complete alignment to the ISO/IEC International Standard.

Adopted as 3rd Edition of Standard ECMA-212 by the General Assembly of December 2001.

Table of contents

1 Scope

This Standard specifies the signalling protocol for the support of the Advice Of Charge supplementary services (SS-AOC) at the Q reference point between Private Integrated services Network eXchanges (PINX) connected together within a Private Integrated Services Network (PISN).

SS-AOC is a set of supplementary services which enable a user to receive information about the charging of its calls that leave the PISN and enter another network.

The three AOC supplementary services are:

a) Charging information at call set-up time (AOC-S)

   SS-AOC-S enables the user to receive information about the charging rates at call set-up time and also to receive further information during the call if there is a change of charging rates.

b) Charging information during the call (AOC-D)

   SS-AOC-D enables a user to receive information on the recorded charges for a call during the active phase of the call.

c) Charging information at the end of the call (AOC-E)

   SS-AOC-E enables a user to receive information on the recorded charges for a call when the call is terminated.

The Q reference point is defined in ECMA-133.

Service specifications are produced in three stages and according to the method specified in ETS 300 387. This Standard contains the stage 3 specification for the Q reference point and satisfies the requirements identified by the stage 1 and stage 2 specifications in ECMA-211.

The signalling protocol for SS-AOC operates on top of the signalling protocol for basic circuit switched call control, as specified in ECMA-143, and uses certain aspects of the generic procedures for the control of supplementary services specified in ECMA-165.

This Standard also specifies additional signalling protocol requirements for the support of interactions at the Q reference point between SS-AOC and other supplementary services and ANFs.

NOTE
Additional interactions that have no impact on the signalling protocol at the Q reference point can be found in the relevant stage 1 specifications.

This Standard is applicable to PINXs which can interconnect to form a PISN.

2 Conformance

In order to conform to this Standard, a PINX shall satisfy the requirements identified in the Protocol Implementation Conformance Statement (PICS) proforma in annex A.

Conformance to this Standard includes conforming to those clauses that specify protocol interactions between SS-AOC and other supplementary services and ANFs for which signalling protocols at the Q reference point are supported in accordance with the stage 3 standards concerned.

3 References (normative)

The following standards contain provisions which, through reference in this text, constitute provisions of this Standard. All standards are subject to revision, and parties to agreements based on this Standard are encouraged to investigate the possibility of applying the most recent editions of the standards indicated below.

In the case of references to ECMA Standards that are aligned with ISO/IEC International Standards, the number of the appropriate ISO/IEC International Standard is given in brackets after the ECMA reference.

ECMA-133   Private Integrated Services Network (PISN) - Reference Configuration for PISN Exchanges (PINX) (International Standard ISO/IEC 11579-1)

ECMA-142   Private Integrated Services Network (PISN) - Circuit-mode 64 kbit/s Bearer Services - Service Description, Functional Capabilities and Information Flows (International Standard ISO/IEC 11574)

ECMA-143   Private Integrated Services Network (PISN) - Circuit-mode Bearer Services - Inter-Exchange Signalling Procedures and Protocol (International Standard ISO/IEC 11572)

ECMA-165   Private Integrated Services Network (PISN) - Generic Functional Protocol for the Support of Supplementary Services - Inter-Exchange Signalling Procedures and Protocol (International Standard ISO/IEC 11582)

ECMA-174   Private Integrated Services Network (PISN) - Inter-Exchange Signalling Protocol - Call Diversion Supplementary Services (International Standard ISO/IEC 13873)

ECMA-177   Private Integrated Services Network (PISN) - Specification, Functional Model and Information Flows - Call Transfer Supplementary Service (International Standard ISO/IEC 13865)

ECMA-178   Private Integrated Services Network (PISN) - Inter-Exchange Signalling Protocol - Call Transfer Supplementary Services (International Standard ISO/IEC 13869)

ECMA-211   Private Integrated Services Network (PISN) - Specification, Functional Model and Information Flows - Advice of Charge Supplementary Services (International Standard ISO/IEC 15049)

ECMA-221   Private Integrated Services Network (PISN) - Inter-Exchange Signalling Protocol - Call Interception Additional Network Feature (International Standard ISO/IEC 15054)

ECMA-304   Private Integrated Services Network (PISN) - Inter-Exchange Signalling Protocol - Wireless Terminal Call Handling Additional Network Features (International Standard ISO/IEC 15431)

ETS 300 387   Private Telecommunication Network (PTN); Method for the specification of basic and supplementary services (1994)

ITU-T Rec. I.112   Vocabulary of terms for ISDNs (1993)

ITU-T Rec. I.210   Principles of telecommunication services supported by an ISDN and the means to describe them (1993)

ITU-T Rec. Q.950   Supplementary services protocols, structure and general principles (2000)

ITU-T Rec. Z.100   Specification and description language (1999)

4   Definitions

For the purposes of this Standard, the following definitions apply.

4.1   External definitions

This Standard uses the following terms defined in other documents:

– Application Protocol Data Unit (APDU)   (ECMA-165)

– Basic Service   (ITU-T Rec. I.210)

– Call, Basic Call   (ECMA-165)

– Originating PINX   (ECMA-143)

– Outgoing Gateway PINX   (ECMA-143)

– Private Integrated Services Network (PISN)   (ECMA-133)

– Private Integrated services Network eXchange (PINX)   (ECMA-133)

− Signalling	(ITU-T Rec. I.112)
− Supplementary Service	(ITU-T Rec. I.210)
− Supplementary Service Control Entity	(ECMA-165)
− Terminating PINX	(ECMA-143)
− Transit PINX	(ECMA-143)
− User	(ECMA-142)
− User A	(ECMA-177)

4.2 Other definitions

4.2.1 Advice mode

The mode in which an Originating PINX receives advice of charge information from a Gateway PINX. This can be charge rate provision, interim charge provision or final charge provision.

4.2.2 Advice mode combination

A combination of one or more advice modes operating simultaneously.

4.2.3 Charge rate provision

The provision to the Originating PINX of information concerning the charge rate for the call.

4.2.4 Interim charge provision

The provision to the Originating PINX of subtotal charge information at intervals during the call and of the total charge information at the end of the call.

NOTE
When interim charge provision is used and the call is transferred, if the transferring user continues to be charged after transfer, no total charge information is provided when the call resulting from transfer finishes.

4.2.5 Final charge provision

The provision to the Originating PINX of total charge information at the end of the call.

NOTE
When final charge provision is used and the call is transferred, if the transferring user continues to be charged after transfer, the total charge information is provided to the Transferring PINX when the call resulting from transfer finishes.

4.2.6 Charging Association Information

Information that allows final charge information to be associated with the call to which it relates.

4.2.7 Accounting Function

The entity that is able to determine charges incurred in another network (e.g. by counting meter pulses, by use of a public ISDN AOC service, or by calculation) and the assignment of these charges to the PISN user(s) involved.

5 List of acronyms

ANF	Additional Network Feature
AOC	Advice of Charge
APDU	Application Protocol Data Unit
ASN.1	Abstract Syntax Notation no. 1
ISDN	Integrated Services Digital Network
NFE	Network Facility Extension
PICS	Protocol Implementation Conformance Statement

PINX   Private Integrated services Network eXchange

PISN   Private Integrated Services Network

SDL   Specification and Description Language

SS   Supplementary Service

6 Signalling protocol for the support of SS-AOC

6.1 SS-AOC description

SS-AOC is a set of supplementary services which enable a user to receive information about the charging of its calls that leave the PISN and enter another network.

The three AOC supplementary services are:

a) Charging information at call set-up time (AOC-S)

SS-AOC-S enables the user to receive information about the charging rates at call set-up time and also to receive further information during the call if there is a change of charging rates.

b) Charging information during the call (AOC-D)

SS-AOC-D enables a user to receive information on the recorded charges for a call during the active phase of the call.

c) Charging information at the end of the call (AOC-E)

SS-AOC-E enables a user to receive information on the recorded charges for a call when the call is terminated.

These three supplementary services are supported across the PISN by charge rate provision, interim charge provision and final charge provision or combinations thereof.

NOTE
For example, SS-AOC-D could be supported by charge rate provision, by interim charge provision, or by charge rate provision in conjunction with final charge provision.

6.2 SS-AOC operational requirements

6.2.1 Requirements on the Originating PINX

Call establishment procedures for the outgoing side of an inter-PINX link and call release procedures, as specified in ECMA-143 shall apply.

Generic procedures for call-related control of supplementary services, as specified in ECMA-165 for an End PINX, shall apply.

6.2.2 Requirements on the Terminating PINX

Call establishment procedures for the incoming side of an inter-PINX link and call release procedures, as specified in ECMA-143, shall apply.

Generic procedures for call-related control of supplementary services, as specified in ECMA-165 for an End PINX, shall apply.

6.2.3 Requirements on the Outgoing Gateway PINX

Call establishment procedures for the incoming side of an inter-PINX link and call release procedures, as specified in ECMA-143, shall apply.

Generic procedures for call-related control of supplementary services, as specified in ECMA-165 for an End PINX, shall apply.

Generic procedures for the call independent control (connection oriented) of supplementary services, as specified in ECMA-165 for an Outgoing Gateway PINX, shall apply.

6.2.4 Requirements on a Transit PINX

Basic call procedures specified in ECMA-143 for a Transit PINX shall apply.

Generic procedures for call-related control of supplementary services, as specified in ECMA-165 for a Transit PINX, shall apply.

6.2.5 Additional requirements for a Transferring or Diverting PINX that can receive final charge information for a call resulting from transfer or a diverted call

Generic procedures for the call independent control (connection oriented) of supplementary services, as specified in ECMA-165 for a Terminating PINX, shall apply.

6.3 SS-AOC coding requirements

6.3.1 Operations

The operations defined in Abstract Syntax Notation number 1 (ASN.1) in table 1 shall apply. The notation is in accordance with ITU-T Rec. X.680 and X.690. The ITU-T Rec. X.208 and X.209 superseded version is in annex E.

Table 1 - Operations in Support of SS-AOC

```
SS-AOC-Operations-asn1-97
{iso (1) standard (0)  pss1-advice-of-charge (15050) advice-of-charge-operations-asn1-97 (1)}


DEFINITIONS EXPLICIT TAGS ::=


BEGIN


IMPORTS              OPERATION, ERROR FROM Remote-Operations-Information-Objects
                        {joint-iso-itu-t (2) remote-operations (4) informationObjects(5) version1(0)}
                     EXTENSION, Extension{} FROM Manufacturer-specific-service-extension-class-asn1-97
                        {iso (1) standard (0) pss1-generic-procedures (11582) msi-class-asn1-97 (11)}
                     notAvailable, supplementaryServiceInteractionNotAllowed
                     FROM General-Error-List
                        {ccitt recommendation q 950 general-error-list (1)}
                     PartyNumber FROM Addressing-Data-Elements-asn1-97
                      { iso (1) standard (0) pss1-generic-procedures (11582)
                        addressing-data-elements-asn1-97 (20) }              ;


AOC-Operations OPERATION ::=      { chargeRequest | getFinalCharge | aocFinal | aocInterim | aocRate |
                                    aocComplete | aocDivChargeReq }


aocRate              OPERATION ::= {
                     ARGUMENT            AocRateArg
                     RETURN RESULT       FALSE
                     ALWAYS RESPONDS     FALSE
                     CODE                local:  63}


AocRateArg           ::=     SEQUENCE {
                                aocRate    CHOICE {
                                        chargeNotAvailable    NULL,
                                        aocSCurrencyInfoList AOCSCurrencyInfoList
                                                },
                                rateArgExtension   CHOICE {
                                        extension [1]    IMPLICIT Extension{{AOCExtSet}},
                                        multipleExtension  [2] IMPLICIT SEQUENCE OF
                                                Extension{{AOCExtSet}} } OPTIONAL
                                }
```

Table 1 - Operations in Support of SS-AOC (continued)

```
aocInterim                OPERATION ::= {
                          ARGUMENT AocInterimArg
                          RETURN RESULT      FALSE
                          ALWAYS RESPONDS    FALSE
                          CODE               local: 62}


AocInterimArg      ::=    SEQUENCE {
                              interimCharge  CHOICE {
                                  chargeNotAvailable [0] IMPLICIT    NULL,
                                  freeOfCharge       [1] IMPLICIT    NULL,
                                  specificCurrency       SEQUENCE {
                                      recordedCurrency [1] IMPLICIT RecordedCurrency,
                                      interimBillingId[2] IMPLICIT InterimBillingId OPTIONAL }
                                                  },
                              interimArgExtension   CHOICE {
                                  extension [1]    IMPLICIT Extension{{AOCExtSet}},
                                  multipleExtension  [2] IMPLICIT SEQUENCE OF
                                                      Extension{{AOCExtSet}} } OPTIONAL
                                  }

aocFinal                  OPERATION ::= {
                          ARGUMENT AocFinalArg
                          RETURN RESULT      FALSE
                          ALWAYS RESPONDS    FALSE
                          CODE               local: 61}


AocFinalArg        ::=    SEQUENCE {
                              finalCharge  CHOICE {
                                  chargeNotAvailable [0] IMPLICIT NULL,
                                  freeOfCharge [1] IMPLICIT NULL,
                                  specificCurrency    SEQUENCE {
                                      recordedCurrency [1] IMPLICIT RecordedCurrency,
                                         finalBillingId[2]  IMPLICIT FinalBillingId OPTIONAL }
                                                     },
                              chargingAssociation   ChargingAssociation        OPTIONAL,
                              finalArgExtension   CHOICE {
                                  extension [1]    IMPLICIT Extension{{AOCExtSet}},
                                  multipleExtension  [2] IMPLICIT SEQUENCE OF
                                                      Extension{{AOCExtSet}} } OPTIONAL
                                  }

AOCSCurrencyInfoList   ::= SEQUENCE SIZE(1..10) OF AOCSCurrencyInfo
```

Table 1 - Operations in Support of SS-AOC (continued)

```
AOCSCurrencyInfo        ::=    SEQUENCE {
                                chargedItem       ChargedItem,
                                rateType    CHOICE {
                                       durationCurrency          [1] IMPLICIT DurationCurrency,
                                       flatRateCurrency          [2] IMPLICIT FlatRateCurrency,
                                       volumeRateCurrency        [3] IMPLICIT VolumeRateCurrency,
                                       specialChargingCode       SpecialChargingCode,
                                       freeOfCharge              [4] IMPLICIT NULL,
                                       currencyInfoNotAvailable  [5] IMPLICIT NULL,
                                       freeOfChargefromBeginning [6] IMPLICIT NULL
                                                    }   }

ChargedItem             ::=    ENUMERATED {
                                       basicCommunication         (0),
                                       callAttempt                (1),
                                       callSetup                  (2),
                                       userToUserInfo             (3),
                                       operationOfSupplementaryServ (4) }


DurationCurrency        ::=    SEQUENCE {
                                       dCurrency     [1] IMPLICIT      Currency,
                                       dAmount       [2] IMPLICIT      Amount,
                                       dChargingType [3] IMPLICIT      ChargingType,
                                       dTime         [4] IMPLICIT      Time,
                                       dGranularity  [5] IMPLICIT      Time OPTIONAL }


FlatRateCurrency        ::=    SEQUENCE {
                                       fRCurrency  [1] IMPLICIT Currency,
                                       fRAmount    [2] IMPLICIT Amount }


VolumeRateCurrency      ::=    SEQUENCE {
                                       vRCurrency    [1] IMPLICIT Currency,
                                       vRAmount      [2] IMPLICIT Amount,
                                       vRVolumeUnit  [3] IMPLICIT VolumeUnit
                                              }

SpecialChargingCode     ::=    INTEGER (1..10)

RecordedCurrency        ::=    SEQUENCE {
                                       rCurrency   [1] IMPLICIT Currency,
                                       rAmount     [2] IMPLICIT Amount }


InterimBillingId        ::=    ENUMERATED {
                                       normalCharging       (0),
                                       creditCardCharging   (2) }
```

Table 1 - Operations in Support of SS-AOC (continued)

```
FinalBillingId          ::=   ENUMERATED {
                                normalCharging          (0),
                                creditCardCharging      (2),
                                callForwardingUnconditional (3),
                                callForwardingBusy      (4),
                                callForwardingNoReply   (5),
                                callDeflection          (6),
                                callTransfer            (7) }


Currency                ::=   IA5String (SIZE (0..10))
                              -- SIZE(0) shall indicate the default currency of the PISN
                              -- The representation of other currencies is outside the scope of this standard


Amount                  ::=   SEQUENCE {
                                currencyAmount    [1] IMPLICIT CurrencyAmount,
                                multiplier        [2] IMPLICIT Multiplier  }


CurrencyAmount          ::=   INTEGER (0..16777215)
Multiplier              ::=   ENUMERATED {
                                oneThousandth (0),
                                oneHundredth  (1),
                                oneTenth      (2),
                                one           (3),
                                ten           (4),
                                hundred       (5),
                                thousand      (6) }


Time                    ::=   SEQUENCE {
                                lengthOfTimeUnit  [1] IMPLICIT LengthOfTimeUnit,
                                scale             [2] IMPLICIT Scale }


LengthOfTimeUnit        ::=   INTEGER (0..16777215)


Scale                   ::=   ENUMERATED {
                                oneHundredthSecond (0),
                                oneTenthSecond     (1),
                                oneSecond          (2),
                                tenSeconds         (3),
                                oneMinute          (4),
                                oneHour            (5),
                                twentyFourHours    (6) }


VolumeUnit              ::=   ENUMERATED {
                                octet       (0),
                                segment     (1),
                                message     (2) }
```

Table 1 - Operations in Support of SS-AOC (continued)

```
ChargingType            ::=     ENUMERATED {
                                    continuousCharging   (0),
                                    stepFunction         (1) }


ChargingAssociation     ::=     CHOICE {
                                    chargeNumber  [0]    PartyNumber,
                                    chargeIdentifier ChargeIdentifier }


ChargeIdentifier        ::=     INTEGER (-32768..32767)


chargeRequest                   OPERATION ::= {
                                ARGUMENT        ChargeRequestArg
                                RESULT          ChargeRequestRes
                                ERRORS          {
                                    freeOfCharge |
                                    supplementaryServiceInteractionNotAllowed |
                                    notAvailable | unspecified }
                                CODE            local: 59}


getFinalCharge                  OPERATION ::= {
                                ARGUMENT        DummyArg
                                RETURN RESULT       FALSE
                                ALWAYS RESPONDS     FALSE
                                CODE                local: 60}


ChargeRequestArg        ::=     SEQUENCE {
                                adviceModeCombinations    SEQUENCE SIZE(0..7) OF
                                                AdviceModeCombination,
                                chargeReqArgExtension  CHOICE {
                                        extension [1]    IMPLICIT Extension{{AOCExtSet}},
                                        multipleExtension  [2] IMPLICIT SEQUENCE OF
                                                Extension{{AOCExtSet}} } OPTIONAL
                                    }


ChargeRequestRes        ::=     SEQUENCE {
                                adviceModeCombination  AdviceModeCombination,
                                chargeReqResExtension      CHOICE {
                                    extension [1]    IMPLICIT Extension{{AOCExtSet}},
                                    multipleExtension  [2] IMPLICIT SEQUENCE OF
                                                Extension{{AOCExtSet}} } OPTIONAL
                                    }
```

Table 1 - Operations in Support of SS-AOC (continued)

```
AdviceModeCombination ::=    ENUMERATED {        -- advice mode combination
                                 rate              (0),   -- charge rate provision
                                 rateInterim       (1),   -- charge rate and interim charge provision
                                 rateFinal         (2),   -- charge rate and final charge provision
                                 interim           (3),   -- interim charge provision
                                 final             (4),   -- final charge provision
                                 interimFinal      (5),   -- interim charge and final charge provision
                                 rateInterimFinal  (6)}   -- charge rate, interim charge and final
                                                          -- charge provision


DummyArg           ::=    CHOICE{
                                 none              NULL,
                                 extension         [1] IMPLICIT Extension{{AOCExtSet}},
                                 multipleExtension [2] IMPLICIT SEQUENCE OF Extension{{AOCExtSet}}
                                 }



-- The following OPERATION applies for the interaction with Call Transfer

aocComplete               OPERATION ::= {
                          ARGUMENT AocCompleteArg
                          RESULT    AocCompleteRes
                          ERRORS    {supplementaryServiceInteractionNotAllowed}
                          CODE      local: 64}

AocCompleteArg     ::=    SEQUENCE {
                                 chargedUser          PartyNumber,
                                 chargingAssociation  ChargingAssociation OPTIONAL,
                                 completeArgExtension    CHOICE {
                                      extension [1]    IMPLICIT Extension{{AOCExtSet}},
                                      multipleExtension [2] IMPLICIT SEQUENCE OF
                                                        Extension{{AOCExtSet}} } OPTIONAL
                                 }

AocCompleteRes::=      SEQUENCE {
                                 chargingOption ChargingOption,
                                 completeResExtension    CHOICE {
                                      extension [1]    IMPLICIT Extension{{AOCExtSet}},
                                      multipleExtension [2] IMPLICIT SEQUENCE OF Extension{{AOCExtSet}}
                                                             } OPTIONAL
                                 }

ChargingOption     ::=    ENUMERATED{
                                 aocFreeOfCharge     (0),
                                 aocContinueCharging (1),
                                 aocStopCharging     (2)
                                     }
```

Table 1 - Operations in Support of SS-AOC (concluded)

```
-- The following OPERATION applies for the interaction with Call Diversion


aocDivChargeReq           OPERATION::= {
                          ARGUMENT AocDivChargeReqArg
                          RETURN RESULT       FALSE
                          ALWAYS RESPONDS     FALSE
                          CODE                local: 65}



AocDivChargeReqArg    ::= SEQUENCE {
                          divertingUser       PartyNumber,
                          chargingAssociation ChargingAssociation      OPTIONAL,
                          diversionType       DiversionType,
                          aocDivChargeReqArgExt   CHOICE {
                              extension [1]   IMPLICIT Extension{{AOCExtSet}},
                              multipleExtension  [2] IMPLICIT SEQUENCE OF Extension{{AOCExtSet}}
                                                  } OPTIONAL
                              }



DiversionType         ::= ENUMERATED {
                              callForwardingUnconditional (0),
                              callForwardingBusy          (1),
                              callForwardingNoReply       (2),
                              callDeflection              (3) }


AOCExtSet EXTENSION ::= {...}

unspecified               ERROR ::= {
                          PARAMETER           Extension{{AOCExtSet}}
                          CODE                local: 1008}

freeOfCharge              ERROR ::= { CODE    local: 1016}

END                       -- of SS-AOC-Operations-asn1-97
```

6.3.2   Information elements
6.3.2.1   Facility information element

The operations defined in 6.3.1 shall be coded in the Facility information element in accordance with ECMA-165.

When conveying the invoke APDU of operations defined in 6.3.1, the destination Entity data element of the NFE shall contain value endPINX.

When conveying the invoke APDU of operation aocFinal, aocInterim, aocRate, or aocDivChargeReq the Interpretation APDU shall contain value discardAnyUnrecognisedInvokePdu.

When conveying the invoke APDU of operation getFinalcharge, the Interpretation APDU shall contain value clearCallIfAnyInvokePduNotRecognised.

For the invoke APDU for all the other operations, the Interpretation APDU shall be either omitted or included with the value rejectUnrecognisedInvokePDU.

6.3.2.2   Other information elements

Any other information element (e.g. Calling party number) shall be coded in accordance with ECMA-143.

6.3.3   Messages

The Facility information element shall be conveyed in the messages as specified in clause 10 of ECMA-165.

6.4   SS-AOC State definitions

6.4.1   States at the Originating PINX

The procedures for the Originating PINX are written in terms of the following conceptual states existing within the SS-AOC Supplementary Service Control entity in that PINX in association with a particular call.

6.4.1.1   State Aoc-Orig-Idle

SS-AOC is not operating.

6.4.1.2   State Aoc-Orig-Wait1-Ack

The chargeRequest invoke APDU has been sent at call establishment with a SETUP message and the Originating PINX is waiting for a response from the Terminating or the Outgoing Gateway PINX.

6.4.1.3   State Aoc-Orig-Wait2-Ack

The chargeRequest invoke APDU has been sent during the call with a FACILITY message and the Originating PINX is waiting for a response from the Terminating or the Outgoing Gateway PINX.

6.4.1.4   State Aoc-Orig-Active

SS-AOC has been invoked successfully.

6.4.1.5   State Aoc-Orig-Wait-Charge

The Originating PINX has sent the getFinalcharge invoke APDU to the Outgoing Gateway PINX in order to obtain the final charge.

6.4.1.6   State Aoc-Orig-Wait-Completion

The aocComplete invoke APDU has been sent to inform the Outgoing Gateway that call transfer is in progress and the Originating PINX is waiting for a response from the Outgoing Gateway PINX.

6.4.2   States at the Outgoing Gateway PINX

The procedures for the Outgoing Gateway PINX are written in terms of the following conceptual states existing within the SS-AOC Supplementary Service Control entity in that PINX in association with a particular call.

6.4.2.1   State Aoc-Ogw-Idle

SS-AOC is not operating.

6.4.2.2   State Aoc-Ogw-Active

SS-AOC has been invoked successfully.

6.4.2.3   State Aoc-Accounting

The Outgoing Gateway PINX waits for an acknowledge or a reject from the accounting function after having received a chargeRequest invoke APDU.

6.4.2.4   State Aoc-Transfer-Accounting

The Outgoing Gateway waits for the charging policy decision taken by the accounting function after a transfer.

6.4.2.5   State Aoc-Final-Accounting

The Outgoing Gateway waits for final charge information at the time of the transfer.

6.4.2.6   State Aoc-Release-Accounting

The Outgoing Gateway waits for final charge information.

6.5   SS-AOC Signalling procedures for activation, deactivation and registration

Not applicable.

6.6   SS-AOC Signalling procedures for invocation and operation

Examples of message sequences are shown in annex B.

6.6.1   Actions at the Originating PINX

The SDL representation of procedures at the Originating PINX is shown in C.1 of annex C.

6.6.1.1   Normal procedures

SS-AOC can be started at call set-up or during the call.

6.6.1.1.1   Invocation of SS-AOC at call set-up

When a new call is established, if there is a requirement to invoke one or more AOC services, the Originating PINX shall select the advice mode combination or combinations that can provide sufficient information to support the particular supplementary service(s) concerned, include a chargeRequest invoke APDU in the SETUP message, and enter state Aoc-Orig-Wait1-Ack. The argument of the invoke APDU shall be encoded in accordance with 6.6.1.1.4.

In state Aoc-Orig-Wait1-Ack, on receipt of a chargeRequest return result APDU in a PROGRESS, an ALERTING, a CONNECT or a FACILITY message, the Originating PINX shall enter state Aoc-Orig-Active. The result indicates the advice mode combination that is in operation.

6.6.1.1.2   Invocation of SS-AOC during the call

During a call while in state Aoc-Orig-Idle, if there is a requirement to invoke one or more AOC services, the Originating PINX shall select the advice mode combination or combinations that can provide sufficient information to support the particular supplementary service(s) concerned, send a chargeRequest invoke APDU in a FACILITY message, start timer T1 and enter state Aoc-Orig-Wait2-Ack. The argument of the invoke APDU shall be encoded in accordance with 6.6.1.1.4.

In state Aoc-Orig-Wait2-Ack, on receipt of a chargeRequest return result APDU in a FACILITY message, the Originating PINX shall stop timer T1 and shall enter state Aoc-Orig-Active. The result indicates the advice mode combination that is in operation.

During a call while in state Aoc-Orig-Active, if there is a requirement to invoke one or more additional AOC supplementary services that cannot be supported by the advice modes that are currently in operation, the Originating PINX shall select the advice mode combination or combinations that can provide sufficient information to support all the required AOC supplementary services (including those already in progress), send a chargeRequest invoke APDU in a FACILITY message, start timer T1 and remain in the same state. The argument of the invoke APDU shall be encoded in accordance with 6.6.1.1.4, except that there shall be at least one element of type AdviceModeCombination.

In state Aoc-Orig-Active, on receipt of a chargeRequest return result APDU in a FACILITY message, the Originating PINX shall stop timer T1 and stay in the same state. The result indicates the new advice modes in operation.

6.6.1.1.3   Normal operation

In state Aoc-Orig-Active, on receipt of an aocInterim or aocRate invoke APDU, the Originating PINX shall stay in state Aoc-Orig-Active.

NOTE
The charging information received should be used to provide charging information to the calling user in accordance with the particular AOC supplementary service(s) invoked.

In state AOC-Orig-Active, on receipt of a release request from the calling user, if interim charge provision or final charge provision is in operation the Originating PINX shall suspend normal basic call clearing procedures, send a getFinalCharge invoke APDU in a FACILITY message, start the timer T2 and enter state Aoc-Orig-Wait-Charge.

In state AOC-Orig-Active, on receipt of a release request from the calling user, if neither interim charge provision nor final charge provision is in operation the Originating PINX shall enter state Aoc-Orig-Idle and shall start basic call clearing procedures.

In state Aoc-Orig-Wait-Charge, on receipt of a DISCONNECT message with an aocFinal invoke APDU, the Originating PINX shall stop timer T2, enter the state Aoc-Orig-Idle and continue call clearing.

NOTE
The charging information received should be used to provide charging information to the calling user in accordance with the particular AOC supplementary service(s) invoked.

In state Aoc-Orig-Active, on receipt of an aocFinal invoke APDU in a DISCONNECT message, the Originating PINX shall enter state Aoc-Orig-Idle and continue normal call clearing.

NOTE
The charging information received should be used to provide charging information to the calling user in accordance with the particular AOC supplementary service(s) invoked.

6.6.1.1.4 Encoding of argument of chargeRequest invoke APDU

If any advice mode combination is acceptable and none is preferred, the argument of the invoke APDU shall contain zero elements of type AdviceModeCombination. If only a single advice mode combination is acceptable, the argument of the invoke APDU shall contain a single element of type AdviceModeCombination indicating the required advice mode combination. If more than one advice mode combination is acceptable, or all advice mode combinations are acceptable but with a preference, the argument of the invoke APDU shall contain multiple elements of type AdviceModeCombination indicating the acceptable advice mode combinations in order of preference, starting with the most preferred advice mode combination.

6.6.1.2 Exceptional procedures

In state Aoc-Orig-Wait1-Ack, on receipt of

– any message containing chargeRequest return error or reject APDU, or

– a CONNECT message without a chargeRequest return result, return error or reject APDU,

the Originating PINX shall enter state Aoc-Orig-Idle.

NOTE
The Originating PINX should indicate to the calling user that charging information is not available.

In state Aoc-Orig-Wait2-Ack, on receipt of a chargeRequest return error or reject APDU, the Originating PINX shall stop timer T1 and enter state Aoc-Orig-Idle.

NOTE
The Originating PINX should indicate to the calling user that charging information is not available.

In state Aoc-Orig-Wait2-Ack, on expiry of timer T1, the Originating PINX shall enter state Aoc-Orig-Idle.

NOTE
The Originating PINX should indicate to the calling user that charging information is not available.

In state Aoc-Orig-Wait-Charge, on receipt of a DISCONNECT message without any aocFinal invoke APDU, the Originating PINX shall stop timer T2, enter the state Aoc-Orig-Idle and continue the clearing of the call.

NOTE
The Originating PINX should inform the calling user that no final charge is available.

In state Aoc-Orig-Wait-Charge, on expiry of timer T2, the Originating PINX shall initiate the release of the call and enter state Aoc-Orig-Idle.

NOTE
The Originating PINX should inform the calling user that no final charge is available.

In state Aoc-Orig-Active, on receipt of a DISCONNECT message without an aocFinal invoke APDU, the Originating PINX shall stop timer T1, if running, enter state Aoc-Orig-Idle and continue normal call clearing.

NOTE
The Originating PINX should inform the calling user that no final charge is available when interim charge provision or final charge provision is in operation.

In state Aoc-Orig-Active, on receipt of a chargeRequest return error or reject APDU, the Originating PINX shall stop timer T1 and remain in the same state. The advice modes that were in operation prior to sending the chargeRequest invoke APDU shall be assumed still to be in operation.

NOTE
The Originating PINX should indicate to the calling user that additional charging information is not available.

In state Aoc-Orig-Active, on expiry of timer T1 the Originating PINX shall remain in the same state. The advice modes that were in operation prior to sending the chargeRequest invoke APDU shall be assumed still to be in operation.

NOTE
The Originating PINX should indicate to the calling user that additional charging information is not available.

6.6.2   Actions at the Outgoing Gateway PINX

The SDL representation of procedures at the Outgoing Gateway PINX is shown in C.2 of annex C.

6.6.2.1   Normal procedures

On receipt of a chargeRequest invoke APDU in a SETUP or FACILITY message, the Outgoing Gateway PINX shall inform the accounting function and enter state Aoc-Accounting.

In state Aoc-Accounting, if the accounting function indicates that it is able to accept one of the advice mode combinations indicated in the chargeRequest invoke APDU, or if there were no advice mode combinations indicated in the argument of the chargeRequest invoke APDU and the accounting function indicates an advice mode combination that it is able to provide, the Outgoing Gateway PINX shall send a chargeRequest return result APDU and enter state Aoc-Ogw-Active. The APDU may be sent in an ALERTING, PROGRESS or CONNECT message, if one of these messages is to be sent at the same time. Otherwise it shall be sent in a FACILITY message. The APDU shall contain the selected advice mode combination in the result and the Outgoing Gateway PINX shall regard these advice modes as being in operation.

In state Aoc-Ogw-Active, on receipt of a chargeRequest invoke APDU in a FACILITY message, the Outgoing Gateway PINX shall inform the accounting function and remain in the same state.

In state Aoc-Ogw-Active, if the accounting function indicates that it is able to accept one of the advice mode combinations indicated in the chargeRequest invoke APDU, the Outgoing Gateway PINX shall send a chargeRequest return result APDU in a FACILITY message. The APDU shall contain the selected advice mode combination in the result and the Outgoing Gateway PINX shall regard these advice modes and no other advice modes as being in operation.

If charge rate provision is in operation, an aocRate invoke APDU shall be sent:

−   with the chargeRequest return result APDU;

−   in a FACILITY message whenever a change of the rate is detected.

If interim charge provision is in operation, an aocInterim invoke APDU shall be sent periodically during the call in a FACILITY message. The rate at which it is sent is an implementation option.

In state Aoc-Ogw-Active, on receipt of a getFinalcharge invoke APDU in a FACILITY message when interim charge provision or final charge provision is in operation, the Outgoing Gateway PINX shall initiate the release of the call to the other network and enter state Aoc-Release-Accounting.

In state Aoc-Release-Accounting, on receipt of the final charge from the accounting function, the Outgoing Gateway PINX shall send an aocFinal invoke APDU in a DISCONNECT message to the Originating PINX and enter state Aoc-Ogw-Idle.

In state Aoc-Ogw-Active, on receipt of final charge information from the accounting function, the Outgoing Gateway PINX, shall send an aocFinal invoke APDU to the Originating PINX in a DISCONNECT message and enter state Aoc-Ogw-Idle.

In state Aoc-Ogw-Active, on receipt of a basic call release message from the Originating PINX, the Outgoing Gateway PINX shall enter state Aoc-Ogw-Idle and continue the normal call clearing as described in ECMA-143.

6.6.2.2   Exceptional procedures

In state Aoc-Accounting, if the request for charge information is rejected by the accounting function, the Outgoing Gateway PINX shall send a chargeRequest return error APDU with error value "notAvailable" or "freeOfCharge" and enter state Aoc-Ogw-Idle.

In state Aoc-Ogw-Active, on receipt of a getFinalcharge invoke APDU in a FACILITY message when neither interim nor final charge provision is in operation, the Outgoing Gateway PINX shall clear the call in both directions and enter state Aoc-Ogw-Idle.

In state Aoc-Ogw-Active, if the accounting function indicates that it is unable to accept any of the advice mode combinations indicated in the chargeRequest invoke APDU, the Outgoing Gateway PINX shall send a chargeRequest return error APDU containing error value "notAvailable" in a FACILITY message and remain in the same state. The advice modes that were in operation prior to receiving the chargeRequest invoke APDU shall be assumed still to be in operation.

6.6.2.3   Additional procedures for Call Transfer
6.6.2.3.1   Normal procedures

In state Aoc-Ogw-Active, on receipt of an aocComplete invoke APDU in a FACILITY message, the Outgoing Gateway PINX shall inform the accounting function that a call transfer is requested and shall enter state Aoc-Transfer-Accounting.

In state Aoc-Transfer-Accounting, if the accounting function decides to charge user A for the call before and after the transfer, the Outgoing Gateway PINX shall send an aocComplete return result APDU in a FACILITY message with the value aocContinueCharging to the Originating PINX and;

– when final charge provision is not in operation, the Outgoing Gateway PINX shall enter state Aoc-Ogw-Idle;

– when final charge provision is in operation, the Outgoing Gateway PINX shall enter state Aoc-Final-Accounting.

In state Aoc-Transfer-Accounting, if the accounting function decides not to charge user A for the call before or after the transfer, the Outgoing Gateway PINX shall send an aocComplete return result APDU in a FACILITY message with the value aocFreeOfCharge to the Originating PINX and enter state Aoc-Ogw-Idle.

In state Aoc-Transfer-Accounting, if the accounting function decides not to charge user A for the call after the transfer, the Outgoing Gateway PINX shall send an aocComplete return result APDU in a FACILITY message with the value aocStopCharging to the Originating PINX and,

– when interim charge or final charge provision is in operation, the Outgoing Gateway PINX shall enter state Aoc-Final-Accounting;

– when neither interim charge provision nor final charge provision is in operation, the Outgoing Gateway PINX shall enter state Aoc-Ogw-Idle.

In state Aoc-Final-Accounting, on receipt of the final charge information from the accounting function, the Outgoing Gateway shall send the final charge with an aocFinal invoke APDU in a SETUP message using the call reference of a call-independent signalling connection. The call-independent signalling connection shall be established in accordance with the procedures specified in 7.3 of ECMA-165. Then the Outgoing Gateway PINX shall enter the state Aoc-Ogw-Idle.

If element chargingAssociation was present in the aocComplete invoke APDU, the same information shall be included in element chargingAssociation in the aocFinal invoke APDU.

The element finalBillingId of the aocFinal invoke APDU shall be coded with the value "callTransfer".

The number to be used in the Called party number information element of the call-independent signalling connection to the Transferring PINX shall be obtained from the element chargedUser in the received aocComplete invoke APDU.

NOTE
If user A continues to be charged for the call resulting from transfer, final charge information will not be available until the call finishes.

6.6.2.3.2 Exceptional procedures

In state Aoc-Final-Accounting, on receipt of a chargeRequest invoke APDU, the Outgoing Gateway PINX shall reject it by sending a chargeRequest return error APDU with the value "notAvailable" and stay in the same state.

6.6.2.4 Additional procedures for Call Diversion

NOTE
These procedures apply only when the diverting user is charged for the diverted call.

6.6.2.4.1 Normal procedures

In state Aoc-Ogw-Idle, on receipt of an aocDivChargeReq invoke APDU, the Outgoing Gateway PINX shall inform the accounting function about the request and enter state Aoc-Final-Accounting.

In state Aoc-Final-Accounting, on receipt of the final charge information from the accounting function, the Outgoing Gateway shall send the final charge with an aocFinal invoke APDU in a SETUP message using the call reference of a call-independent signalling connection. The call-independent signalling connection shall be established in accordance with the procedures specified in 7.3 of ECMA-165. Then the Outgoing Gateway PINX shall enter the state Aoc-Ogw-Idle.

If element chargingAssociation was present in the aocDivChargeReq invoke APDU, the same information shall be included in element chargingAssociation in the aocFinal invoke APDU.

The element finalBillingId of the aocFinal invoke APDU sent back to the diverting user shall be coded with one of the following values according to the type of diversion: callForwardingUnconditional, callForwardingBusy, callForwardingNoReply or callDeflection.

The routeing of the call-independent signalling connection to the Diverting PINX is based on the element divertingUser received in the aocDivChargeReq invoke APDU.

6.6.2.4.2 Exceptional procedures

In state Aoc-Final-Accounting, on receipt of a chargeRequest invoke APDU, the Outgoing Gateway PINX shall reject it by sending a chargeRequest return error APDU with the value "notAvailable" and stay in the same state.

6.6.3 Actions at a Transit PINX

Not applicable.

6.6.4 Actions at the Terminating PINX

The SDL representation of procedures at the Terminating PINX is shown in C.3 of annex C.

6.6.4.1 Normal procedures

Not applicable.

6.6.4.2 Exceptional procedures

When sending a CONNECT message, if the SETUP message contained a chargeRequest invoke APDU, the Terminating PINX shall reject it by including a chargeRequest return error APDU containing error "freeOfCharge".

On receipt of a chargeRequest invoke APDU in a FACILITY message, the Terminating PINX shall reject it by sending a chargeRequest return error APDU containing error "freeOfCharge" in a FACILITY message.

6.6.4.3 Additional procedures for Call Diversion

6.6.4.3.1 Normal procedures

On receipt of an aocDivChargeReq invoke APDU in a FACILITY message, the Terminating PINX shall inform the accounting function about the request and enter state Aoc-Final-Accounting.

NOTE
In this situation, the accounting function will always indicate that the call is free of charge and therefore the aocFinal invoke APDU will be coded freeOfCharge.

6.7 SS-AOC Impact of interworking with public ISDNs

Not applicable.

NOTE
The accounting entity at the Outgoing Gateway PINX can use a service of the public ISDN network to obtain charging information.

6.8 SS-AOC Impact of interworking with non-ISDNs

Not applicable.

NOTE
The accounting entity at the Outgoing Gateway PINX can use a service of the public non-ISDN network to obtain charging information.

6.9 Protocol interactions between SS-AOC and other supplementary services and ANFs

This clause specifies protocol interactions with other supplementary services and ANFs for which stage 3 standards had been published at the time of publication of this Standard. For interactions with supplementary services and ANFs for which stage 3 standards are published subsequent to the publication of this Standard, see those other stage 3 standards.

NOTE
Simultaneous conveyance of APDUs for SS-AOC and another supplementary service or ANF in the same message, each in accordance with the requirements of its respective stage 3 Standard, does not, on its own, constitute a protocol interaction.

6.9.1 Interaction with Calling Name Identification Presentation (SS-CNIP)

No interaction.

6.9.2 Interaction with Connected Name Identification Presentation (SS-CONP)

No interaction.

6.9.3 Interaction with Call Transfer (SS-CT)

The following interaction shall apply if SS-CT is supported in accordance with ECMA-178.

6.9.3.1 Actions at an SS-AOC Originating PINX

The SDL representation of procedures at the Originating PINX is shown in C.4 of annex C.

6.9.3.1.1 Normal procedures

In state Aoc-Orig-Active, on a Call Transfer request, the Originating PINX shall delay procedures of ECMA-178, send an aocComplete invoke APDU in a FACILITY message to the Outgoing Gateway PINX, start timer T1 and enter state Aoc-Orig-Wait-Completion.

The number of user A shall be included in element chargedUser.

If charging association information is available, it shall be included in element chargingAssociation.

In state Aoc-Orig-Wait-Completion, on receipt of an aocComplete return result APDU in a FACILITY message coded aocFreeOfCharge, i.e. user A is not charged for the call before or after the transfer, the Originating PINX shall stop timer T1, start the Call Transfer procedures as described in ECMA-178 and enter state Aoc-Orig-Idle.

NOTE
The Originating PINX should send to user A the indication that the call is free of charge in accordance with the particular AOC supplementary services invoked.

In state Aoc-Orig-Wait-Completion, on receipt of an aocComplete return result APDU in a FACILITY message coded aocContinueCharging, i.e. user A continues to be charged, the Originating PINX shall stop timer T1, start the Call Transfer procedures as described in ECMA-178 and enter state Aoc-Orig-Idle.

NOTE
In the case of AOC-D, the Originating PINX should send the subtotal charge to user A.

In state Aoc-Orig-Wait-Completion, on receipt of an aocComplete return result APDU in a FACILITY message coded aocStopCharging, i.e. user A is charged only for the call prior to transfer, the Originating PINX shall stop timer T1, start the Call Transfer procedures described in ECMA-178 and enter state Aoc-Orig-Idle.

In state Aoc-Orig-Idle, on receipt of an aocFinal invoke APDU in a SETUP message using the call reference of a call-independent signalling connection, the Originating PINX shall stay in state Aoc-Orig-Idle and clear the call-independent signalling connection.

NOTE
The final charging information received should be indicated to the user as identified by the content of the Called party number information element in the SETUP message. If element chargingAssociation is present in the invoke APDU, it should also be indicated to the user.

6.9.3.1.2   Exceptional procedures

In state Aoc-Orig-Wait-Completion, on expiry of timer T1, the Originating PINX shall start the Call Transfer procedures as described in ECMA-178 and enter state Aoc-Orig-Idle.

In state Aoc-Orig-Wait-Completion, on receipt of an aocComplete return error or reject APDU the Originating PINX shall stop timer T1, start the Call Transfer procedures as described in ECMA-178 and enter state Aoc-Orig-Idle.

In state Aoc-Orig-Wait-Completion, on receipt of a call clearing message relating to one of the calls to be transferred, the Originating PINX shall allow the clearing of that call to continue and shall also initiate the clearing of the other call to be transferred.

6.9.3.2   Actions at the Primary PINX

In state Aoc-Orig-Idle, after receipt of a callTransferComplete invoke APDU (transfer by join) or a callTransferSetup return result APDU (transfer by rerouting), the Primary PINX may act as an Originating PINX for SS-AOC in accordance with the procedures of 6.6.1, using the procedures of 6.6.1.1.2 to invoke SS-AOC.

6.9.3.3   Actions at the Secondary PINX

In state Aoc-Orig-Idle, after receipt of a callTransferComplete invoke APDU (transfer by join) or a callTransferSetup return result APDU (transfer by rerouting), the Secondary PINX may act as an Originating PINX for SS-AOC in accordance with the procedures of 6.6.1, using the procedures of 6.6.1.1.2 to invoke SS-AOC.

6.9.4   Interaction with Call Diversion (SS-DIV)

The following interaction shall apply if SS-DIV is supported in accordance with ECMA-174.

6.9.4.1   Actions at an SS-DIV Served User PINX

The SDL representation of procedures at the Served user PINX is shown in C.5 of annex C.

When sending a callRerouting invoke APDU to the Rerouting PINX in accordance with ECMA-174, if final charge provision is required the Served User PINX shall include an aocDivChargeReq invoke

APDU in the same FACILITY message. Element divertingUser shall contain the number of the diverting user.

The diversionType element of the aocDivChargeReq invoke APDU shall contain the Diversion service invoked at the Served User PINX.

In state Aoc-Orig-Idle, on receipt of an aocFinal invoke APDU in a SETUP message using the call reference of a call-independent signalling connection established in accordance with 7.3 of ECMA-165, the Served User PINX shall stay in state Aoc-Orig-Idle and clear the call-independent signalling connection towards the Outgoing Gateway PINX.

NOTE
The final charging information received should be indicated to the user as identified by the content of the called party number information element in the SETUP message. If element chargingAssociation is present in the invoke APDU, this should also be indicated to the user.

6.9.4.2 Actions at an SS-DIV Rerouting PINX

On reception of an aocDivChargeReq invoke APDU together with the callRerouting invoke APDU, the Rerouting PINX shall send the aocDivChargeReq invoke APDU in the SETUP message used to send the divertingLegInformation2 invoke APDU to the Diverted-to PINX.

When executing Call Forwarding, the Rerouting PINX shall include a chargeRequest invoke APDU in the SETUP message to the Diverted-to PINX if this was included in the SETUP message to the Diverting PINX.

6.9.5 Interaction with Completion of Call on Busy Subscriber (SS-CCBS)

No interaction.

6.9.6 Interaction with Completion of Call on No Reply (SS-CCNR)

No interaction.

6.9.7 Interaction with Call Offer (SS-CO)

No interaction.

6.9.8 Interaction with Call Intrusion (SS-CI)

No interaction.

6.9.9 Interaction with Do Not Disturb (SS-DND)

No interaction.

6.9.10 Interaction with Do Not Disturb Override (SS-DNDO)

No interaction.

6.9.11 Interaction with Path Replacement (ANF-PR)

No interaction.

6.9.12 Interaction with Recall (SS-RE)

No interaction.

6.9.13 Interaction with Call Interception (ANF-CINT)

The following interaction shall apply if ANF-CINT is supported in accordance with ECMA-221.

6.9.13.1 Actions at an ANF-CINT Intercepting PINX

When executing ANF-CINT, the Intercepting PINX shall include a chargeRequest invoke APDU in the SETUP message to the Intercepted-to PINX if this was included in the SETUP message to the Intercepting PINX.

6.9.14 Interaction with Wireless Terminal Location Registration (SS-WTLR)

No interaction.

6.9.15 Interaction with Wireless Terminal Mobility Incoming Call (ANF-WTMI)

The following interaction shall apply if ANF-WTMI is supported in accordance with ECMA-304.

6.9.15.1 Actions at an ANF-WTMI Rerouting PINX

When executing ANF-WTMI, the Rerouting PINX shall include a chargeRequest invoke APDU in the SETUP message to the Visitor PINX if this was included in the SETUP message to the WTMI-Detect PINX.

6.10 SS-AOC Parameter values (timers)

Timer T1

Timer T1 shall operate at the Originating PINX during state Aoc-Orig-Wait2-Ack, Aoc-Orig-Active or Aoc-Orig-Wait-Completion. Its purpose is to protect against an absence of response to chargeRequest invoke APDU sent during the call.

Timer T1 shall have a value not less than 15 seconds.

Timer T2

Timer T2 shall operate at the Originating PINX during state Aoc-Orig-Wait-Charge. Its purpose is to protect against an absence of response to getFinalcharge invoke APDU.

Timer T2 shall have a value not less than 15 seconds.

Annex A
(normative)

Protocol Implementation Conformance Statement (PICS) proforma

A.1 Introduction

The supplier of a protocol implementation which is claimed to conform to this Standard shall complete the following Protocol Implementation Conformance Statement (PICS) proforma.

A completed PICS proforma is the PICS for the implementation in question. The PICS is a statement of which capabilities and options of the protocol have been implemented. The PICS can have a number of uses, including use:

– by the protocol implementor, as a check list to reduce the risk of failure to conform to the Standard through oversight;

– by the supplier and acquirer, or potential acquirer, of the implementation, as a detailed indication of the capabilities of the implementation, stated relative to the common basis for understanding provided by the Standard's PICS proforma;

– by the user or potential user of the implementation, as a basis for initially checking the possibility of interworking with another implementation - while interworking can never be guaranteed, failure to interwork can often be predicted from incompatible PICSs;

– by a protocol tester, as the basis for selecting appropriate tests against which to assess the claim for conformance of the implementation.

A.2 Instructions for completing the PICS proforma

A.2.1 General structure of the PICS proforma

The PICS proforma is a fixed format questionnaire divided into sub-clauses each containing a group of individual items. Each item is identified by an item number, the name of the item (question to be answered), and the reference(s) to the clause(s) that specifies (specify) the item in the main body of this Standard.

The "Status" column indicates whether an item is applicable and if so whether support is mandatory or optional. The following terms are used:

m	mandatory (the capability is required for conformance to the protocol);
o	optional (the capability is not required for conformance to the protocol, but if the capability is implemented it is required to conform to the protocol specifications);
o.<n>	optional, but support of at least one of the group of options labelled by the same numeral <n> is required;
x	prohibited;
<c.cond>	conditional requirement, depending on support for the item or items listed in condition <cond>;
<item>:m	simple conditional requirement, the capability being mandatory if item number <item> is supported, otherwise not applicable;
<item>:o	simple conditional requirement, the capability being optional if item number <item> is supported, otherwise not applicable.

Answers to the questionnaire items are to be provided either in the "Support" column, by simply marking an answer to indicate a restricted choice (Yes or No), or in the "Not Applicable" column (N/A).

A.2.2 Additional information

Items of Additional Information allow a supplier to provide further information intended to assist the interpretation of the PICS. It is not intended or expected that a large quantity will be supplied, and a PICS can be considered complete without any such information. Examples might be an outline of the ways in which a (single) implementation can be set up to operate in a variety of environments and configurations.

References to items of Additional Information may be entered next to any answer in the questionnaire, and may be included in items of Exception information.

A.2.3 Exception information

It may occasionally happen that a supplier will wish to answer an item with mandatory or prohibited status (after any conditions have been applied) in a way that conflicts with the indicated requirement. No pre-printed answer will be found in the Support column for this. Instead, the supplier is required to write into the support column an x.<i> reference to an item of Exception Information, and to provide the appropriate rationale in the Exception item itself.

An implementation for which an Exception item is required in this way does not conform to this Standard. A possible reason for the situation described above is that a defect in the Standard has been reported, a correction for which is expected to change the requirement not met by the implementation.

A.3   PICS proforma for ECMA-212

A.3.1   Implementation identification

Supplier	
Contact point for queries about the PICS	
Implementation name(s) and version(s)	
Other information necessary for full identification, e.g., name(s) and version(s) for machines and/or operating systems; system name(s)	

Only the first three items are required for all implementations; other information may be completed as appropriate in meeting the requirement for full identification.

The terms Name and Version should be interpreted appropriately to correspond with a suppliers terminology (e.g., Type, Series, Model).

A.3.2   Protocol summary

Protocol version	1.0
Addenda Implemented (if applicable)	
Amendments Implemented	
Have any exception items been required (see A.2.3)	No [ ] Yes [ ] (The answer Yes means that the implementation does not conform to this Standard)

Date of Statement	

A.3.3 General

Item	Question/feature	References	Status	N/A	Support
A1	Support of SS-AOC in Originating PINX?		o.1		Yes [ ] No[ ]
A2	Support of SS-AOC in Terminating PINX?		o.1		Yes [ ] No[ ]
A3	Support of SS-AOC in Outgoing Gateway PINX?		o.1		Yes [ ] No[ ]
A4	Support of charge rate provision		o.2		Yes [ ] No[ ]
A5	Support of interim charge provision		o.2		Yes [ ] No[ ]
A6	Support of final charge provision		o.2		Yes [ ] No[ ]

A.3.4 Procedures

Item	Question/feature	References	Status	N/A	Support
B1	Support of relevant ECMA-143 and ECMA-165 procedures at the Originating PINX	6.2.1	A1:m	[ ]	m:Yes [ ]
B2	Support of relevant ECMA-143 and ECMA-165 procedures at the Terminating PINX	6.2.2	A2:m	[ ]	m:Yes [ ]
B3	Support of relevant ECMA-143 and ECMA-165 procedures at the Outgoing Gateway PINX	6.2.3	A3:m	[ ]	m:Yes [ ]
B4	Procedures at the Originating PINX	6.6.1.1.1, 6.6.1.1.3, 6.6.1.1.4, 6.6.1.2	A1:m	[ ]	m:Yes [ ]
B5	Additional procedures at the Originating PINX for invocation during the call	6.6.1.1.2	A1:o	[ ]	o:Yes [ ] No [ ]
B6	Procedures at the Terminating PINX	6.6.4	A2:m	[ ]	m:Yes [ ]
B7	Procedures at the Outgoing Gateway PINX	6.6.2	A3:m	[ ]	m:Yes [ ]

A.3.5 Coding

Item	Question/feature	References	Status	N/A	Support
C1	Sending of chargeRequest invoke APDU and receipt of chargeRequest return result and error APDU	6.3.1	A1:m	[ ]	m:Yes [ ]
C2	Receipt of chargeRequest invoke APDU and sending of chargeRequest return result and error APDU	6.3.1	c.1	[ ]	m:Yes [ ]
C3	Sending of getFinalcharge invoke APDU	6.3.1	c.2	[ ]	m:Yes [ ]
C4	Sending of aocFinal invoke APDU	6.3.1	c.3	[ ]	m:Yes [ ]
C5	Sending of aocInterim invoke APDU	6.3.1	c.4	[ ]	m:Yes [ ]
C6	Sending of aocRate invoke APDU	6.3.1	c.5	[ ]	m:Yes [ ]
C7	Receipt of getFinalcharge invoke APDU	6.3.1	c.6	[ ]	m:Yes [ ]
C8	Receipt of aocFinal invoke APDU	6.3.1	c.7	[ ]	m:Yes [ ]
C9	Receipt of aocInterim invoke APDU	6.3.1	c.8	[ ]	m:Yes [ ]
C10	Receipt of aocRate invoke APDU	6.3.1	c.9	[ ]	m:Yes [ ]
C11	Receipt of aocComplete invoke APDU	6.3.1	A3:m	[ ]	m:Yes [ ]
C12	Receipt of aocDivChargeReq invoke APDU	6.3.1	c.3	[ ]	m:Yes [ ]

c.1   if A2 or A3 then mandatory, else N/A

c.2   if (A1 and (A5 or A6)) then mandatory, else N/A

c.3   if (A3 and (A5 or A6)) then mandatory, else N/A

c.4   if (A3 and A5) then mandatory, else N/A

c.5   if (A3 and A4) then mandatory, else N/A

c.6   if (A3 and (A5 or A6)) then mandatory, else N/A

c.7   if (A1 and (A5 or A6)) then mandatory, else N/A

c.8   if (A1 and A5) then mandatory, else N/A

c.9   if (A1 and A4) then mandatory, else N/A

A.3.6 Timers

Item	Question/feature	References	Status	N/A	Support
D1	Support of timer T1	6.10	A1:m	[ ]	m:Yes [ ]
D2	Support of timer T2	6.10	A1:m	[ ]	m:Yes [ ]

A.3.7 Interactions between SS-AOC and SS-CT

Item	Question/feature	References	Status	N/A	Support
E1	Support of SS-CT		o		Yes [ ] No [ ]
E2	Actions at an SS-AOC Originating PINX	6.9.3.1	c.11	[ ]	m:Yes [ ]
E3	Actions at a Primary PINX	6.9.3.2	c.12	[ ]	o:Yes [ ] No [ ]
E4	Actions at a Secondary PINX	6.9.3.3	c.12	[ ]	o:Yes [ ] No [ ]
E5	Sending of aocComplete invoke APDU	6.3.1	E2:m	[ ]	m:Yes [ ]

c.11   if (E1 and A1) then mandatory, else N/A

c.12   if (E1 and A1) then optional, else N/A

A.3.8 Interactions between SS-AOC and SS-DIV

Item	Question/feature	References	Status	N/A	Support
F1	Support of SS-DIV		o		Yes [ ] No [ ]
F2	Able to act as an SS-DIV Rerouting PINX		F1:o	[ ]	o:Yes [ ] No [ ]
F3	Actions at an SS-DIV Served User PINX	6.9.4.1	c.13	[ ]	o:Yes [ ] No [ ]
F4	Actions at an SS-DIV Rerouting PINX	6.9.4.2	F2:m	[ ]	m:Yes [ ]
F5	Sending of aocDivChargeReq invoke APDU	6.3.1	F3:m	[ ]	m:Yes [ ]

c.13   if (F1 and A1) then optional, else N/A

A.3.9 Interactions between SS-AOC and ANF-CINT

Item	Question/feature	References	Status	N/A	Support
G1	Support of ANF-CINT		o		Yes [ ] No [ ]
G2	Interaction at an ANF-CINT Intercepting PINX	6.9.13.1	G1:m	[ ]	m:Yes [ ]

A.3.10 Interactions between SS-AOC and ANF-WTMI

Item	Question/feature	References	Status	N/A	Support
H1	Support of ANF-WTMI		o		Yes [ ] No [ ]
H2	Able to act as an ANF-WTMI Rerouting PINX		H1:o	[ ]	o:Yes [ ] No [ ]
H3	Actions at an ANF-WTMI Rerouting PINX	6.9.15.1	H2:m	[ ]	m:Yes [ ]

Annex B

Examples of message sequences

This annex describes some typical message flows for SS-AOC. The following conventions are used in the figures of this annex.

1. The following notation is used:

    ➤ Basic call message containing SS-AOC information

    → Basic call message without SS-AOC information

2. The figures show messages exchanged via Protocol Control between PINXs involved in SS-AOC. Only messages relevant to SS-AOC are shown.

3. Only the relevant information content (e.g., remote operation APDUs, notifications, information elements) is listed below each message name. The Facility and Notification indicator information elements containing remote operation APDUs and notifications are not explicitly shown. Information with no impact on SS-AOC is not shown.

4. Some interactions with users are included in the form of symbolic primitives. The actual protocol at the terminal interface is outside the scope of this Standard.

B.1 Example message sequence for normal operation of SS-AOC with request during call set-up

Figure B.1 shows an example of normal operation of SS-AOC when the invocation of the service is done at call set-up. Two adviceModeCombinations, "rate" and "final", are requested and "final" is accepted.
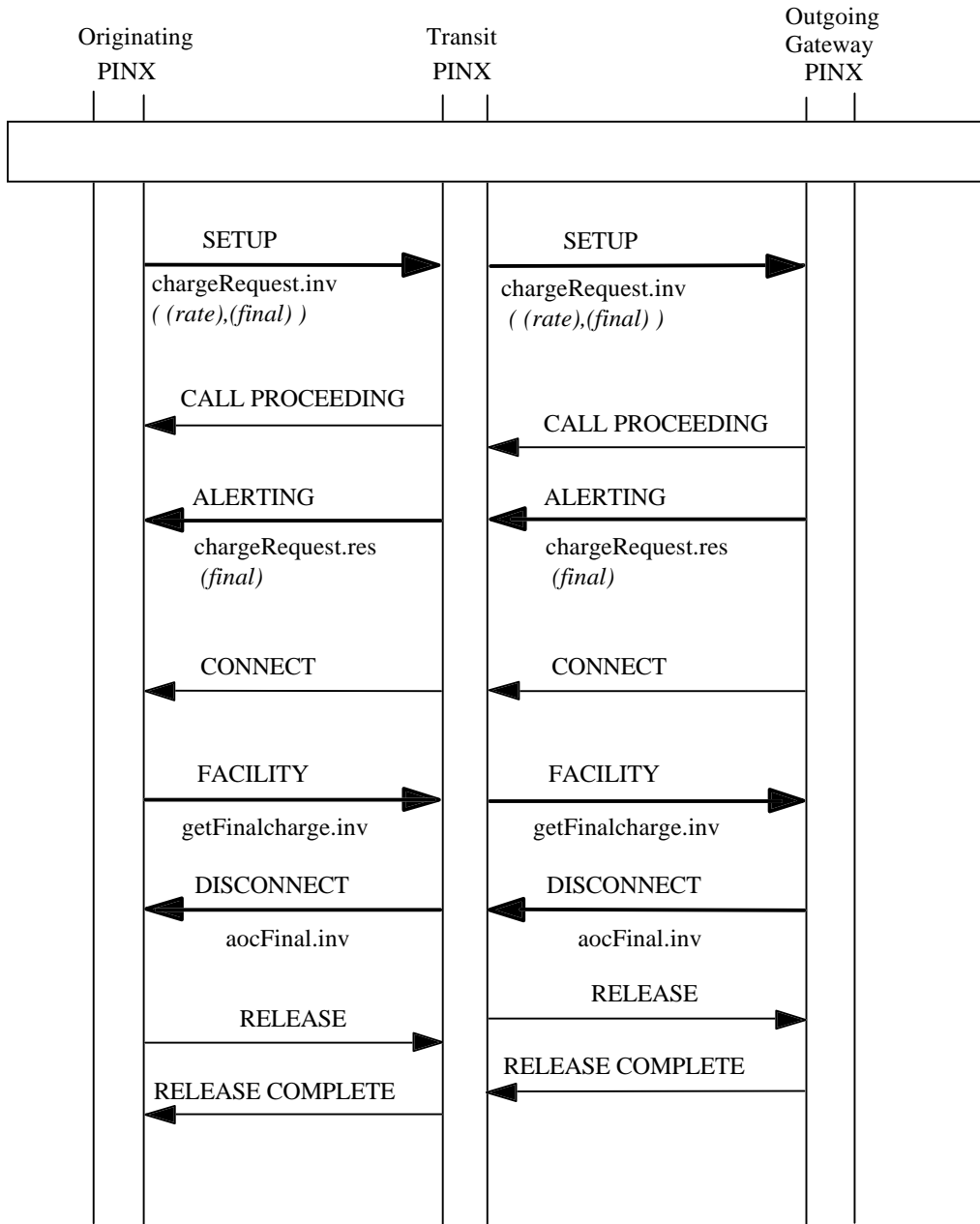
Figure B.1 - Example of normal operation of SS-AOC. Invocation during call set-up

B.2 Example message sequence for normal operation of SS-AOC with request during the call

Figure B.2 shows an example of normal operation of SS-AOC when the invocation of the service is done during the call. Three adviceModeCombinations, "interim", "rate" and "rateIterim", are requested and "rateInterim" is accepted.

Figure B.2 - Example of normal operation of SS-AOC. Invocation during the call

B.3 Example message sequence showing rejection of a request for charge information when the call terminates within the PISN

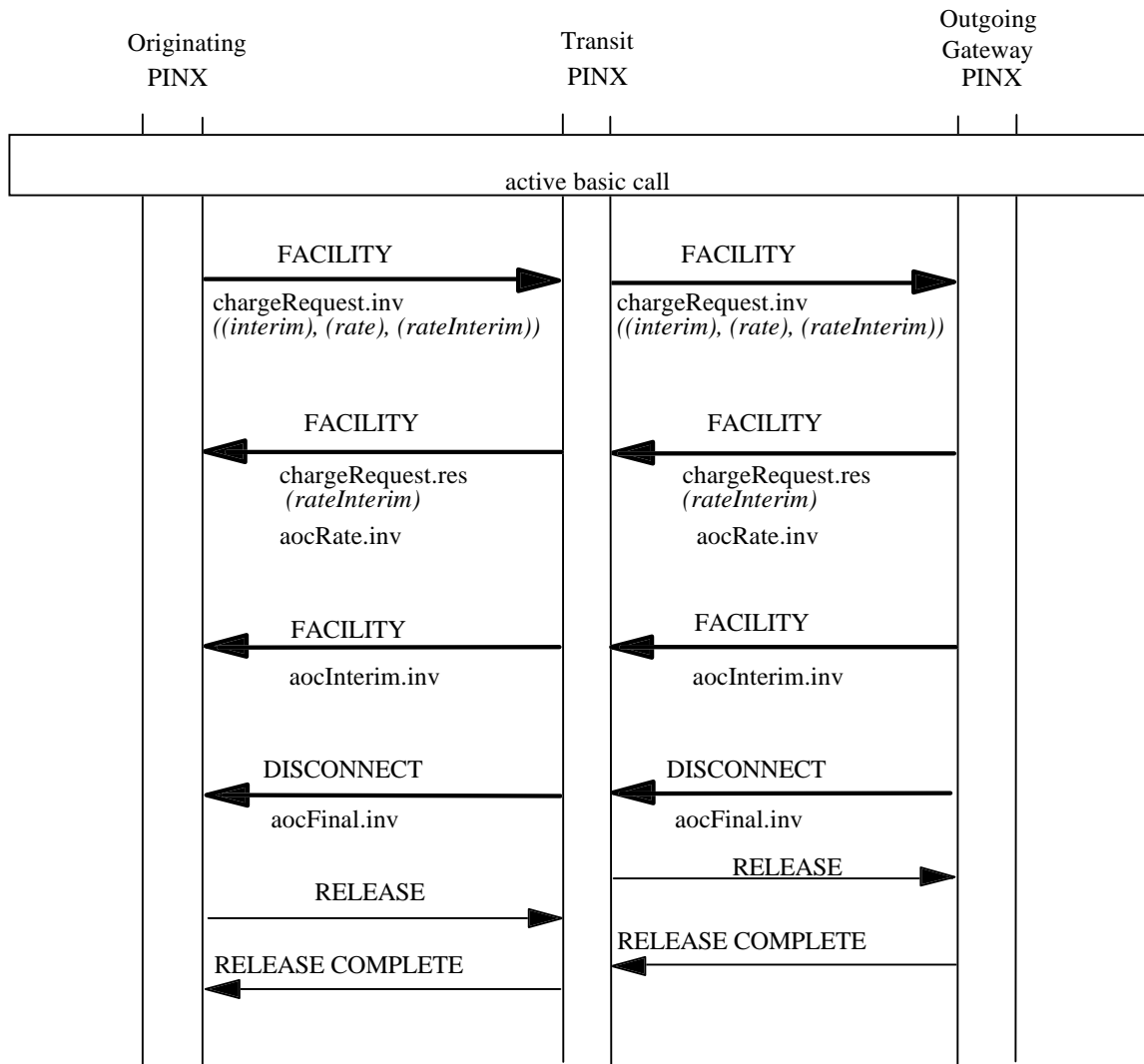Figure B.3 shows an example of abnormal operation of SS-AOC when the invocation of the service is rejected at the Terminating PINX.
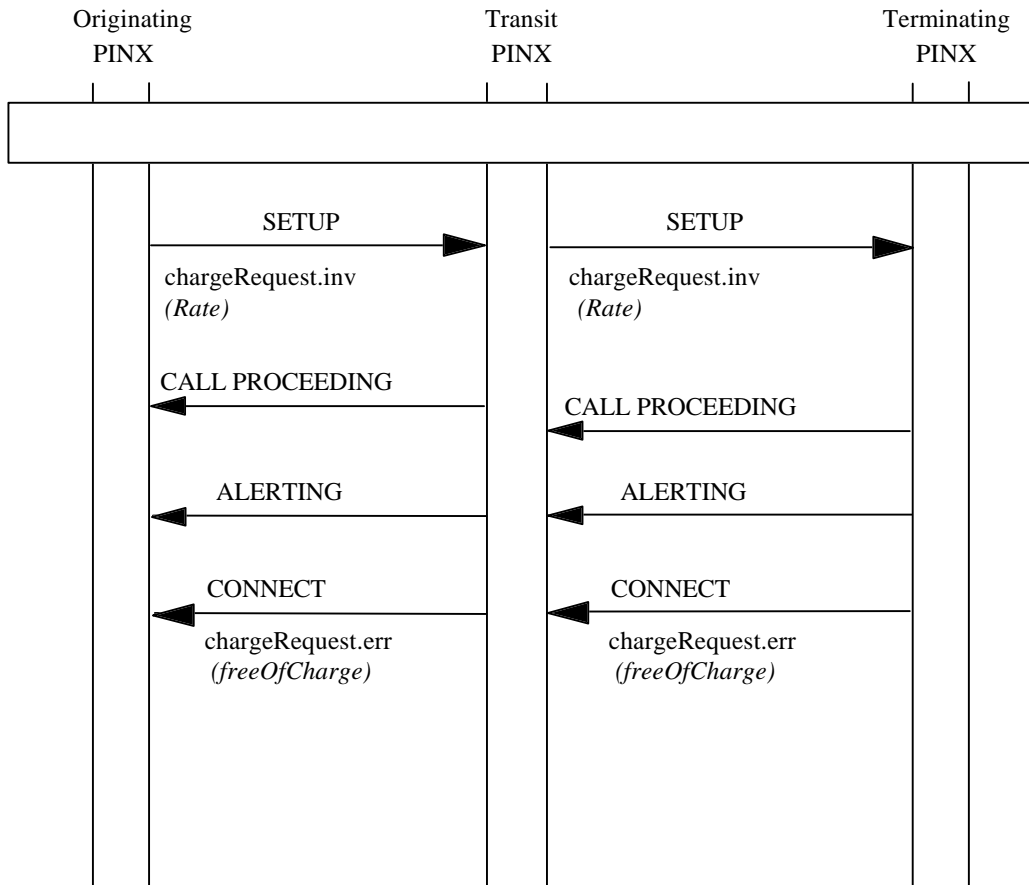
Figure B.3 - Example of abnormal operation of SS-AOC. Invocation during call set-up

Annex C

Specification and Description Language (SDL)
Representation of procedures

The diagrams in this annex use the Specification and Description Language defined in ITU-T Recommendation Z.100 (1999).

Each diagram represents the behaviour of an SS-AOC Supplementary Service Control entity at a particular type of PINX. In accordance with the protocol model described in ECMA-165, the Supplementary Service Control entity uses, via the Coordination Function, the services of Generic Functional Procedures Control and Basic Call Control.

Where an output symbol represents a primitive to the Coordination Function, and that primitive results in a message being sent, the output symbol bears the name of the message and any remote operations APDU(s) or notification(s) contained in that message. In the case of a message specified in ECMA-143, basic call actions associated with the sending of that message are deemed to occur.

Where an input symbol represents a primitive from the Coordination Function, and that primitive is the result of a message being received, the input symbol bears the name of the message and any remote operations APDU(s) or notification(s) contained in that message. In the case of a message specified in ECMA-143, basic call actions associated with the receipt of that message are deemed to have occurred.

The following abbreviations are used:

inv.   invoke APDU

res.   return result APDU

err.   return error APDU

rej.   reject APDU

C.1   SDL Representation of SS-AOC at the Originating PINX

Figure C.1 shows the behaviour of an SS-AOC Supplementary Service Control entity within the Originating PINX.

Input signals from the left and output signals to the left represent internal primitives.

Input signals from the right and output signals to the right represent primitives to and from the Coordination Function in respect of messages sent and received.

Process QSIG_AOC   Orig_PTNX (18)

Figure C.1.1 - SDL Representation of SS-AOC at the Originating PINX

Process QSIG_AOC   Orig_2 (18)

Process QSIG_AOC   Orig_3 (18)

Aoc-Orig-Active

DISCONNECT
aocFinal.inv

DISCONNECT
without
aocFinal.inv

No   interim or final
                charge
                provision

                                Yes

Final Charge   No Final
                                Charge

Continue
basic call
release

Aoc-Orig-Idle

Aoc-Orig-Active

aocInterim.inv   aocRate.inv

Charge
Information

Aoc-Orig-Active

Figure C.1.3 - SDL Representation of SS-AOC at the Originating PINX (continued)

Process QSIG_AOC   Orig_4 (18)

Figure C.1.4 - SDL Representation of SS-AOC at the Originating PINX (continued)

Process QSIG_AOC   Orig_5 (18)

Aoc-Orig-Active

Release request

interim or final charge provision

No →

Yes →

Release basic call

Aoc-Orig-Idle

FACILITY getFinalcharge.inv

Start T2

Aoc-Orig-Wait-Charge

DISCONNECT

DISCONNECT aocFinal.inv

T2 Expiry

Stop T2

Stop T2

No Final Charge

no Final Charge

Final Charge

Release basic call

Continue Basic call clearing

Continue Basic Call Clearing

Aoc-Orig-Idle

Aoc-Orig-Idle

Aoc-Orig-Idle

Figure C.1.5 - SDL Representation of SS-AOC at the Originating PINX (continued)

C.2   SDL Representation of SS-AOC at the Outgoing Gateway PINX

Figure C.2 shows the behaviour of an SS-AOC Supplementary Service Control entity within the Outgoing Gateway PINX.

Input signals from the right and output signals to the right represent primitives to and from the accounting function.

Input signals from the left and output signals to the left represent primitives to and from the Coordination Function in respect of messages sent and received.

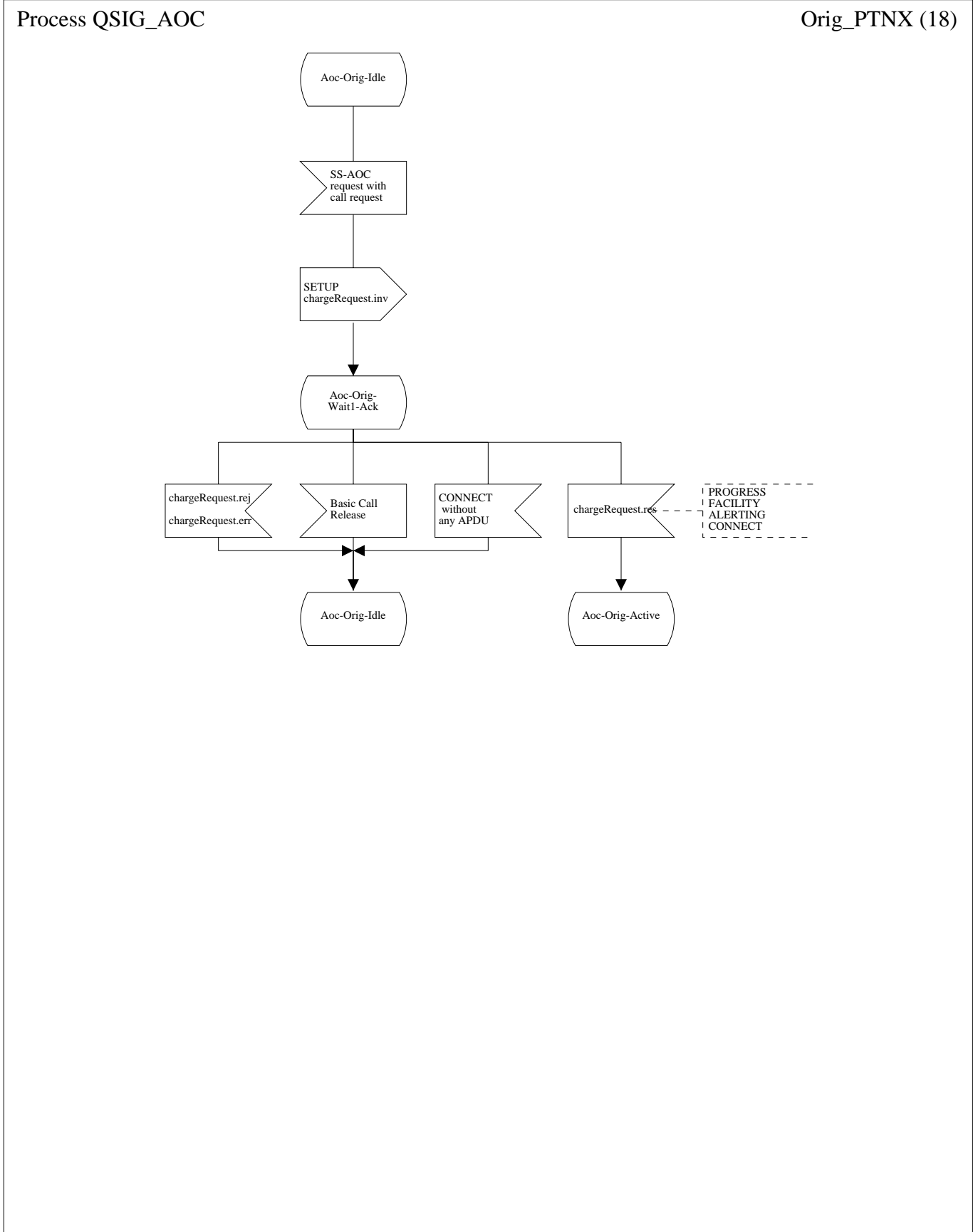Process QSIG_AOC   Out_Gwy_PTNX (16)
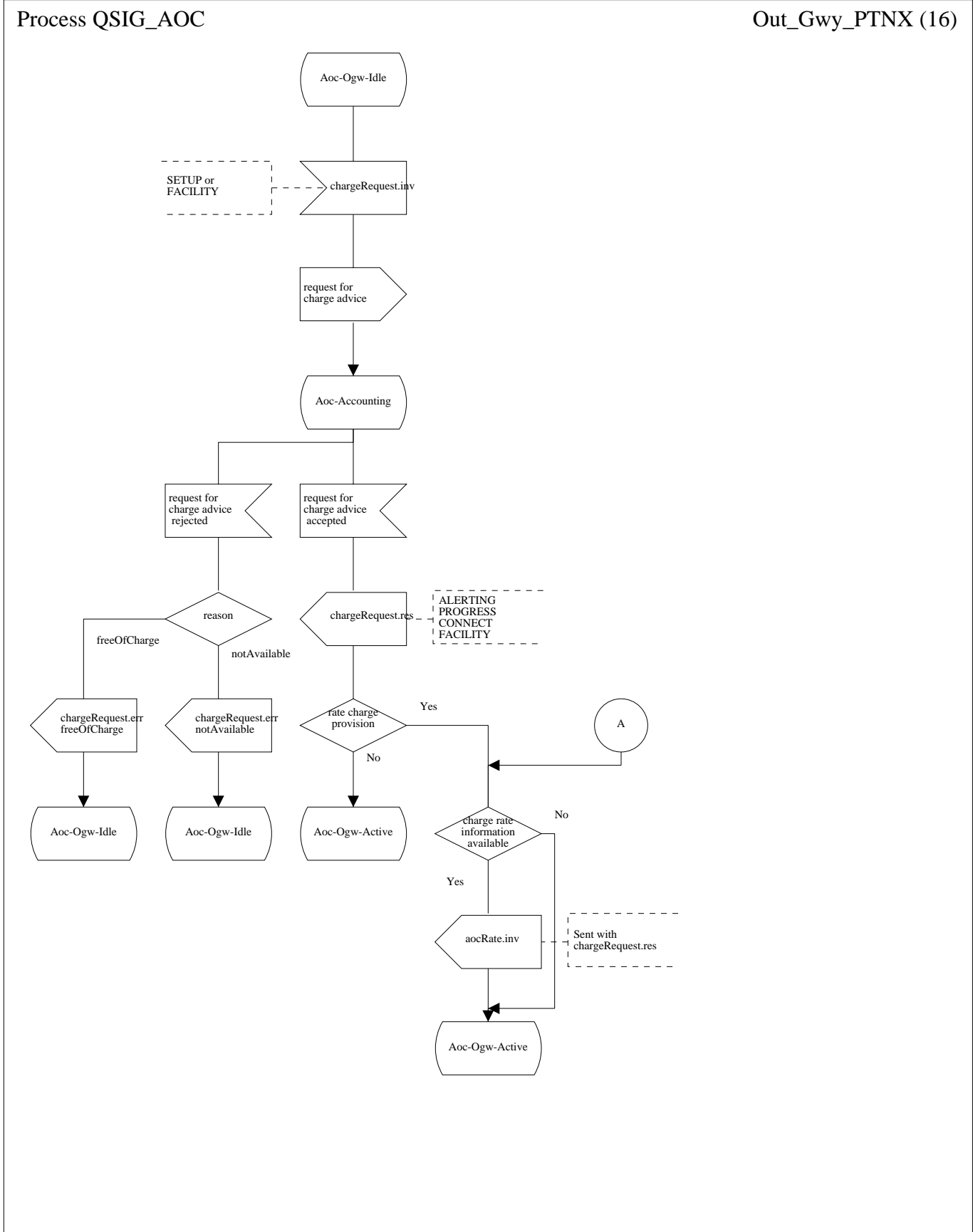
Figure C.2.1 - SDL Representation of SS-AOC at the Outgoing Gateway PINX

Figure C.2.2 - SDL Representation of SS-AOC at the Outgoing Gateway PINX (continued)

Process QSIG_AOC   Out_Gwy_3 (16)
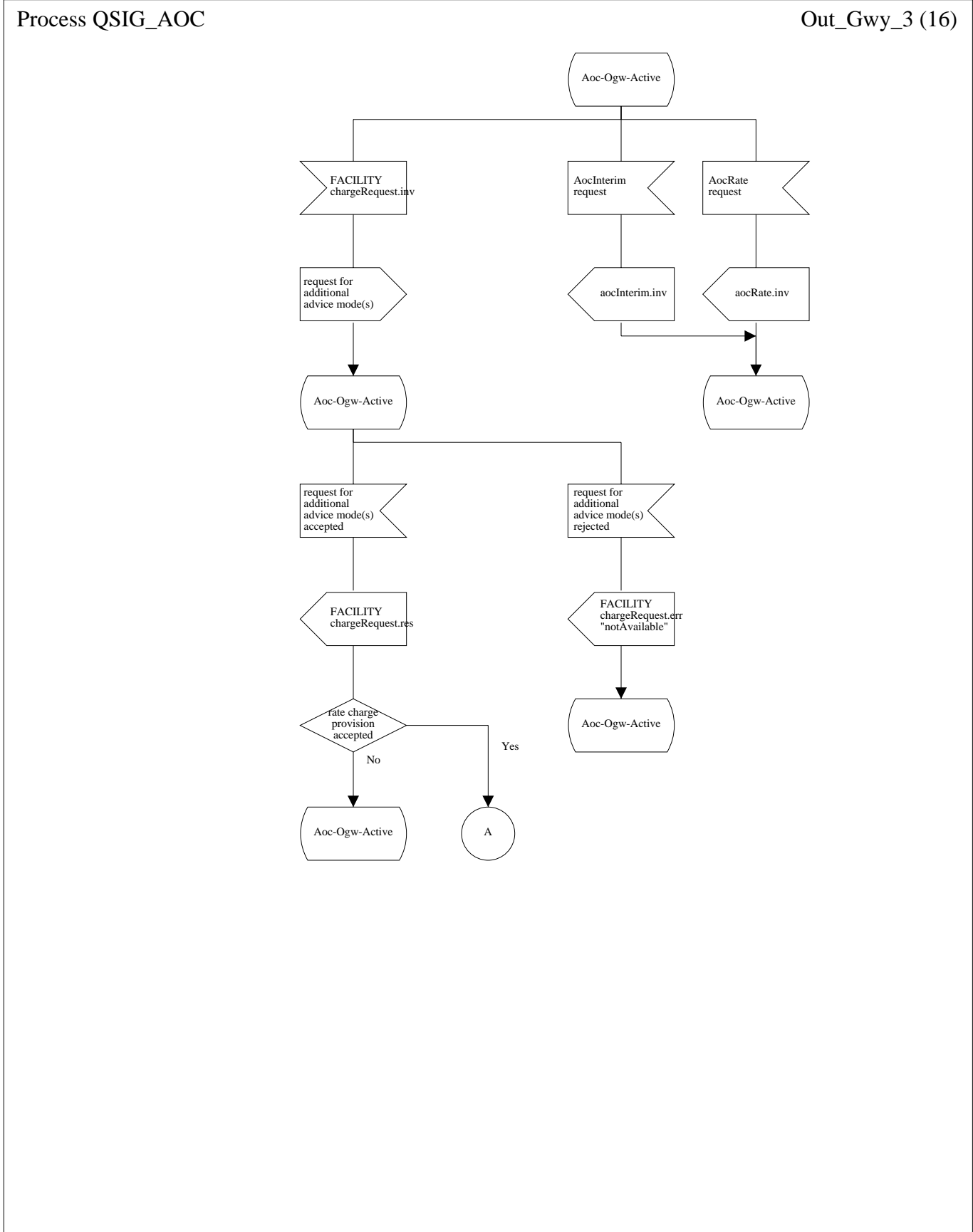
Figure C.2.3 - SDL Representation of SS-AOC at the Outgoing Gateway PINX (continued)

Process QSIG_AOC   Out_Gwy_4 (18)

Aoc-Ogw-Active

FACILITY
aocComplete.inv

Transfer invoked

Aoc-Transfer-
Accounting

user A continues	user A is not	user A stops
to be charged	charged before	being charged
	or after transfer	

FACILITY
aocComplete.res
"aocContinueCharging"

FACILITY
aocComplete.res
"aocFreeOfCharge"

FACILITY
aocComplete.res
"aocStopCharging"

final
charge   No
provision

Aoc-Ogw-Idle

interim or final
charge   No
provision

Yes

Aoc-Final-
Accounting

Aoc-Ogw-Idle

Yes

Aoc-Final-
Accounting

Aoc-Ogw-Idle

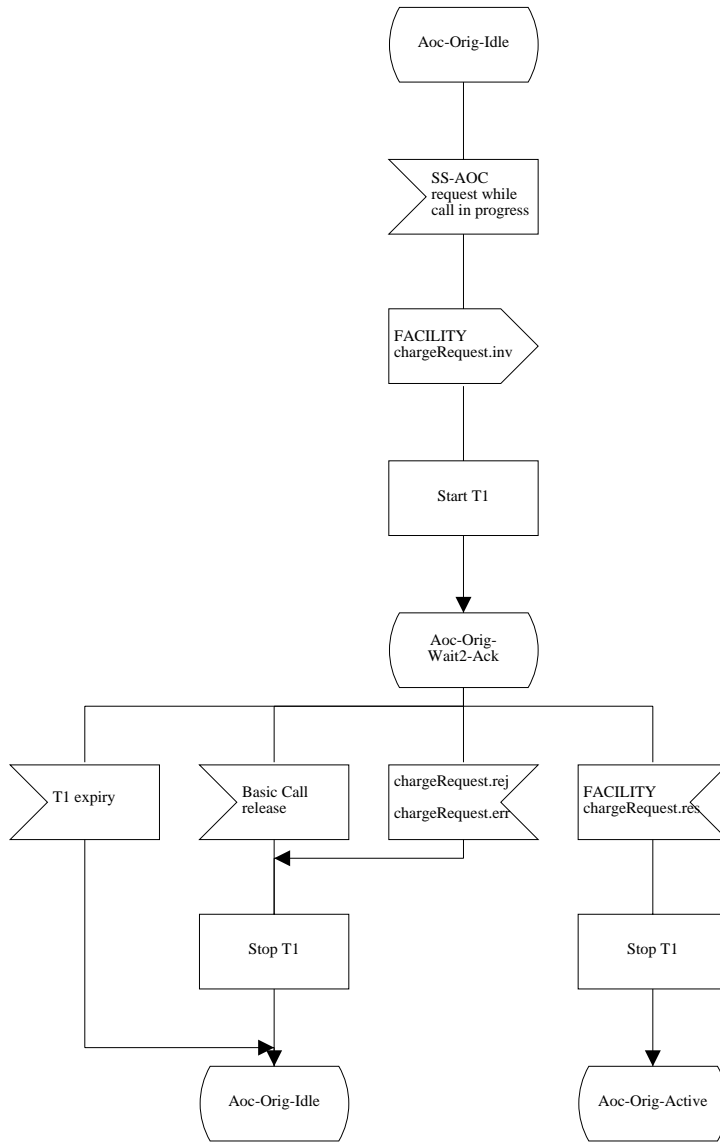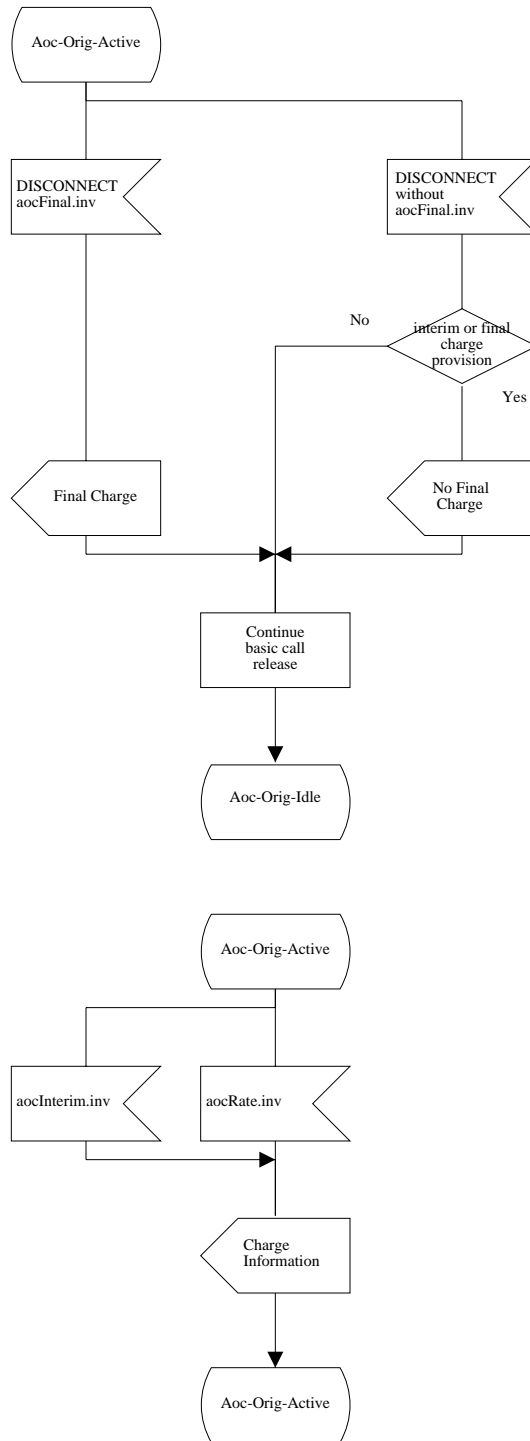**Figure C.2.4 - SDL Representation of SS-AOC at the Outgoing Gateway PINX (continued)**

Process QSIG_AOC   Out_Gwy_5 (18)
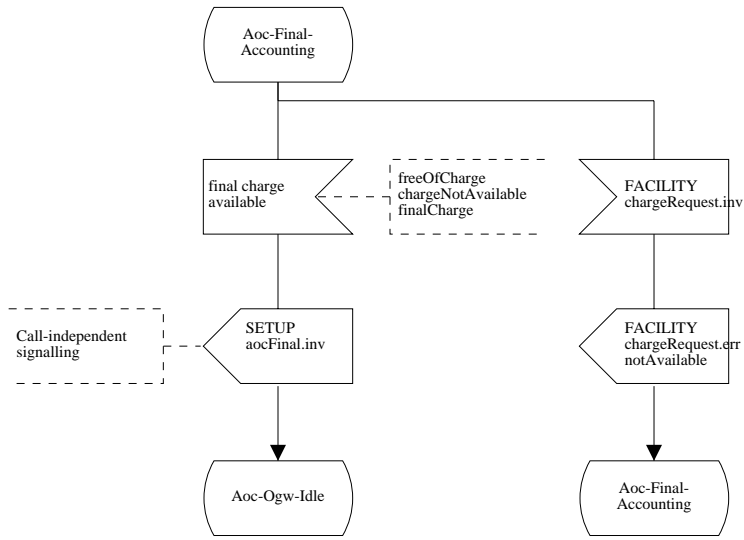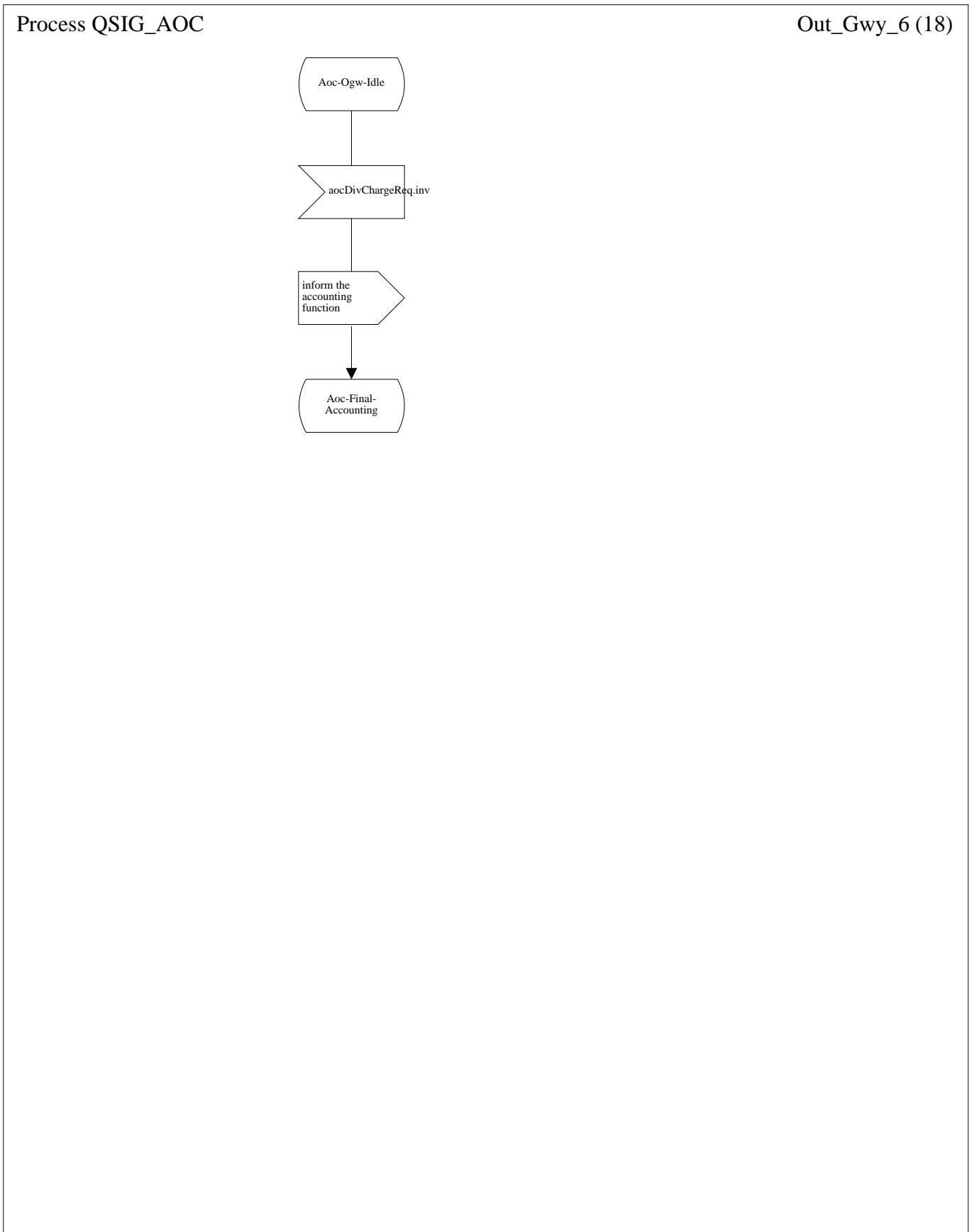
Figure C.2.6 - SDL Representation of SS-AOC at the Outgoing Gateway PINX

C.3 SDL Representation of SS-AOC at the Terminating PINX

Figure C.3 shows the behaviour of an SS-AOC Supplementary Service Control entity within the Terminating PINX.

Input signals from the left and output signals to the left represent primitives to and from the Coordination Function in respect of messages sent and received.

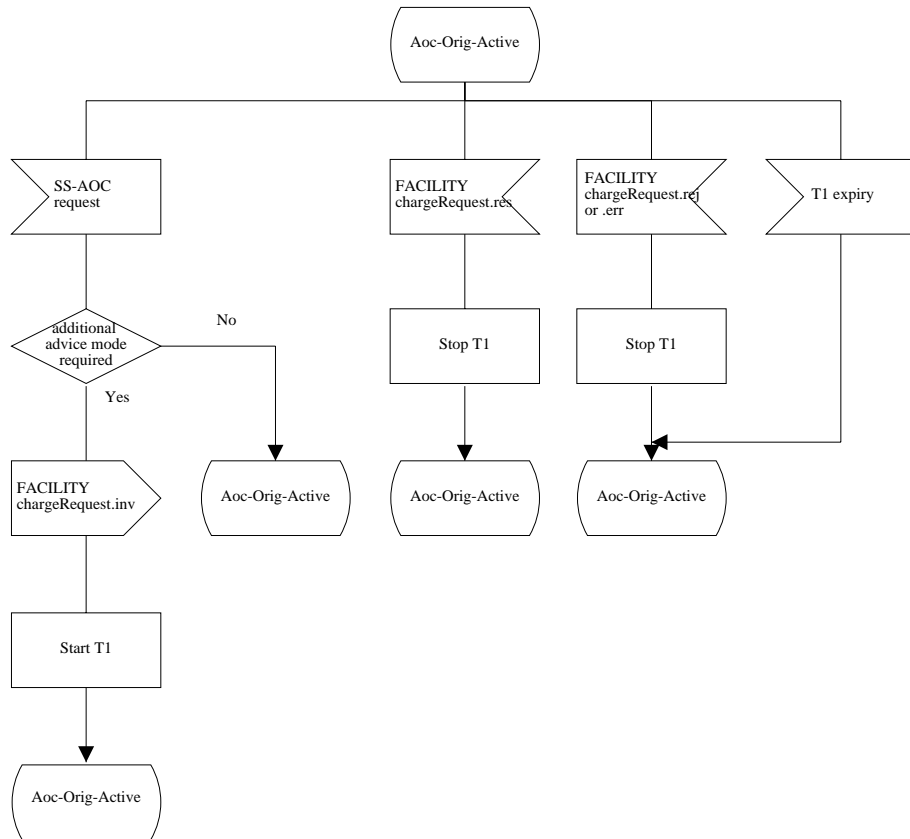Process QSIG_AOC   Term_PTNX_bis (18)
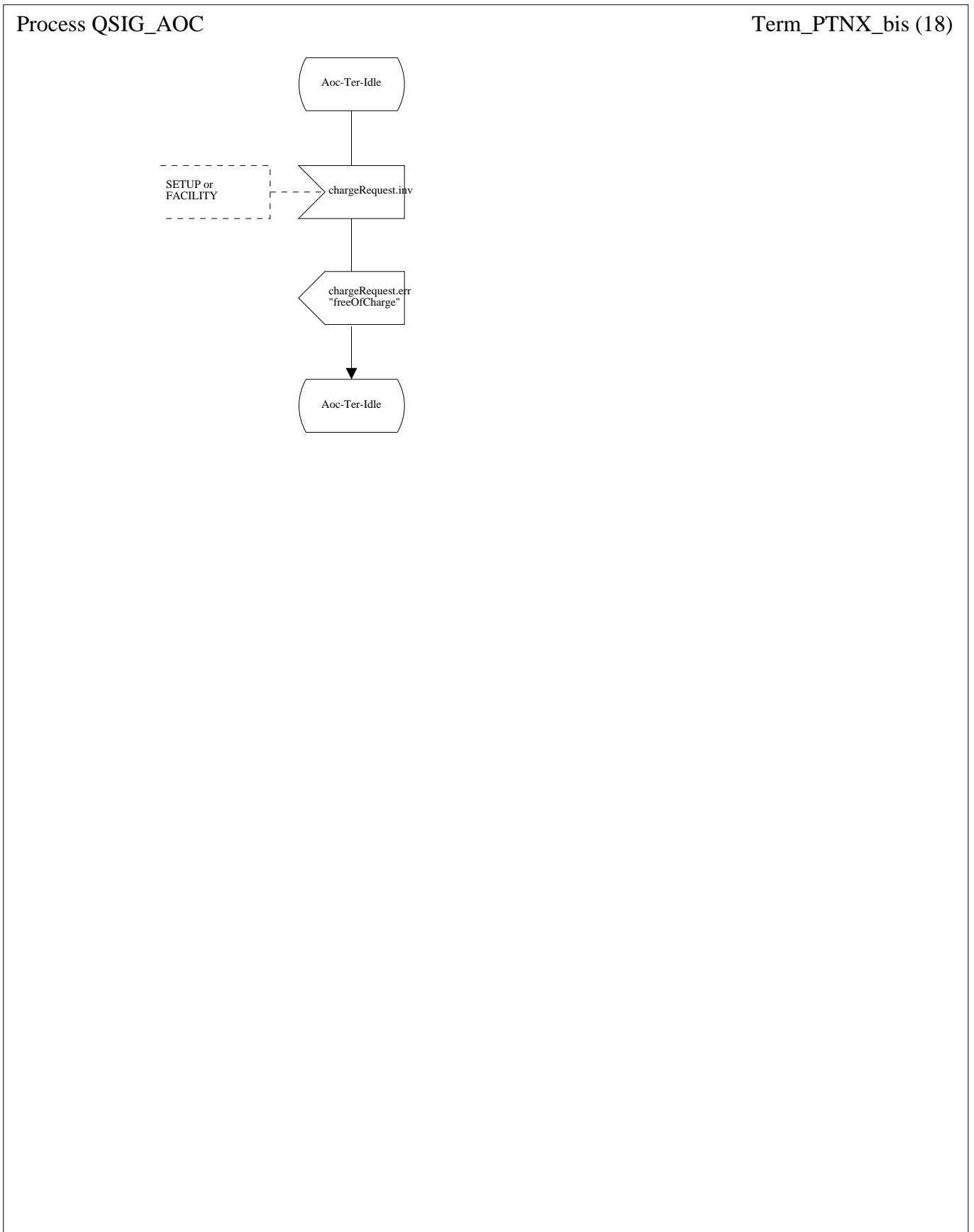
Figure C.3.1 - SDL Representation of SS-AOC at the Terminating PINX

C.4 SDL Representation of SS-AOC at the Originating PINX for the interaction with Call Transfer

Figure C.4 shows the behaviour of an SS-AOC Supplementary Service Control entity within the Originating PINX.

Input signals from the left and output signals to the left represent internal primitives.

Input signals from the right and output signals to the right represent internal primitives to and from the Coordination Function in respect of messages sent and received.

Process QSIG_AOC   Orig_CT_PTNX_bis (18)
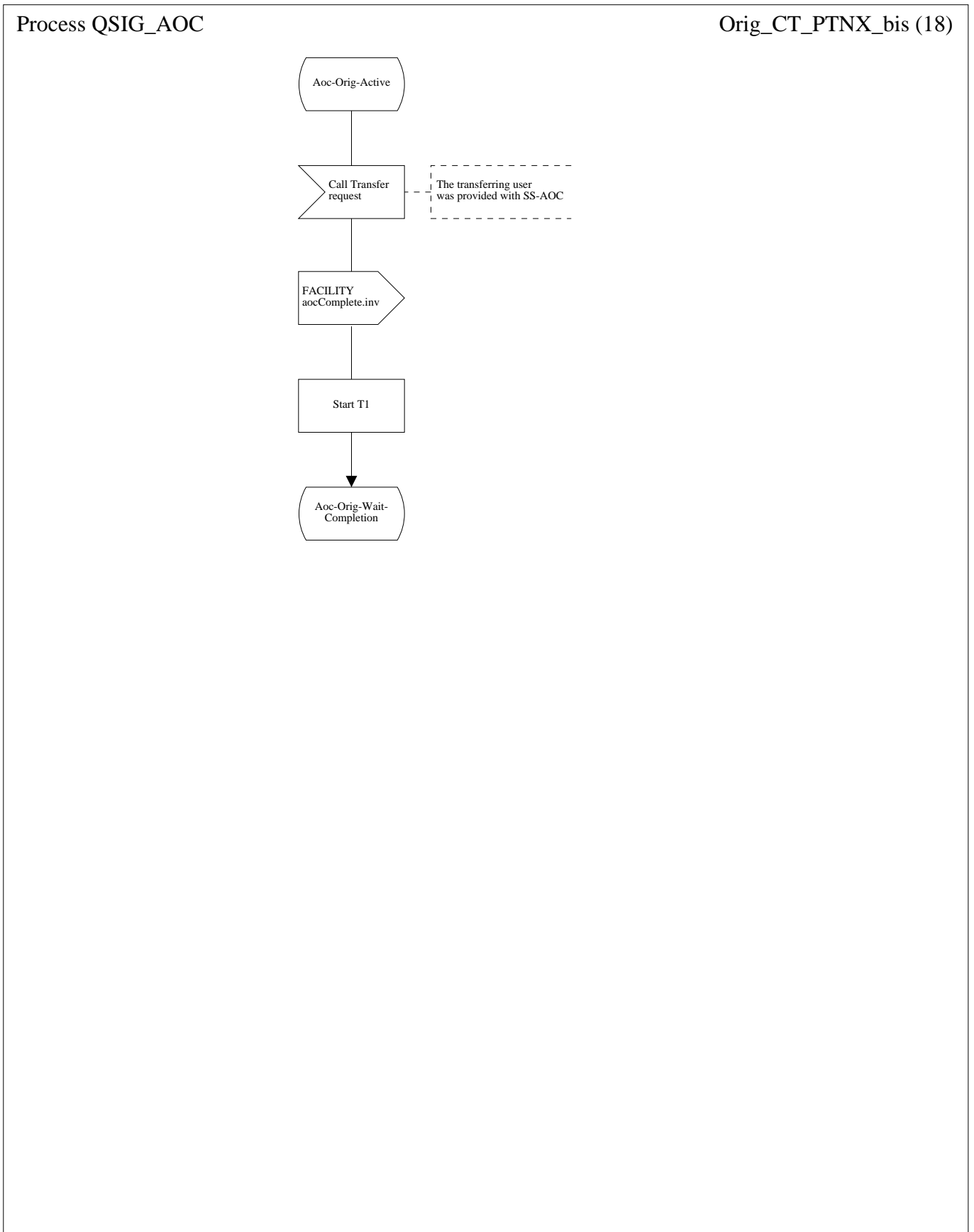
Figure C.4.1 - SDL Representation of SS-AOC at the Originating PINX - interaction with SS-CT

Process QSIG_AOC   Orig_CT_PTNX_Bis_2 (18)

Process QSIG_AOC   Orig_CT_2 (18)
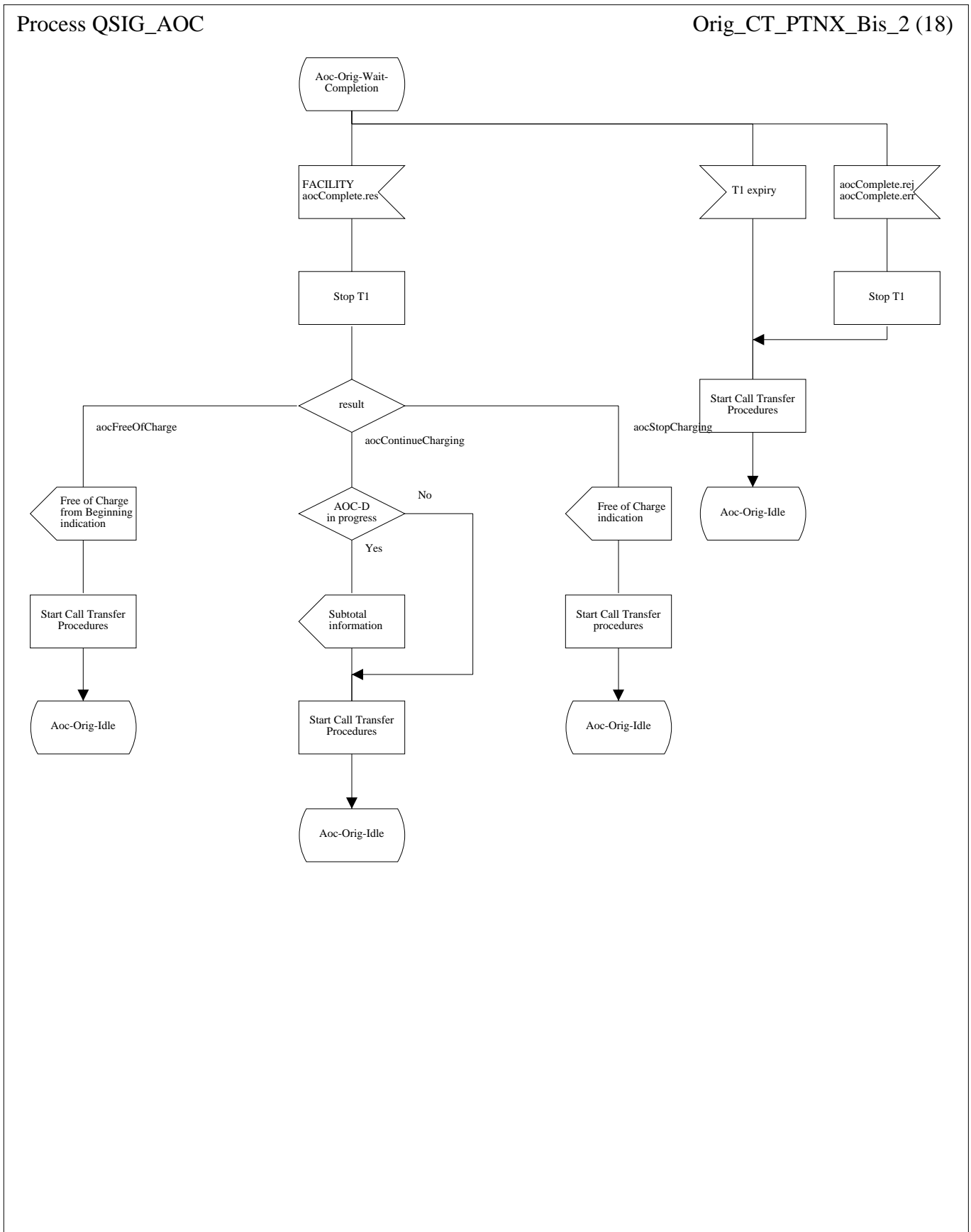
Figure C.4.3 - SDL Representation of SS-AOC at the Originating PINX - interaction with SS-CT (continued)

C.5 SDL Representation of SS-AOC at the Served PINX for the interaction with Call Diversion

Figure C.5 shows the behaviour of an SS-AOC Supplementary Service Control entity within the Served PINX.

Input signals from the left and output signals to the left represent internal primitives.

Input signals from the right and output signals to the right represent internal primitives to and from the Coordination Function in respect of messages sent and received.
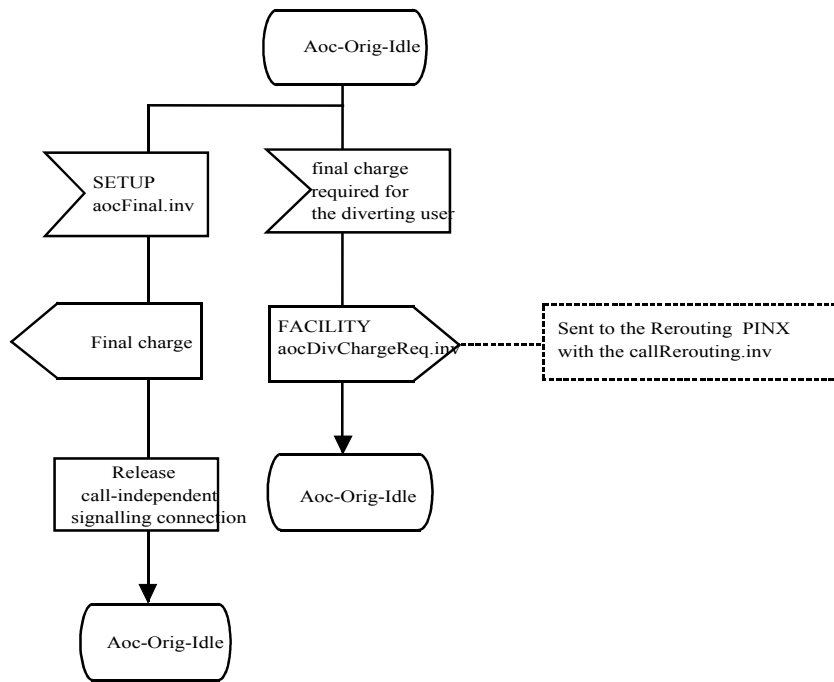
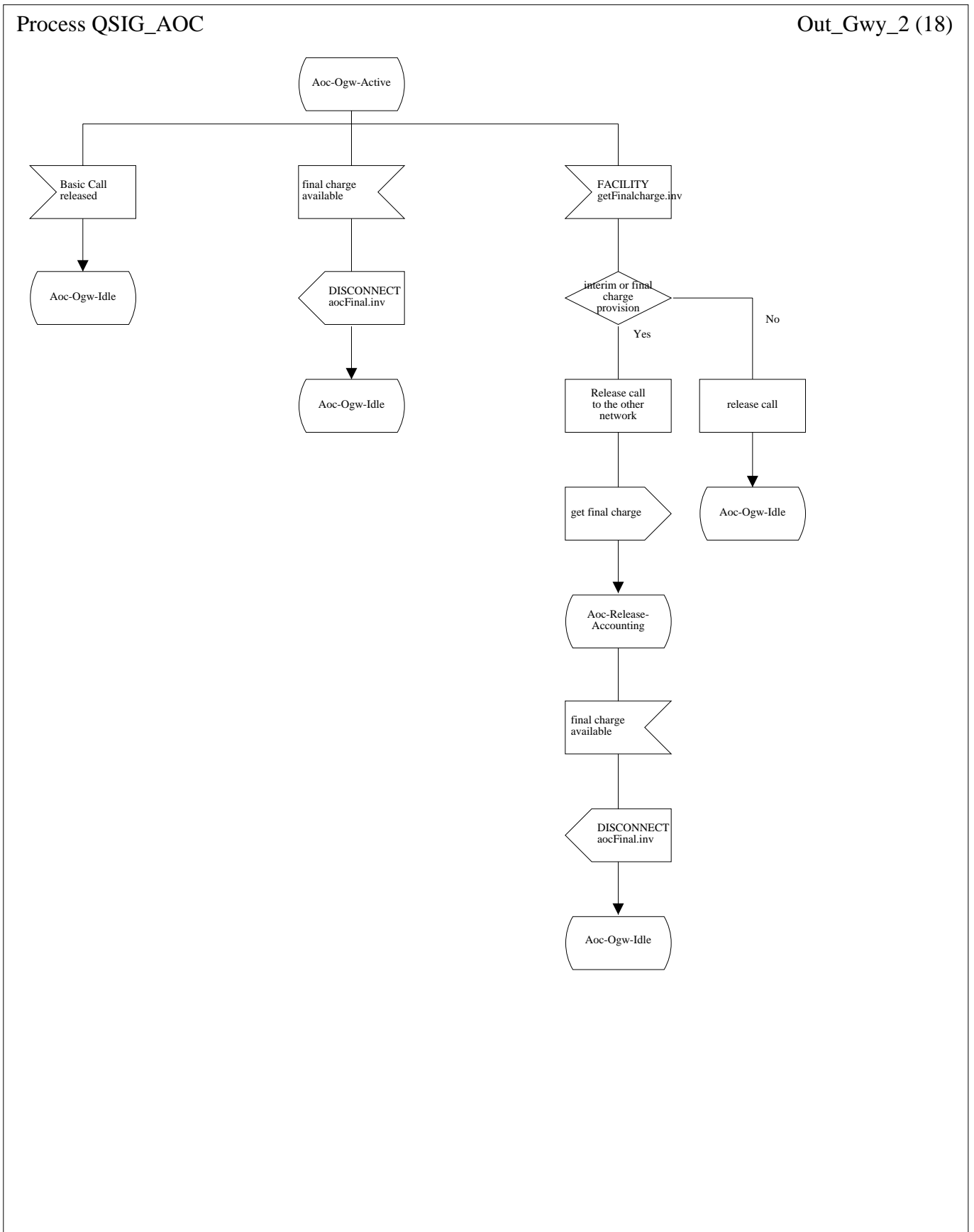**Figure C.5.1 - SDL Representation of SS-AOC at the Served User PINX - interaction with SS-DIV**

Annex D
(informative)

Imported ASN.1 definitions

The content of this annex has been deleted to remove duplicate ASN.1 definitions defined elsewhere.

Annex E
(normative)

ASN.1 definitions according to ITU-T Recs. X.208 / X.209

This annex lists all ASN.1 modules as they were defined in the second edition of ECMA-212, i.e. based on ITU-T Recommendations X.208 / X.209. Starting with the third edition the ASN.1 modules within ECMA-212 comply with ITU-T Recommendations X.680 / X.690. Please note that regardless of which version of these modules is used as a base of a QSIG implementation, the line encoding remains unchanged. Changes in future editions to modules based on X.680 / X.690 ASN.1 are not reflected in the modules in this annex.

Table E.1 - SS-AOC-Operations - based on ITU-T Recs. X.208 / X.209

```
SS-AOC-Operations
                    {iso (1) standard (0)  pss1-advice-of-charge (15050) advice-of-charge-operations (0)}


DEFINITIONS EXPLICIT TAGS ::=


BEGIN


IMPORTS          OPERATION, ERROR FROM Remote-Operation-Notation
                        {joint-iso-ccitt (2) remote-operations (4) notation (0)}
                 Extension FROM Manufacturer-specific-service-extension-definition
                        {iso (1) standard (0) pss1-generic-procedures (11582) msi-definition (0)}
                 notAvailable, supplementaryServiceInteractionNotAllowed
                 FROM General-Error-List
                        {ccitt (0) recommendation (0) q (17) 950 general-error-list (1)}
                 PartyNumber FROM Addressing-Data-Elements
                  { iso (1) standard (0) pss1-generic-procedures (11582) addressing-data-elements (9) };


AocRate              ::=    OPERATION
                            ARGUMENT AocRateArg


AocRateArg           ::=    SEQUENCE {
                                    aocRate     CHOICE {
                                            chargeNotAvailable    NULL,
                                            aocSCurrencyInfoList AOCSCurrencyInfoList },
                                    rateArgExtension   CHOICE {
                                        extension [1]    IMPLICIT Extension,
                                        multipleExtension  [2] IMPLICIT SEQUENCE OF Extension
                                                        } OPTIONAL}


AocInterim           ::=    OPERATION
                            ARGUMENT AocInterimArg
```

```
AocInterimArg          ::=    SEQUENCE {
                              interimCharge  CHOICE {
                                    chargeNotAvailable [0] IMPLICIT     NULL,
                                    freeOfCharge       [1] IMPLICIT     NULL,
                                    specificCurrency        SEQUENCE {
                                          recordedCurrency [1] IMPLICIT RecordedCurrency,

                                          interimBillingId[2] IMPLICIT InterimBillingId OPTIONAL }
                                                        },
                              interimArgExtension   CHOICE {
                                    extension [1]    IMPLICIT Extension,
                                    multipleExtension  [2] IMPLICIT SEQUENCE OF Extension
                                                        } OPTIONAL
                              }


AocFinal               ::=    OPERATION
                              ARGUMENT AocFinalArg


AocFinalArg            ::=    SEQUENCE {
                              finalCharge  CHOICE {
                                    chargeNotAvailable [0] IMPLICIT NULL,
                                    freeOfCharge [1] IMPLICIT NULL,
                                    specificCurrency    SEQUENCE {
                                          recordedCurrency  [1] IMPLICIT RecordedCurrency,
                                             finalBillingId[2]  IMPLICIT FinalBillingId OPTIONAL }
                                                         },
                              chargingAssociation   ChargingAssociation        OPTIONAL,
                              finalArgExtension   CHOICE {
                                    extension [1]    IMPLICIT Extension,
                                    multipleExtension  [2] IMPLICIT SEQUENCE OF Extension
                                                        } OPTIONAL
                              }

AOCSCurrencyInfoList   ::= SEQUENCE SIZE(1..10) OF AOCSCurrencyInfo


AOCSCurrencyInfo       ::=    SEQUENCE {
                             chargedItem         ChargedItem,
                              rateType    CHOICE {
                                    durationCurrency        [1] IMPLICIT DurationCurrency,
                                    flatRateCurrency        [2] IMPLICIT FlatRateCurrency,
                                    volumeRateCurrency [3] IMPLICIT VolumeRateCurrency,
                                    specialChargingCode  SpecialChargingCode,
                                    freeOfCharge              [4] IMPLICIT NULL,
                                    currencyInfoNotAvailable [5] IMPLICIT NULL,
                                    freeOfChargefromBeginning [6] IMPLICIT NULL
                                          }
                                    }
```

ChargedItem	::=	ENUMERATED {
		    basicCommunication (0),
		    callAttempt (1),
		    callSetup (2),
		    userToUserInfo (3),
		    operationOfSupplementaryServ (4) }

DurationCurrency	::=	SEQUENCE {
		    dCurrency [1] IMPLICIT Currency,
		    dAmount [2] IMPLICIT Amount,
		    dChargingType [3] IMPLICIT ChargingType,
		    dTime [4] IMPLICIT Time,
		    dGranularity [5] IMPLICIT Time OPTIONAL }

FlatRateCurrency	::=	SEQUENCE {
		    fRCurrency [1] IMPLICIT Currency,
		    fRAmount [2] IMPLICIT Amount }

VolumeRateCurrency	::=	SEQUENCE {
		    vRCurrency [1] IMPLICIT Currency,
		    vRAmount [2] IMPLICIT Amount,
		    vRVolumeUnit [3] IMPLICIT VolumeUnit
		         }

SpecialChargingCode	::=	INTEGER (1..10)

RecordedCurrency	::=	SEQUENCE {
		    rCurrency [1] IMPLICIT Currency,
		    rAmount [2] IMPLICIT Amount }

InterimBillingId	::=	ENUMERATED {
		    normalCharging (0),
		    creditCardCharging (2) }

FinalBillingId	::=	ENUMERATED {
		    normalCharging (0),
		    creditCardCharging (2),
		    callForwardingUnconditional (3),
		    callForwardingBusy (4),
		    callForwardingNoReply (5),
		    callDeflection (6),
		    callTransfer (7) }

Currency	::=	IA5String (SIZE (0..10)) -- SIZE(0) shall indicate the default currency of the PISN -- The representation of other currencies is outside the scope of this standard

Table E.1 - SS-AOC-Operations - based on ITU-T Recs. X.208 / X.209 (continued)

```
Amount                ::=    SEQUENCE {
                             currencyAmount    [1] IMPLICIT CurrencyAmount,
                             multiplier        [2] IMPLICIT Multiplier  }


CurrencyAmount        ::=    INTEGER (0..16777215)


Multiplier            ::=    ENUMERATED {
                             oneThousandth (0),
                             oneHundredth  (1),
                             oneTenth      (2),
                             one           (3),
                             ten           (4),
                             hundred       (5),
                             thousand      (6) }


Time                  ::=    SEQUENCE {
                             lengthOfTimeUnit  [1] IMPLICIT LengthOfTimeUnit,
                             scale             [2] IMPLICIT Scale }


LengthOfTimeUnit      ::=    INTEGER (0..16777215)


Scale                 ::=    ENUMERATED {
                             oneHundredthSecond (0),
                             oneTenthSecond     (1),
                             oneSecond          (2),
                             tenSeconds         (3),
                             oneMinute          (4),
                             oneHour            (5),
                             twentyFourHours    (6) }


VolumeUnit            ::=    ENUMERATED {
                             octet     (0),
                             segment   (1),
                             message   (2) }


ChargingType          ::=    ENUMERATED {
                             continuousCharging  (0),
                             stepFunction        (1) }


ChargingAssociation   ::=    CHOICE {
                             chargeNumber [0]    PartyNumber,
                             chargeIdentifier    ChargeIdentifier }


ChargeIdentifier      ::=    INTEGER (-32768..32767)
```

Table E.1 - SS-AOC-Operations - based on ITU-T Recs. X.208 / X.209 (continued)

```
ChargeRequest          ::=    OPERATION
                              ARGUMENT        ChargeRequestArg
                              RESULT          ChargeRequestRes
                              ERRORS          {
                                      freeOfCharge,
                                      supplementaryServiceInteractionNotAllowed,
                                      notAvailable, unspecified }


GetFinalcharge         ::=    OPERATION
                              ARGUMENT        DummyArg


ChargeRequestArg       ::=    SEQUENCE {
                       adviceModeCombinations    SEQUENCE SIZE(0..7) OF
                                        AdviceModeCombination,
                       chargeReqArgExtension  CHOICE {
                                      extension [1]    IMPLICIT Extension,
                                      multipleExtension  [2] IMPLICIT SEQUENCE OF Extension
                                              } OPTIONAL
                                      }


ChargeRequestRes       ::=    SEQUENCE {
                       adviceModeCombination  AdviceModeCombination,
                       chargeReqResExtension     CHOICE {
                              extension [1]    IMPLICIT Extension,
                              multipleExtension  [2] IMPLICIT SEQUENCE OF Extension
                                              } OPTIONAL
                                      }


AdviceModeCombination ::=     ENUMERATED {        -- advice mode combination
                              rate                (0),    -- charge rate provision
                              rateInterim         (1),    -- charge rate and interim charge provision
                              rateFinal           (2),    -- charge rate and final charge provision
                              interim             (3),    -- interim charge provision
                              final               (4),    -- final charge provision
                              interimFinal        (5),    -- interim charge and final charge provision
                              rateInterimFinal    (6)}    -- charge rate, interim charge and final
                                                          -- charge provision


DummyArg               ::=    CHOICE{
                              none                NULL,
                              extension           [1] IMPLICIT Extension,
                              multipleExtension  [2] IMPLICIT SEQUENCE OF Extension
                              }
```

Table E.1 - SS-AOC-Operations - based on ITU-T Recs. X.208 / X.209 (continued)

```
-- The following OPERATION applies for the interaction with Call Transfer

AocComplete            ::=    OPERATION
                              ARGUMENT AocCompleteArg
                              RESULT    AocCompleteRes
                              ERRORS    {supplementaryServiceInteractionNotAllowed}

AocCompleteArg         ::=    SEQUENCE {
                                      chargedUser            PartyNumber,
                                      chargingAssociation   ChargingAssociation OPTIONAL,
                                      completeArgExtension    CHOICE {
                                              extension [1]    IMPLICIT Extension,
                                              multipleExtension  [2] IMPLICIT SEQUENCE OF Extension
                                                                  } OPTIONAL
                                      }
AocCompleteRes::=      SEQUENCE {
                                      chargingOption ChargingOption,
                                      completeResExtension    CHOICE {
                                              extension [1]    IMPLICIT Extension,
                                              multipleExtension  [2] IMPLICIT SEQUENCE OF Extension
                                                                  } OPTIONAL
                                      }

ChargingOption         ::=    ENUMERATED{
                                      aocFreeOfCharge      (0),
                                      aocContinueCharging (1),
                                      aocStopCharging      (2)
                                              }
-- The following OPERATION applies for the interaction with Call Diversion

AocDivChargeReq        ::=    OPERATION
                                      ARGUMENT AocDivChargeReqArg

AocDivChargeReqArg    ::= SEQUENCE {
                                      divertingUser          PartyNumber,
                                      chargingAssociation   ChargingAssociation        OPTIONAL,
                                      diversionType          DiversionType,
                                      aocDivChargeReqArgExt    CHOICE {
                                              extension [1]    IMPLICIT Extension,
                                              multipleExtension  [2] IMPLICIT SEQUENCE OF Extension
                                                                      } OPTIONAL
                                      }

DiversionType          ::= ENUMERATED {
                                      callForwardingUnconditional (0),
                                      callForwardingBusy            (1),
                                      callForwardingNoReply        (2),
                                      callDeflection                (3) }
```

Table E.1 - SS-AOC-Operations - based on ITU-T Recs. X.208 / X.209 (concluded)

chargeRequest	ChargeRequest	::=	localValue 59
getFinalCharge	GetFinalcharge	::=	localValue 60
aocFinal	AocFinal	::=	localValue 61
aocInterim	AocInterim	::=	localValue 62
aocRate	AocRate	::=	localValue 63
aocComplete	AocComplete	::=	localValue 64
aocDivChargeReq	AocDivChargeReq ::=		localValue 65
Unspecified	::= ERROR PARAMETER Extension		
unspecified	Unspecified ::= localValue 1008		
freeOfCharge	ERROR ::= localValue 1016		
END	-- of SS-AOC-Operations		