MASTER

Interworking functions in a wireless broadband local area network

Peters, M.J.H.

Award date:
1996

FACULTEIT DER ELEKTROTECHNIEK
TECHNISCHE UNIVERSITEIT EINDHOVEN
VAKGROEP TELECOMMUNICATIE EC

INTERWORKING FUNCTIONS IN A WIRELESS BROADBAND LOCAL AREA NETWORK (PART 1)

BY

ING. M.J.H. PETERS

Verslag van het afstudeerwerk
uitgevoerd van 15-11-1995 tot 27-6-1996
Afstudeerhoogleraar: prof.dr.ir. G. Brussaard
Begeleider: dr.ir. P.F.M. Smulders

Summary

The reason for this research was the presence of a need for developing and implementing interworking functions in the Advanced Communications Technologies and Services (ACTS) project MEDIAN (wireless broadband customer premises network/ local area network for professional and residential multimedia applications). The MEDIAN project is characterized by:

- the transmission frequencies, considered to be placed in the millimeter wave band (60 GHz), which was up to days usually used for military purpose;
- the used transfer technology, being the Asynchronous Transfer Mode (ATM);
- the high bitrates available to the users, 155 Mbit/s maximum throughput;

The research has been done in association with TNO-FEL and University of Rome "La Sapeinza". Within the MEDIAN project TNO-FEL is (among other things) responsible for the implementation of the interworking layer. University of Rome "La Sapeinza" is (among other things) responsible for the development and implementation of the Medium Access Control (MAC) layer.

The three different network units on which the MEDIAN system is based, are:

- the MEDIAN Server Station (MSS): local exchange with the core network;
- the MEDIAN Remote Station (MRS): unit which can be used to setup a local area network;
- the MEDIAN Portable Station (MPS): the interface with the MEDIAN end-user.

In *Chapter one* an introduction is given to ATM. Because of the large number of networks (telephone network, telex network, private network) and the services (including multi-media) needed to be transported by those networks, the infrastructure is very complex and the accompanying administration loads are heavy. The simplification of the infrastructure by sharing one network for the transportation of different services could be a solution to both the increasing complexity of the infra structure and the heavy administrative load. This simplification could be carried out by the introduction of the Asynchronous Transfer Mode (ATM). This chapter also contains a description of the MEDIAN project and its scenarios of application.

In *Chapter two* the aspects of signalling, i.e., establishing/maintaining/releasing of a call/connection are treated. This chapter called SIGNALLING describes the different states in which an ATM call/connection can be situated. It continuous with the introduction and explanation of the needed signalling messages. Within the MEDIAN system three different types of call/connections are identified, being an interurban call/connection initiated by a non-MEDIAN end-user or initiated by a MEDIAN end-user, or a local call/connection. The chapter ends with a brief discussion about signalling within the MEDIAN project.

In *Chapter three* the Protocol Reference Model (PRM) of the Broadband Integrated Service Digital Network (B-ISDN) is treated. This B-ISDN PRM is used to express the needed layers (functions) in the different network units within the MEDIAN system. The B-ISDN PRM is subdivided into three layers (physical layer, ATM layer, ATM adaptation layer) which are extensively discussed, and one general layer (describing the higher layers) which is only briefly addressed.

Within the MEDIAN system, optical fibre channels and radio channels are being used. Radio channels are not the issue of the B-ISDN PRM model, because the B-ISDN PRM is developed

by considering the system point and not from the radio point of view. Therefore additional layers are introduced to the B-ISDN PRM to overcome this problem. The needed additional layers are the MAC layer and the interworking layer, which are briefly addressed. The next step taken in the process of developing and implementing interworking functions is the specification of the MEDIAN demonstrator protocol stack.

In *Chapter four* the assumptions and the targets underlaying the development of the MEDIAN interworking layer in the MSS are treated. The approach followed in the process of developing the protocol description is subdivided into four steps. The first step defines the different scenarios in which the MEDIAN system can be operating. In the second step the message flow, concentrating on the internal MEDIAN message flow, is derived using the scenarios resulting from step 1. The third step derives from the different message flows (derived in step 2) a detailed description of the actions which have to be executed in the MSS (concentrating on the actions in the interworking layer). The final step uses the results of step 3 as an input to derived a detailed protocol description of the protocol used in the MSS interworking layer to control the call/connection establishment, maintenance, and release. The final step produces three different protocol description, which together define the protocol of the MSS interworking layer.

In *Chapter five* the assumptions and the targets concerning the development of the MEDIAN interworking layer in the MPS are treated. The MPS is subdivided into a portable radio part (PRP) and an user terminal equipment (UTE). Focusing on the PRP because the UTE which is able to communicate using an optical fibre and ATM already exists. The assumptions made and the targets of the PRP are treated. The approach used in chapter four is also applied in the development of the protocol description of the MEDIAN interworking layer in the MPS.

Ending with *Chapter six* treating the conclusions and recommendations made.

Table of contents

PART 1

PART 2

APPENDIX A: Scenarios of the MEDIAN system

APPENDIX B: Scenario specification of the MEDIAN system

APPENDIX C: Protocol description of the UTE-PRP

APPENDIX D: Protocol description of the MSS-PRP

APPENDIX E: Protocol description of the MSS call release

APPENDIX F: Protocol description of the PRP to MSS

APPENDIX G: Protocol description of the ATM switch to MSS

APPENDIX H: Signalling messages descriptions

List of tables

List of figures

1. General introduction

The merging of computing, measurements, multimedia applications and communication technologies in general is revolutionizing the way people gather and share information. Emerging global networks and providing comfortable wireless communication capabilities will help people to share all kind of information effectively. The data rate requirement of modern commercial wireless communications systems is permanently growing. The multimedia environment and user requirement for high data rates encourage the research, telecommunication, and microwave enterprises to search for new practical solutions. The European bodies decided to support their activities. The transmission frequencies are considered to be placed in millimeter wave band (60 GHz), which was up to days usually used for the military purposes. The worldwide investigations in this field show very first demonstrators and the applicability of the 60 GHz band for high data rate communication.

Present-day telecommunication is characterized by a large number of networks: telephone network, telex network, packet switched network, private network (PABX). The application of these networks is the transportation of voice, data, images (multimedia). There is a need to transport these services (multimedia) using only one transfer method. The solution could be carried out by the introduction of the Asynchronous Transfer Mode (ATM).

The department Telecommunication of the Eindhoven University of Technology (EUT) is involved in research concerning broadband wireless local area networks. This research is one of the main reasons for joining the ACTS project MEDIAN. The aim of the Advanced Communications Technologies and Services (ACTS) project MEDIAN (wireless broadband customer premises network/ local area network for professional and residential multimedia applications) is, among other things, the evaluation and optimalisation of the performance of a wireless local area network (LAN) able to transport multimedia applications (broadband data, sounds and pictures) using the Asynchronous Transfer Mode (ATM).

Properties that discriminate the MEDIAN system from the conventional wireless LAN's are:

- a matchlessly high information transport capacity available to the users (150 Mbit/s total transport capacity);
- the transmission frequencies considered to be placed in the millimeter wave band (around 60 GHz), which was up to days used for military purpose;
- flexibility in the assignment of the network capacity, inherently associated to the usage of ATM;
- interworking with the B-ISDN network (which is based on ATM technology).

One of the responsibility of the EUT within the MEDIAN project is to describe the various interworking functions in the demonstrator scenario (the demonstrator is addressed in this chapter). Another TUE responsibility is the determination of the 60 GHz radio characteristics.

The aim of this report is to describe the interworking functionality's needed in the MEDIAN Server Station (MSS) and the MEDIAN portable station (MPS).

In order to give an objective of the MEDIAN wireless customer premises network different scenarios of application are defined. MEDIAN in its final evolution will be a wireless customer premises network containing all usage and maintenance functions necessary to operate as a local area network, metropolitan area network or as a wide area network.

The MEDIAN system is build-up around three network units:

1. the MEDIAN server station (MSS);
2. the MEDIAN remote station (MRS);
3. the MEDIAN portable station (MPS).

The MSS connects the MEDIAN system with the fixed ATM network via the ATM interface. By using the MRS it is possible to construct, for example, a LAN on a specific floor in a building. The MPS forms the interface between the MEDIAN system and the user.

Local Customer Premises Network (LCPN)

The main application will be a local CPN with different possibilities of configuration. These configurations depend on the user specifics like desired size of area coverage, desired number of simultaneous users and the geographical environment of application. Figure 1-1 illustrates a LCPN application.

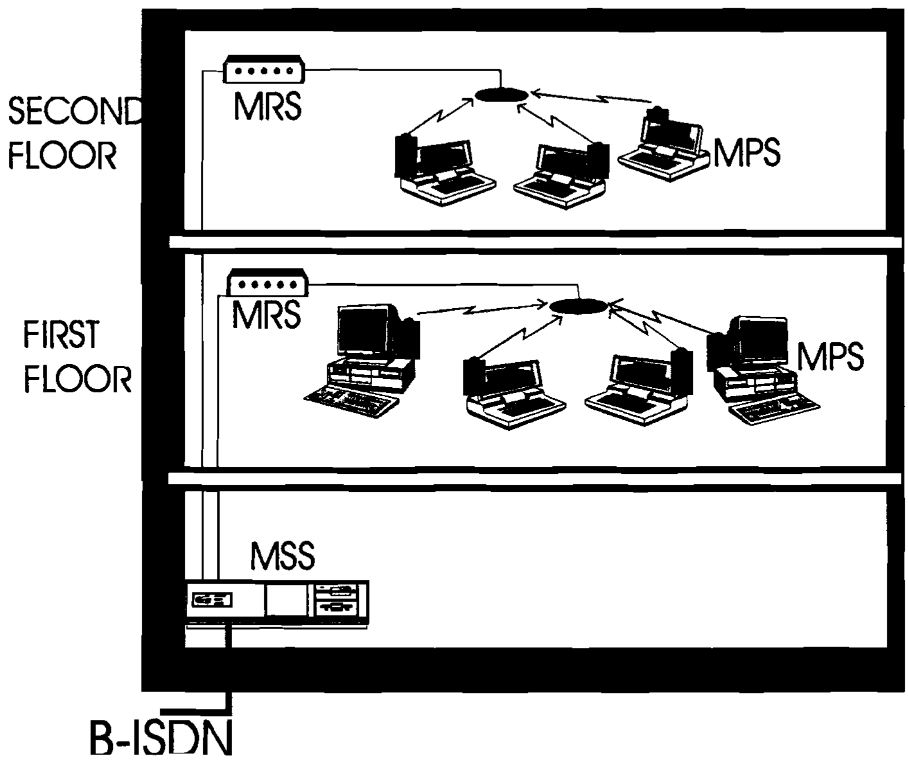


Figure 1-1:Scenarios of application LCPN

Metropolitan CPN (MCPN)

A wide area application of the final MEDIAN system is seen as a metropolitan CPN. Users might be local governments, town-wide spread institutions like public transportation, universities, different business offices of companies. The terms metropolitan and CPN seem to be inherently contradictory. But this is not the case because only limited usage or even no usage of the public switching network with its different operators are necessary to perform a MCPN.

MEDIAN demonstrator

Another, very important to UTE, scenario which needs to be addressed is the scenario of the demonstrator. The MEDIAN demonstrator is a subset of the overall MEDIAN system.

The demonstrator relies on a multicarrier modulation scheme, which is adaptive to transmitted data rates and channel characteristics, and on a wireless ATM network extension. The system, connected to the fixed network via the ATM interface (SONET 155Mbit/s), utilizes the 60 GHz band. The system is built from available components, however, realization studies in the areas of very large scale integration (VLSI) and the 60 GHz technology (MMIC) are included. The demonstrator consists of one MSS and two MPS.

The purpose of the demonstrator is to show the feasibility of the system concept.

Communication within the MEDIAN system always runs via the MSS. The MSS decides whether the call/connection should remain in the MEDIAN system or not. The maximum capacity of one of the MPS is 150 Mbit/s and of the other 30 Mbit/s.

Figure 1-2 depicts the demonstrator scenario.

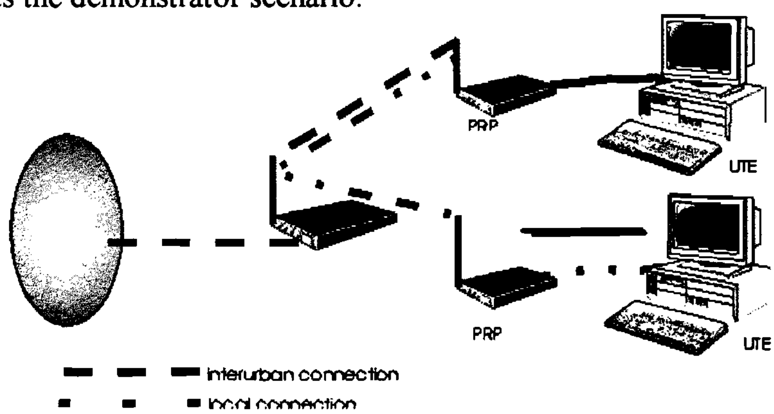


Figure 1-2: Demonstrator scenario

The service trial (demonstrator) will lead to a field of specific results related to the investigation of:

- the necessary quality of service;
- the coverage area of WCPN;
- the applicability of the services to an application/business-oriented environment;
- cost statements of service provision and network operation;
- the definition and specification of the demand on information capacity and bandwidth efficiency in these environments;
- the definition and specification of the demand on service-related data rate requirements in combination with frame and data bit error rates.

Operational requirements
The description of the operational requirements is based on the general needs from user's point of view. Users are application/end users and network maintenance personal.
Operational requirements:
1. MEDIAN should provide all comparable fixed network LAN services;
2. MEDIAN should save costs for the user (due to avoidance of rewiring or even first wiring and due to avoiding the need of an external network operator and/or service provider in case of internal working);
3. MEDIAN should be as reliable as a fixed network (as possible) and be simple in managing (little additional knowledge needed);
4. MEDIAN must provide access to most common networks at present and in future (MEDIAN, UMTS, GSM, MBS, DECT);
5. MEDIAN has to support the Universal Personal Telecommunications (UPT) concept;
6. the MEDIAN system should provide competitive advantages to the user;
7. the MEDIAN system should provide portability of users and/or terminals;
8. MEDIAN system should be able to handle very densely placed terminals;
9. MEDIAN system must provide telecommunication services up to the standardized ATM data rate of 155 Mbit/s;
10. the MEDIAN system may be planned for and has to guarantee a maximum probability of blocking (1% is reasonable);
11. MEDIAN has to provide network management functions to the super-user in order to maintain the system data base, the security management, fault handling and system resources.

2. Signalling

2.1 Introduction

The signalling which will be used corresponds to the ATM user-network interface (UNI) specification version 3.1 of the ATM Forum (compatible with Q.2931).
While B-ISDN is a definition for public networks, ATM can also be used within private networking products. In recognition of this fact and of clarity, two distinct forms of ATM UNI are defined:

1. Public UNI - which will typically be used to interconnect an ATM user with an ATM switch deployed in a public service provider's network;
2. Private UNI - which will typically be used to interconnect an ATM user with an ATM switch that is managed as part of the same corporate network.

The primary distinction between these two classes of UNI is physical reach. There are also some functionality differences between the public and the private UNI due to the applicable requirements associated with each of these interfaces.

Signalling messages will be conveyed out-of-band in dedicated **signalling virtual channels** (SVCs). Possible signalling virtual channels at B-ISDN user-network interface are listed in Table 2-1.

Table 2-1: SVC at B-ISDN UNI

Meta signalling channel	Bi-directional	1
General broadcast SVC	Unidirectional	1
Selective broadcast SVC	Unidirectional	Several possible
Point-to-point	Bi-directional	One per signaling endpoint

Signalling messages are used to control calls/connections. Control means establishing, maintaining and releasing of a call/connection. In the following section the different call states are addressed. This is followed by the different messages which can be used to provide the needed signalling. After a small introduction of the theoretical background of signalling (states, messages) three different call/connection types concerning the MEDIAN system are addressed. At the end of the Section 2.2, the three phases a connection can be in (establishment, maintenance, released are briefly discussed and related to the previous section.

2.2 ATM call states

2.2.1 Introduction

All the states are viewed from the users point of view. This implies, that a connection is in an outgoing state if the message is sent from user to network. A connection is in an incoming state if the user receives an message from the network.

2.2.2 Null (U0)

This state exists when no call exists. At start-up of the station all the calls/connections start in the null state.

2.2.3 Call initiated (U1)

This state exists for an outgoing call when the user requests call establishment from the network. When a user decides to send a call/connection setup message to the network, then the state of the call/connection is changed into call initiated.

2.2.4 Outgoing call proceeding (U3)

This state exists for an outgoing call when the user has received acknowledgement that the network has received all call/connection information to effect call/connection establishment. Some examples of call/connection information that could be of interest are the quality of service, maximum capacity uplink and downlink, constant bit rate or variable bitrate, timing requirements etc.

2.2.5 Call present (U6)

This state exists for an incoming call/connection when the user has received a call establishment request but has not yet responded.

2.2.6 Connect request (U8)

This state exists for an incoming call when the user has answered the call and is waiting to be awarded the call. After receiptance of the call/connection establishment request the user sends a message indicating that the establishment request is accepted to the network. At this moment the call/connection is in the connect request.

2.2.7 Incoming call proceeding (U9)

This state exists for an incoming call when the user has sent acknowledgement that the user has received all call/connection information necessary to effect call/connection establishment.

2.2.8 Active (U10)

This state exists for an incoming call when the user has received an acknowledgement from the network that the user has been awarded the call. This state exists for an outgoing call when the user has received an indication that the remote user has answered the call.

2.2.9 Release request (U11)

This state exists when the user has requested the network to clear the end-to-end connection (if any) and is waiting for a response.

2.2.10Release indication (U12)

This state exists when the user has received an invitation to disconnect because the network has disconnected the end-to-end connection (if any).

2.3 Message functional definitions and contents

2.3.1 Introduction

Within the protocol, every signalling message consists of the following parts:
1. protocol discriminator;
2. call reference;
3. message type;
4. message length;
5. variable length information elements, as required.

Information elements 1, 2, 3, and 4 are common to all the signalling messages and shall always be present, while information element 5 is specific to each message type. Because of the commonality of the first four parts of every message, attention is focused on the variable length information elements of the specific messages.
A particular message may contain more information than a particular (user or network) equipment needs or can understand. All equipment shall be able to ignore any extra information present in a message, which is not required for the proper operation of that equipment.

2.3.2 SETUP

This message is sent by the calling user requesting the establishment of a call/connection. The call/connection is characterized by a number of fields which are included in the SETUP message.
At least the following information needs to be present in the variable length information element of the SETUP message:
- ATM traffic descriptor, defining the traffic characteristics;
- broadband bearer capability, indicating a requested broadband connection oriented bearer service;
- called party number, uniquely identifying the called party;
- quality of service, defining the service class which is requested.

2.3.3 CALL PROCEEDING

This message is sent by the called user to the network or by the network to the calling user to indicate that the requested call establishment has been initiated and no more call establishment information will be accepted. The sending of this message is optional but receiptance is required.
The additional information fields which could be included are:
- connection identifier, locally identifying the call/connection;
- endpoint reference, only needed in point-to-multipoint calls/connections (not applicable in the MEDIAN demonstrator).

2.3.4 CONNECT

This message is sent by the called user to the network and by the network to the calling user to indicate call acceptance by the called user.
The additional information elements of the variable length information element are:

- AAL parameters, encapsulating information concerning the AAL type (1, 3/4, 5 or user-defined), CBR rate, error correction method
- Broadband low layer information, purpose is to provide a means which should be used for compatibility checking by an addressed entity;
- Connection identifier, it is mandatory in the network-to-user direction if this message is the first message in response to a SETUP message. It is mandatory in the user-to-network direction if this message is the first message in response to a SETUP message, unless the user accepts the connection identifier indicated in the SETUP message;
- Endpoint reference, is mandatory if the endpoint reference was included in the SETUP message.

2.3.5 CONNECT ACKNOWLEDGE

This message is sent by the network to the called user to indicate the user has been awarded the call. It is also sent by the calling user to the network to allow symmetrical call control procedures. There is no possibility to add extra information elements to the connect acknowledge message.

2.3.6 RELEASE

This message is sent by the user to request the network to clear the end-to-end connection (if any) or is sent by the network to indicate that the en-to-end connection is cleared and that the receiving equipment should release the virtual channel and prepare to release the call reference after sending a RELEASE COMPLETE. When this message is sent a additional Cause field must be added. The cause field indicates the reason why the connection will be released, the length of the cause field is in the range between 6 and 34 octets.

2.3.7 RELEASE COMPLETE

This message is sent by the user or the network to indicate that the equipment sending the message has released the virtual channel and call reference, the virtual channel is available for reuse, and the receiving equipment shall release the call reference. When this message is sent a additional Cause field must be added. The cause field indicates the reason why the connection will be released, the length of the cause field is in the range between 4 and 34 octets. The adding of the cause information element is mandatory in the first call clearing message; including when the RELEASE COMPLETE is sent as a result of an error condition.

2.3.8 STATUS ENQUIRY

The status enquiry message is sent by the user or the network at any time to solicit a STATUS message from the peer layer 3 entity. Besides the four usual information elements the use of the endpoint reference information element is optional.

2.3.9 STATUS

This message is sent by the user or the network in response to a STATUS ENQUIRY message or at any time to report certain error conditions. Two additional fields are mandatory and two are optional. The two mandatory fields are the call state and the cause field.

2.4 Signalling ATM adaptation layer

This paragraph specifies the signalling ATM adaptation layer (SAAL) for use at the UNI interface. The SAAL resides between the ATM layer and Q.2931 layer. The purpose of the SAAL is to provide reliable transport of Q.2931 messages between peer Q.2931 entities over the ATM layer. The SAAL is composed of two sublayers, a common part and a service specific part.

The service specific part is further subdivided into a service specific co-ordination function (SSCF), and a service specific connection oriented protocol (SSCOP).
Figure 2-1 illustrates the structure of the SAAL.

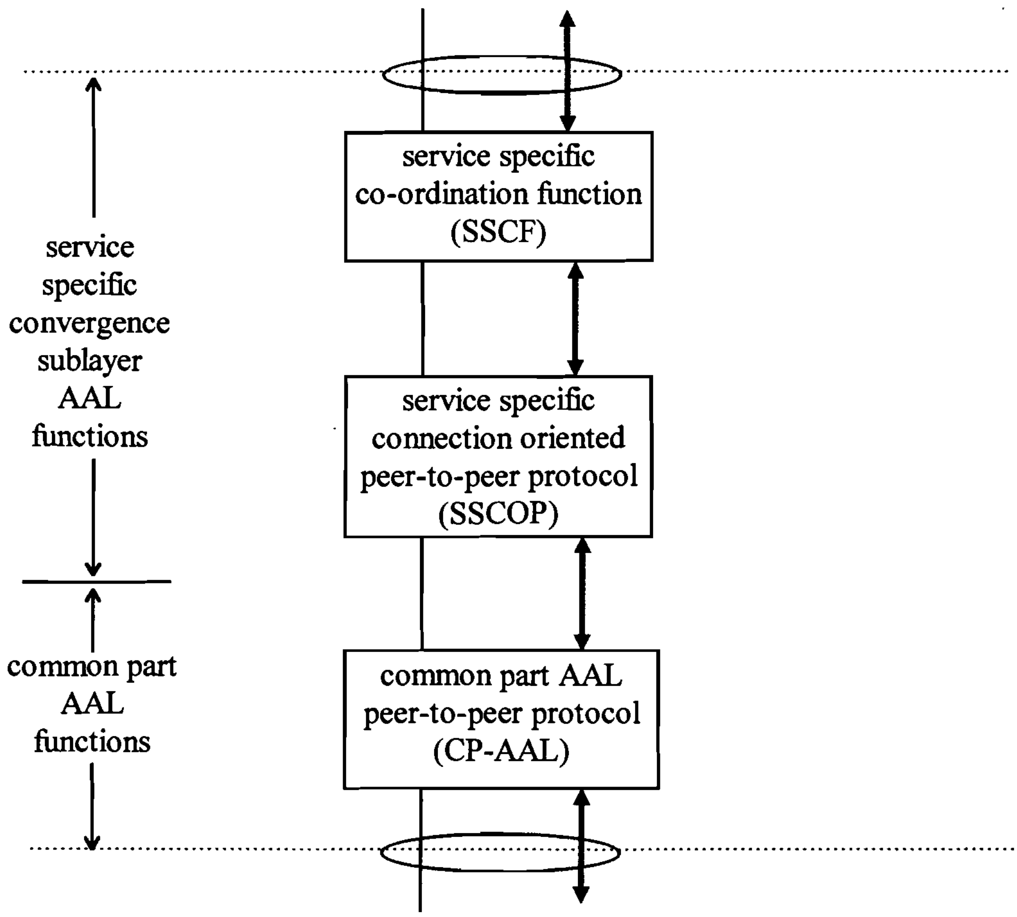


Figure 2-1: structure of the SAAL

The common part AAL protocol provides unassured information transfer and a mechanism for detecting corruption of service data units (SDUs). AAL type 5 common part protocol shall be used in supporting the signalling.
The SSCOP resides in the service specific convergence sublayer (SSCS) of the SAAL. SSCOP is used to transfer variable length SDUs between users of SSCOP. SSCOP provides for the recovery of lost or corrupted SDUs.

A SSCF maps the service of SSCOP to the needs of the SSCF user. Different SSCFs may be defined to support the needs of different AAL users

2.5 Three different call/connection types

2.5.1 Introduction

With respect to a MEDIAN LAN it is desirable to establish two different types of calls/connections: a call/connection established between two MEDIAN users, or a call/ connection established between a MEDIAN user and a non MEDIAN user. We have to distinguish between these two types of calls because, for example, the MSS needs to know if the call which is requested to be established remains in the MEDIAN system or not. It is also very important, if we want to make a specific network which needs to be profitable, to comply with one of the central issues: to have compatibility with users outside the specific network to increase the feasibility of the specific network.
Distinction must be made between the calling party (initiates the call/connection) and the called party. When the calling party is a MEDIAN user then the network needs to determine if the called party indicated in the call/connection establishment request is a MEDIAN user or not.
Summarizing, we have the following three call/connection types:

- an interurban call/connection initiated by a MEDIAN user;
- an interurban call/connection initiated by a non-MEDIAN user;
- a connection between two users of the same MEDIAN LAN (local call/connection).

The next paragraphs describes, in more detail, the previous made separation in call/connection types. A call/connection which remains within the MEDIAN system is called a local call/connection, and a call/connection which does not remain within the MEDIAN system an interurban call/connection.

2.5.2 An interurban call/connection initiated by a non-MEDIAN user

The MSS receives a request to establish an interurban connection with one of its users (MEDIAN users) from the public network (ATM switch) end of the MSS. The MSS is involved in the following actions concerning the call/connection establishment:

- recognition of received message;
- deduction of call/connection characterizing parameters, like: peak bitrate; timing requirements; quality of service class;
- deduction/checking of called party address;
- checking if the MEDIAN system is able to guarantee the requested characteristics specified;
- to accept/reject call/connection establishment;

The MSS decides to accept the call/connection establishment whenever there are enough resources available to guarantee the requested call/connection characteristics (the opposite implies for the call/connection rejection).

In Chapter 4 a more detailed description is given of the actions that the MSS takes on receiptance of a call establishment request.

The MSS is also involved in a number of other actions, such as call/connection maintenance and call/connection release.

2.5.3 A interurban call/connection initiated by a MEDIAN user

In general, there is not much of a difference between this type of call/connection and the one described in the previous section. The difference is found in the side that receives the establishment of the call/connection. In this situation the call/connection is received from the MEDIAN network side. The MSS notices that the calling party is an non-MEDIAN user.
The MSS decides the accept the call/connection establishment whenever there are enough resources available to guarantee the requested call/connection characteristics (the opposite implies for the call/connection rejectance).

See Chapter 4 section 4.4.2 for a more detailed description of the action which the MSS takes on receiptance of a call establishment request.

2.5.4 A local call/connection

This call/connection establishment exists whenever the called party and the calling party are both registered as being MEDIAN users in the MSS. All the actions taken, and messages sent to establish/maintain/release the local call/connection remain within the MEDIAN system. Not much difference is noticed compared to the interurban call/connection establishment initiated by a MEDIAN user. One of the differences is, that the capacity available must at least be equal to two times the requested capacity (equal to the sum of the peak capacity in the forward (uplink) and the peak capacity in the backward (downlink)).

Continuing with a description of the three phases of a call/connection.

2.6 Call/connection phases

2.6.1 Introduction

As briefly mentioned above, a call/connection begins with the call establishment phase. There are two other phases defined, being the maintenance and the release phase. As the name implies, the maintenance phase is the phase where the MSS maintains the call/connection. Maintenance actions, for example, are deriving and reporting the status, monitoring of the call/connection, monitoring if actual traffic complies with the contract. In the release phase the existing call/connection is released, meaning that all the reservations made and used in the call establishment phase and the maintenance phase are released. The resources released, can be (re)used to establish a new call/connection.

In the next three sections the actions the MSS takes whenever it is in the specific phase are described.

2.6.2 Call/connection establishment

Before the MSS is able to determine whether it wants to accept the call/connection it first has to receive all the information from the calling party. This information is maintained in the SETUP message sent by the calling party to the MSS.
Every message contains the call reference information element (as described in Section 2.3).
One of the elements of the call reference information element is the call reference value. The call reference value is used to uniquely identify the call/connection at the local user-network interface to which the particular message applies. The call reference value does not have end-to-end significance across ATM networks. The call reference value is used to address a static

table. The static table maintains all the necessary information needed to maintain/release the call/connection which has been established. The static table is called a static because its contents does not change between the time that a call/connection is established and the time that it is released. A detailed description of the contents of the static table is included in the chapters describing the MEDIAN protocols (section 4 and 5). The third information element is the message type information element which identifies the type of the received message (SETUP, CALL PROCEEDING, ..., RELEASE COMPLETE). As mentioned earlier, contains the SETUP message all the information needed to decide between call/connection acceptance or rejectance. In the call/connection establishment phase the following messages are used:

- SETUP;
- CALL PROCEEDING;
- CONNECT;
- CONNECT ACKNOWLEDGE;
- RELEASE COMPLETE.

2.6.3 Call/connection maintenance

After a call/connection has been established, the actual transfer of user data (data, voice, picture or combinations) can take place. The MSS is involved in actions which, for example, guarantee that the accepted quality of service, resources, broadband bearer services are available to the specific call/connection. In the call/connection maintenance the following messages are used:

- STATUS;
- STATUS ENQUIRY.

Concerning the transportation of the user date the MSS needs to:

- map the incoming VPI/VCI on a timeslot, in case of user data arriving from the public network side of the MSS in an interurban call/connection;
- map the incoming timeslot arriving in a uplink timeslot on a downlink timeslot, in case of a local call/connection when data is transmitted from one MPS to the other;
- no mapping is needed in an interurban call/connection when user-data arrives from the MPS;

2.6.4 Call/connection release

First of all a distinction has been made between the stations that are able to release a call/connection in the peer-to-peer user connection. These stations are the MPS (in later stage not seen as a release candidate and the MSS in the MEDIAN system, and the public network and the non-MEDIAN user. The messages which are involved in the call/connection releasing are:

- RELEASE;
- RELEASE COMPLETE;

In the RELEASE and/or RELEASE COMPLETE message a cause field is included. The cause information field describes the reason for generating this message, provides diagnostic information in the event of procedural errors, and indicates the location of the cause originator. After successfully releasing the call/connection the resources are available for re-use in other call/connections.

3. Protocol stack

3.1 Introduction

The protocol stack is used to identify the needed layer functions within certain network units. The introduced B-ISDN protocol reference model can be used to refer certain functions to standardized layer by the International Standardization Organization (ISO).

3.2 B-ISDN protocol reference model (PRM)

Figure 3-1 shows the B-ISDN PRM, which consists of three planes:
- user plane;
- control plane;
- management plane.

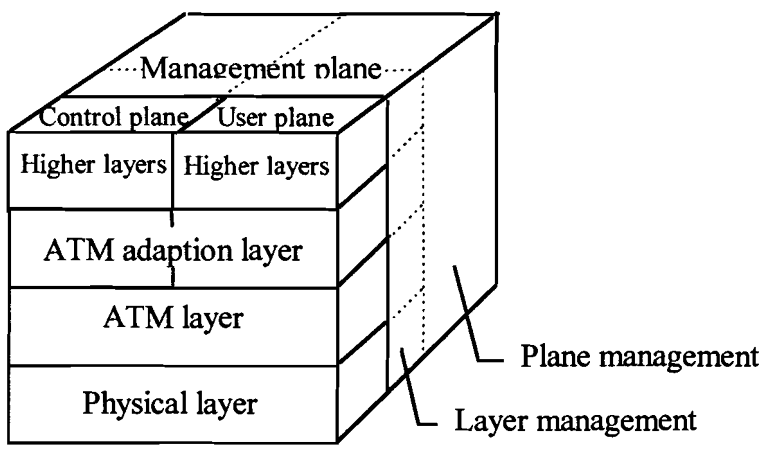


Figure 3-1: B-ISDN protocol reference model

The *management plane* includes two types of functions, called:
1. layer management functions and
2. plane management functions.

Layer management has a layered structure. It performs the management functions relating to resources and parameter residing in its protocol entities. Layer management handles the specific OAM information flows for each layer.

Plane management is responsible for the management of the different planes (user plane, control plane, management plane).

The *user plane* provides the transfer of user information. All associated mechanism, like flow control and recovery from errors are included.

The *control plane* is responsible for the call control and connection control functions. These are all signalling functions which are necessary to set up, supervise and release a call connection.

The protocol reference model is used to identify functionality's (services) needed in the specific network units within the framestructure of the ATM Forum standardization. If, for example, VPI/VCI translation of ATM cells is needed then at least the ATM layer (or a subset of the functionality's of it) needs to be implemented. The information exchange between the adjacent layers makes use of, and is controlled by, primitives. A service is formally specified by primitives which are available to the entity[1] which has access to the service.
There are four different classes of primitives defined, see Table 3-1.

Table 3-1: primitives

primitive	meaning
request	entity asks to execute a service
indication	entity needs to be informed about event
response	entity wants to react to an event
confirm	entity needs information concerning the request

The used ATM Forum 3.1 specification supports two of the above described primitives. Being the request and the indication primitive.

[1]An entity is an active element within a layer.

Every layer is characterized by a number of functions available in that specific layer. Figure 3-2 shows the lower layers of the B-ISDN PRM and their functions.

L A Y E R M A N A G E M E N T	Higher layer functions	Higher layers	
	Convergence	CS	AAL
	Segmentation and reassembly	SAR	
	Generic flow control Cell header generation/extraction Cell VPI/VCI translation Cell multiplex and demultiplex	ATM	
	Cell rate decoupling HEC sequence generation/verification Cell delineation Transmission frame adaption Transmission frame generation/recovery	TC	PL
	Bit timing Physical medium	PM	

AAL	ATM adaption layer	PM	Physical medium
ATM	Asynchronous transfer mode	SAR	Segmentation and reassembly
CS	Convergence sublayer	TC	Transmission convergence
HEC	Header error control	VCI	Virtual channel identifier
PL	Physical layer	VPI	Virtual path identifier

Figure 3-2: Functions in the protocol reference model

In the following paragraphs the different layers of the protocol reference model are addressed.

3.2.1 Physical layer functions

The physical layer is subdivided into the:
- physical medium (PM) sublayer and
- transmission convergence (TC) sublayer.

The PM sublayer provides the bit transmission capability including bit alignment. Line coding and , if necessary, electrical/optical conversion is performed by this sublayer.

Bit timing functions are the generation and reception of waveforms which are suitable for the medium, insertion and extraction of bit timing information, and line coding if required.

The TC sublayer performs five functions:
1. transmission frame generation/recovery;
2. transmission frame adaption;
3. cell delineation;
4. HEC sequence generation/extraction;
5. cell rate decoupling.

Transmission frame adaption is responsible for all actions which are necessary to adapt the cell flow according to the used payload structure of the transmission system in the sending direction. In the opposite direction it extracts the cell flow from the transmission frame.
The functions mentioned so far are specific to the transmission frame.
Cell delineation is the mechanism that enables the receiver to recover the cell boundaries. To protect the cell delineation mechanism from malicious attack, the information field of a cell is scrambled before transmission. Descrambling is performed at the receiver side.
HEC sequence generation is done in the transmit direction. The HEC sequence is inserted in its appropriate field within the header.
In the sending direction, the **cell rate decoupling** mechanism inserts idle cells (containing no information) in order to adapt the rate of ATM cells to the payload capacity of the transmission system. In the receiving direction this mechanism suppresses all idle cells. Only assigned (provides a service to an application using the ATM layer service) and unassigned cells (not an assigned cell) are passed to the ATM layer.

3.2.2 ATM layer functions

Four functions of this layer have been identified:
1. cell multiplex and demultiplex;
2. cell VPI/VCI translation;
3. cell header generation/extraction;
4. generic flow control.

In the transmit direction, cells from individual VPs and VCs are multiplexed into one resulting cell stream by the **cell multiplexing** functions. The composite stream is normally a non-continuous cell flow. At the receiving side the **cell demultiplexing** functions splits the arriving cell stream into individual cell flows appropriate to the VP or VC.

VPI and VCI translation are performed at ATM switching nodes and/or at cross-connect nodes. Within a VP node the value of the VPI field of each incoming cell is translated into a new VPI value for the outgoing cell. The values of the VPI and VCI are translated into new values at a VC switch.

The **cell header generation/extraction** function is applied at the termination points of the ATM layer. In the transmit direction, after receiving the cell information field from the AAL, the cell header generation adds the appropriate ATM cell header except for the HEC values. VPI/VCI values could be obtained by translation from the SAP identifier. In the opposite direction, the cell header extraction functions removes the cell header. Only the cell information field is passed to the AAL. This function could also translate a VPI/VCI into a SAP identifier.

The generic flow control (GFC) function is only defined at the B-ISDN UNI (user-network interface). GFC supports control of the ATM traffic flow in a customer network.

3.2.3 ATM adaptation layer functions

The AAL is subdivided into the:

- segmentation and reassembly (SAR) and
- convergence sublayer (CS).

Its basic function is the enhanced adaption of services provided by the ATM layer to the requirements of the higher layers. Higher layer PDUs (Protocol Data Units) are mapped into the information field of an ATM cell. AAL entities exchange information with their peer AAL entities to support AAL functions

AAL functions are organized in two sublayers. The essential functions of the SAR sublayer are, at the transmitting side, segmentation of higher layer PDUs into a suitable size for the information field of the ATM cell and, at the receiving side, reassembly of the particular information fields into higher layer PDUs.

3.2.4 Higher layers

The higher layers corresponding to OSI model are:

- the transport layer;
- the session layer;
- the presentation layer;
- the application layer;

The main function of the transport layer are: the acceptance of data from the session layer; segment (if needed) this information in smaller packets; deliver the packets to the network layer (AAL layer); take care of reliable transportation of the packets to the destination. This has to be done on a efficient way, which isolates the session layer from changes in the hardware (lower layers). The transportation layer is an end-to-end layer, meaning that there is only communicates between source and destination (calling and called party).

The session layer permits users on different stations, to establish a common session. An example of a session service is the login on a time-sharing system.

The presentation layer executes certain functions which will be used frequently. A good example of a service delivered by the presentation layer is coding of data. Most of the user programms exchange information like names, dates, accounts, data. These information sources can be presented in many ways (like characters, floating point numbers, integers, data structures, etc.). Different computers use different representations of the information. To make it possible that computers with different representations are able to communicate with each other, an abstract way of coding is used for all the different information resources.

The application layer contains a number of frequently used protocols. For example, there are hundreds of incompatible types of computers. The application layer introduces a virtual terminal, which can be used by different programs.

There will be no additional discussions about the higher layers in this report because they are not relevant for the MEDIAN demonstrator part to be implemented.

3.3 Additional layers

3.3.1 Introduction

Networks can be subdivided in point-to-point and point-to-multipoint networks. The MEDIAN demonstrator is a point-to-multipoint network (except when only one user is in call) and hence a mechanism is needed to control the access of the users to the channel. This mechanism is located in a so called Medium Access Control (MAC) layer.
Within the MEDIAN consortium it was not clear which functionality's should be implemented (are needed) on top of the MAC layer in the MSS and in the MPS. We had to define these functionality's in a so called interworking layer.
The functionality's which need to be implemented in MAC and/or interworking layer of the MSS and the MPS differ from each other. In the following two sections, a short introduction is made on the MAC layer and the interworking layer with no differentiation into MSS and MPS. This differentiation takes places in Section 1.2 specifying the MSS and MPS interworking layer.

3.3.2 MAC layer

From the MAC point of view, a WLAN environment yields the following problems:

- in typical B-ISDN services high variance of the information rate is expected. This implies the necessity of a dynamic rearrangement of the resources-to-calls assignment, since, in resulting case the available resources were fixedly partitioned among the various calls, the system could be highly inefficient;
- in order to increase capacity, the use of small coverage areas is required. However, small coverage areas are subject to extremely unpredictable temporal and geographical variations in traffic density; this means that, in the case the available resources are fixedly partitioned among the various areas, the system could result highly inefficient.

The proposed MAC layer uses a transmission frame with constant length. The specification of the transmission frame is subdivided into an uplinkframe and a downlink frame. The duration of both the uplink and downlink frame can be varied to fit the present traffic characteristics (dynamically changed) to improve the efficiency. The uplink and downlink are consecutively transmitted resulting in a time division duplex (TDD) access technique.

Each frame consists of cells with length equal to the length of an *extended cell.* An extended cell consists of an ATM cell and additional MEDIAN overhead. The MEDIAN overhead consists, among other things, of a MEDIAN call/connection identifier and coding needed to improve the poor radio-channel characteristics. The extended cell is transported on the radio-interface in an extended slot. The identification of the allocation of the extended cells to the MPSs is given in the broadcast cell. The broadcast cell is transmitted on a fixed place in every uplink-frame. The presence of the downlink broadcast cell, i.e. extended cell broadcasted by the MSS (or MRS) towards all the served MPSs, is a key issue for the correct system way of working.

Figure 3-3 illustrates the reference terminology.

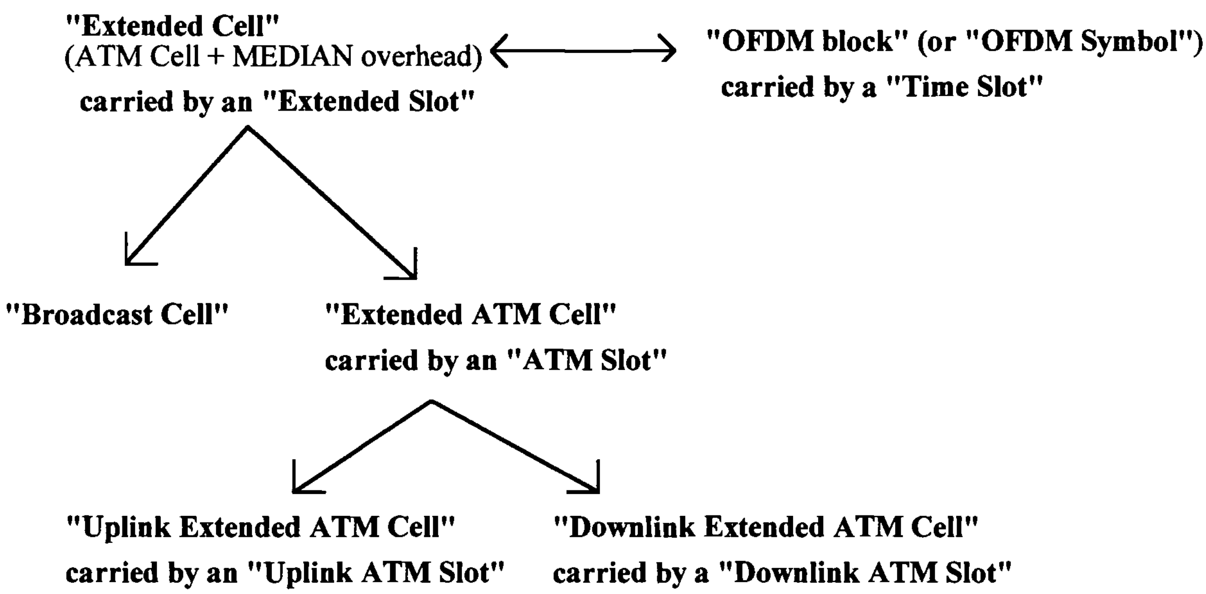


Figure 3-3: Reference terminology

The information to be carried by the broadcast cell contains:

- synchronization help for the MPS;
- information, necessary to address the called MPS in case of fixed network initiated calls;
- information concerning the duration of the TDD, uplink and downlink frame duration;
- information concerning the extended cell time slot devoted to either to random access or to reservation access in the framework of a packet reservation multiple access (PRMA) technique;

Figure 3-4 illustrates the example of the TDD access technique.

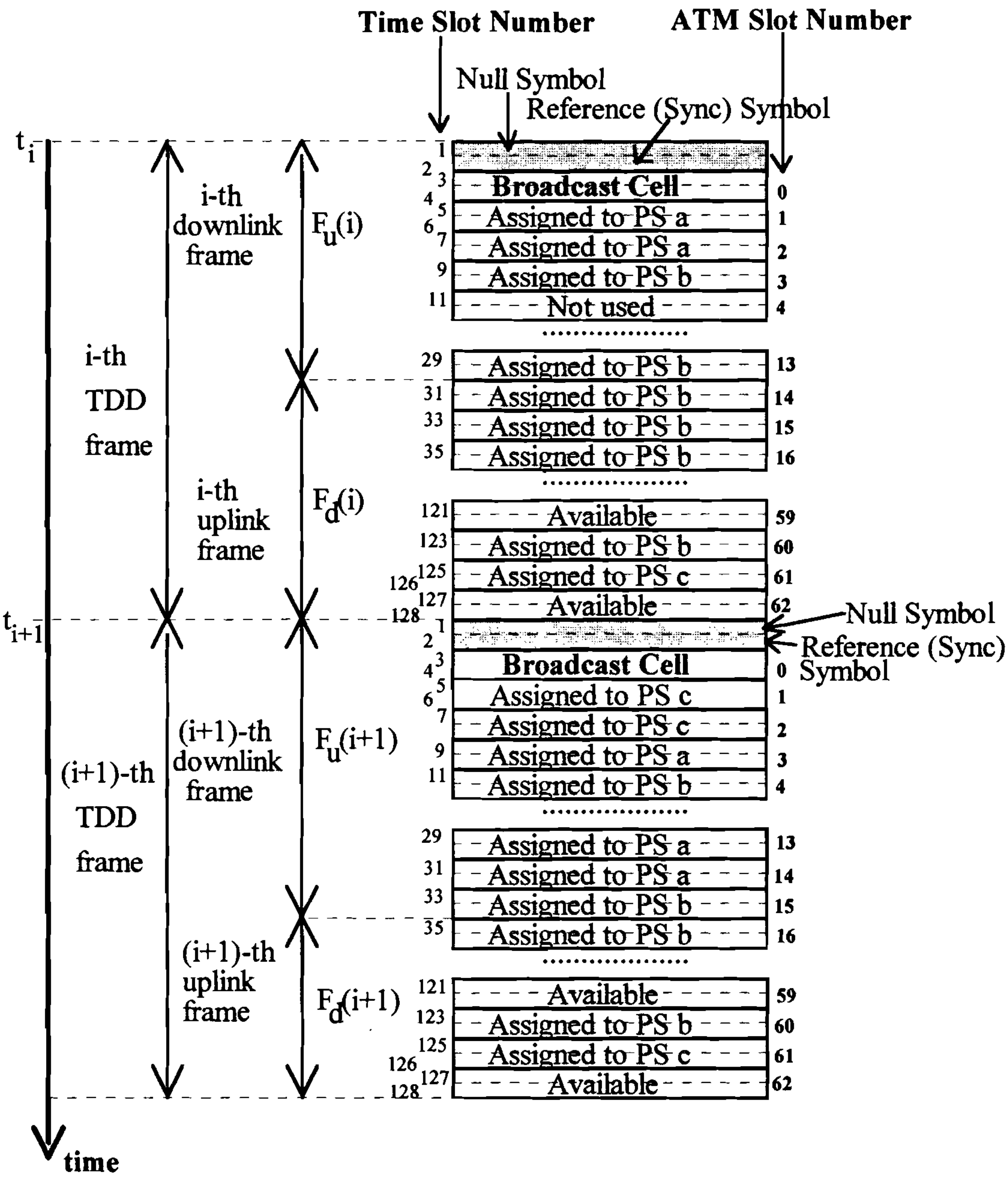
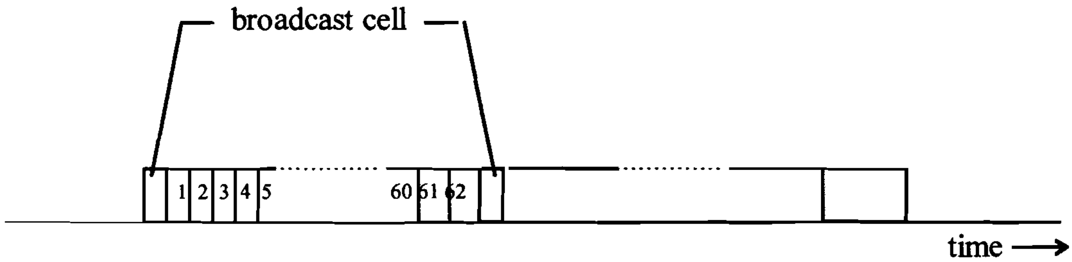


Figure 3-4: TDD access technique example

A problem arises concerning the reading of the broadcast cell, which contains the slot reservation list. The MPS needs some time to read this broadcast cell and to deduce which timeslot is dedicated to it. The problem is solved on a simple way. Suppose that the maximum time needed by a MPS to read the broadcast cell equals two timeslots. Then the first timeslot in the slot reservation list refers to the third timeslot after the broadcast slot, and not the first.

The example in Figure 3-5 illustrates this idea.



slot reservation list	
slot number	reserved to
3	MPS1
4	free
5	MPS3
.	.
61	MPS15
62	free
1	MPS1
2	MPS2

Figure 3-5: displacement of the slot reservation list

The time needed to read/process the broadcast cell is eliminated by the shift in the slot reservation list. The shift is proportional to the time the 'slowest' MPS needs to read/process the broadcast cell.

More generally, the MAC is responsible for:
- mapping of logical to physical channels
- distribution of access;
- collision avoidance.

3.3.3 Interworking layer

The position of the interworking layer is always above the MAC layer
As mentioned earlier vagueness exists about the functionality's which need to be implemented in the interworking layer. Listed are a number of actions which could be implemented in the interworking layer.

- produce/deliver all the information the MAC needs to function properly;
- call control:
  - call/connection establishment;
  - call/connection monitoring (maintenance);
  - call/connection release;
- generation of address information;
- generation of call/connection internal identifier;

The protocols that are developed in Section 4 and 5 are implemented in the Interworking-layer.

3.4MEDIAN demonstrator protocol stack

3.4.1 Introduction

In the previous sections of Chapter 3, the different layers positioned in the protocol stack were described. In this section we continue with the specific protocol stack of the MEDIAN demonstrator.
The overall protocol stack is subdivided into an user and a control plane protocol stack. In the following sections the user plane protocol stack and the control plane protocol stack are described belonging to the MSS, the MPS and the entire MEDIAN demonstrator.

Note, that we are considering the MEDIAN demonstrator and not the overall MEDIAN system! Like described earlier, the MEDIAN demonstrator consists of two different stations:

- the MSS, and
- the MPS (two MPSs);

Figure 3-6 depicts the MEDIAN demonstrator.

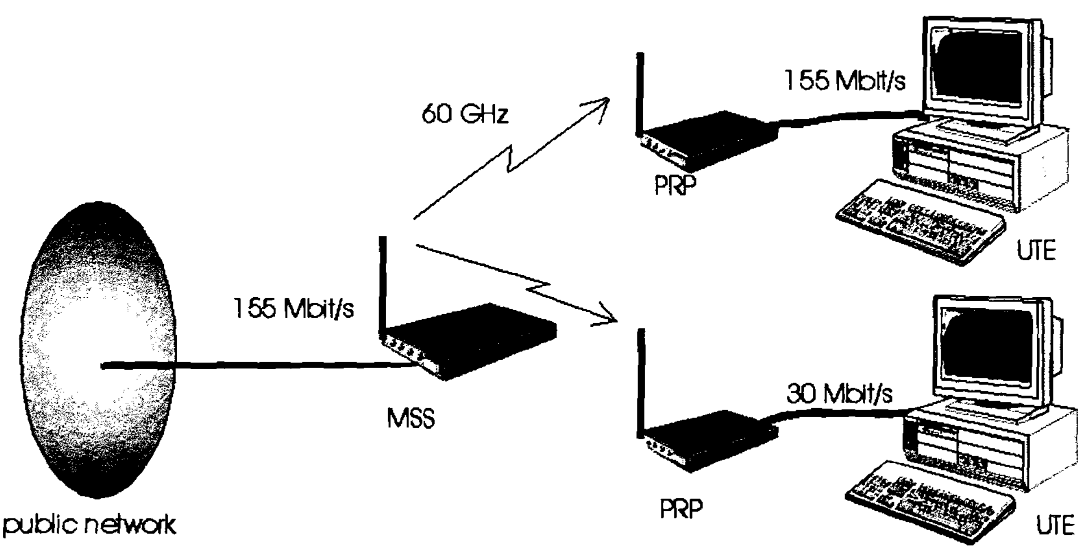


Figure 3-6: the MEDIAN demonstrator

3.4.2 MSS user plane protocol stack

The MSS user plane protocol stack describes the different layers needed in the MSS to handle the user data between

- two MPSs;
- MPS and non-MEDIAN user.

In the public network direction the MSS handles the ATM cells coming from the public network. The route that the ATM cells have to travel within the MEDIAN system is known because ATM is a connection oriented technology, meaning that the route is assigned during call setup. Every call/connection is identified by an unique VPI/VCI combination. The MSS is aware of a mapping of the VPI/VCI on an unique internal identifier specifying the timeslot reserved in the downlink frame.

In the MSS towards MPS direction the MSS handles ATM cells coming from the public network or from another MPS. If the MSS receives an user data cell from the MPS in a dedicated timeslot (identified by an unique internal identifier, called Radio Virtual Channel Identifier(RCVI)), then the MSS maps the RVCI belonging to this call/connection on another dedicated (reserved) RVCI, used to specify the timeslot in the downlink frame. If the MSS receives a user data cell from the public network, then it maps the VPI/VCI on a dedicated (reserved) internal identifier specifying the downlink timeslot.
The channel between the MSS and the MPS is a 60 GHz radio channel. Because of the higher bit error rate (BER) on a radio channel when comparing to the reliable optical channel (where ATM is developed for) additional error control functionality's need to be implemented in the MSS to reduce the negative consequences of the usage of a radio channel. The first improvement of the BER on the radio channel is achieved by using a forward error correction (FEC) coding mechanism in the physical layer[2]. The prediction[3] is that the developed coding decreases the BER from approximately 10^{-4} to 10^{-7}. Before we are able to determine the BER more accurately, a detailed description of the 60 GHz radio channel is needed.
The MEDIAN system should be able to provide a high class of quality of service . The possibility exists that the BER of 10^{-7} is still not good enough or may not be reached in practice. Another way of improving the error rate is to make use of retransmission (if necessary). A trade-off has to be made between the number of bits added in the FEC coding and the expected number of retransmission (FEC always uses a partition of the available capacity in contrast to the retransmissions which use temporary more capacity). The retransmission function is executed by a data link control (DLC) layer.
ATM networks are usually architected to provide the performance guarantees required by constant bit rate (CBR) and variable bit rate (VBR) traffic and allow available bit rate (ABR) traffic to use the remaining bandwidth (capacity).

[2]this coding has to be done on the lowest possible layer, because if this coding takes place on a higher layer then the additional layer overhead of the lower layers is not protected by this coding.
[3]done by one of the consortium partners

Table 3-1 summarizes the three types of traffic and their network requirements

Table 3-1: summary of network traffic types and their requirements

traffic type	example	bandwidth required	cell delay variation	latency
constant	voice, circuit emulation	guaranteed	minimal	low
variable	compressed video	guaranteed	variable	low
available	data	not guaranteed	wide variation	moderate to high

The Figure 3-7 illustrates the link usage by the above defined different traffic types.

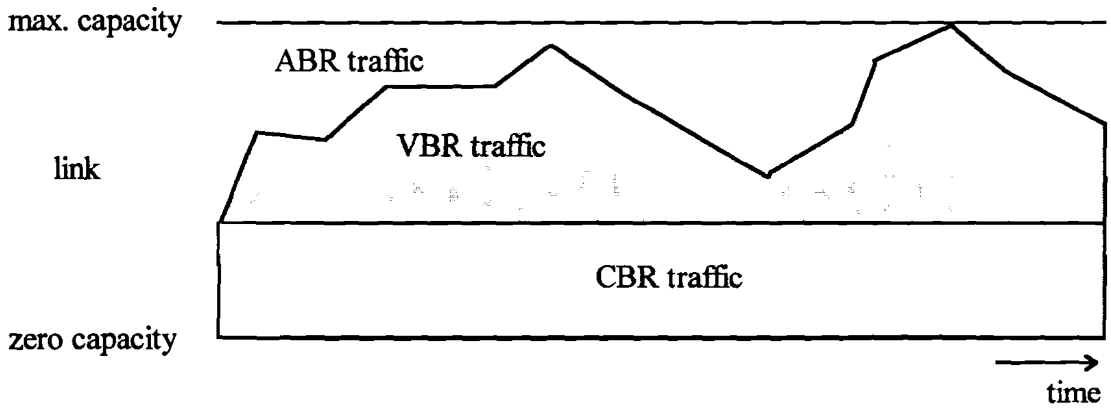


Figure 3-7: link usage by different traffic types

The type of AAL strongly depends on the service (traffic type) it has to support. There are four different AAL types defined, being the:

- AAL type 1: normally, CBR. Timing information is also transferred between source and destination. Indication of lost or errored information is sent to the higher layer if these failures cannot be recovered within the AAL;
- AAL type 2: proposed for VBR with a timing relation between source and destination. If needed, the higher layers can be informed about errors which cannot be corrected by the AAL;
- AAL type 3/4: two modes of services are defined: message mode service which can be used for framed data transfer (HDLC frame) and streaming mode service suitable for the transfer of low-speed data with low delay requirements. Two peer-to-peer operation procedures are offered by both service modes:
    1. assured operation: the assured operation retransmits missing or error AAL-SDUs, therefore flow control is provided as a mandatory feature;
    2. non-assured operation: lost or errored AAL-SDUs are not corrected by retransmission. The delivery of corrupted AAL-SDUs to the user may be provided as in optional feature.
- AAL type 5: will be applied to VBR sources without timing relation between the source and destination. AAL type 5 will be used for signalling and frame relay over ATM

AAL type 3/4 and AAL type 5 both support error recovery by retransmission, meaning that the user can rely on a (more or less) errorless transportation of the information. In case AAL type 3/4 or AAL type 5 traffic is transported, an additional data link control (DLC) layer is added to the protocol stack, which takes care of retransmission of user data transported by the radio channel.

Figure 3-8 depicts the MSS user plane protocol stack.

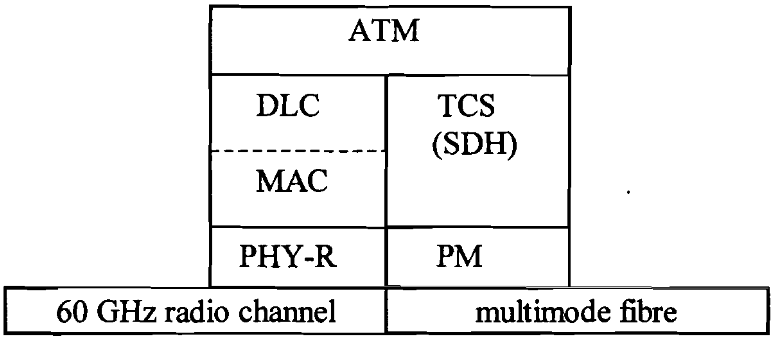


Figure 3-8: MSS user plane protocol stack

3.4.3 MSS control plane protocol stack

The MSS control plane protocol stack depicts the different layers needed for call/connection control at the MSS. First of all, a protocol stack with the ATM layer on top of the protocol stack cannot fulfil the needs for call/connection control because of the fact that the control messages are packet organized (one packet equals one total message and consists of several ATM-cells) and not ATM cell organized. Because of this packet organized structure the need for a signalling ATM adaptation layer (SAAL) is obvious. The SAAL, among other things, provides reliable transport of control messages between peer control entities (Q.2931 entities like ATM switch, MSS and the MPS) as described previously in Section 2.4 and assembles (or segmentates) the control messages. The SAAL (SSCOP part) establishes a reliable signalling channel between the ATM switch and the MSS, and between the MSS and the MPS. The signalling data is discriminated from user data by its VPI/VCI (= 0/5) combination.

On top of this SAAL a layer needs to be implemented that takes care of the handling/generation of the control messages. It is possible to implement a Q.2931 layer on top of the SAAL, but only a partition of the available functionality's are needed. Therefore a specially developed interworking layer will be implemented on top of the SAAL layer. This interworking layer consists of a subset of the Q.2931 functions with additional MEDIAN system related functions. The protocol which needs to run in this interworking-layer, concerning the control functionality's, is described in Section 4 and Section 5.

Another point of consideration is that the connection from public network (ATM switch) to MSS is an optical SONET channel (155 Mbit/s) and the connection from MSS to MPS is a radio channel able to support at least the 150 Mbit/s. There is an adaption needed of the optical characteristics on the radio channel characteristics, because the ATM technology is originally developed to be used on a (very) reliable transmission medium , like an optical fibre. The MEDIAN system is based on a point-to-multipoint network architecture like depicted in Figure 3-9.

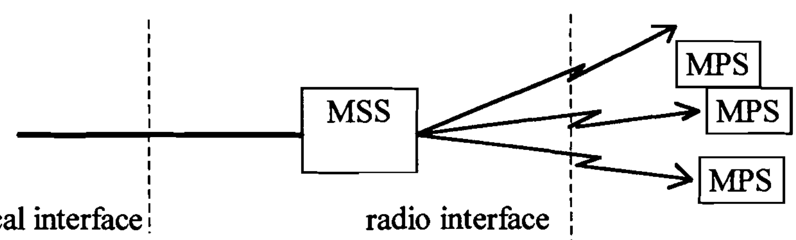


Figure 3-9: point-to-multipoint network architecture

The access to the radio channel needs to be controlled by the MSS. The MAC layer takes care of this access control.

Figure 3-10 illustrates the control plane protocol stack of the MSS.

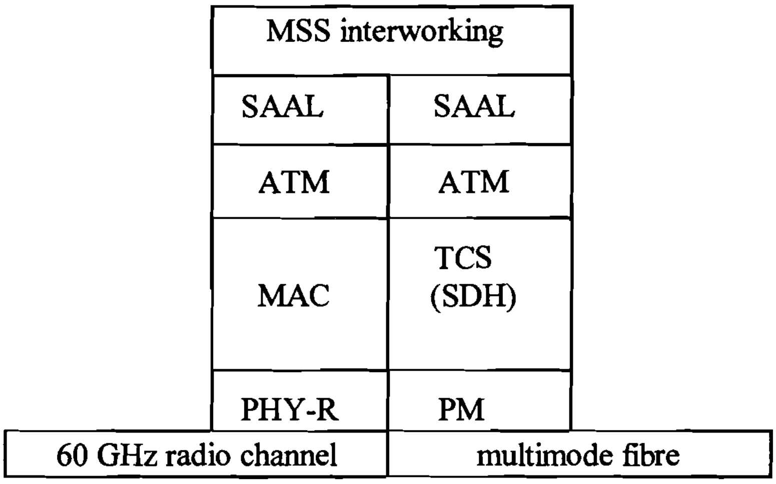


Figure 3-10: MSS control plane protocol stack

The MAC layer does not totally map to the used B-ISDN PRM. Therefore a kind of transition exists between the way of presenting the protocol stack. This is clarified in Figure 3-11.

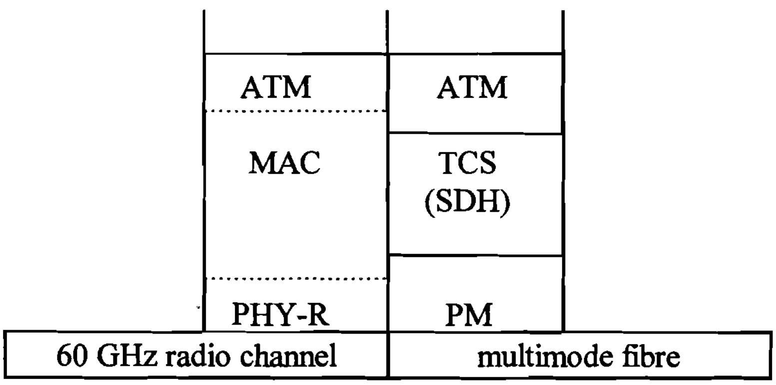


Figure 3-11: mapping of MAC on B-ISDN PRM

From now on no attention will be paid to this mapping irregularity.

3.4.4 MPS user plane protocol stack

The MPS is subdivided into two parts. The first part is the user terminal equipment (UTE). The protocol stack of the UTE already exists, it is the protocol stack of (for example) the ATM compatible workstation. The second part is the portable radio part (PRP) which, among other things, takes care of the conversion of an optical ATM cell stream into a radio channel cell stream (and also in the reverse direction) and takes care of the access control of the radio channel. The splitting is done because it is recommended not (and not practical feasible) to change the UTE internally.
The user plane protocol stack of the PRP does not differ much from the user plane protocol stack of the MSS.

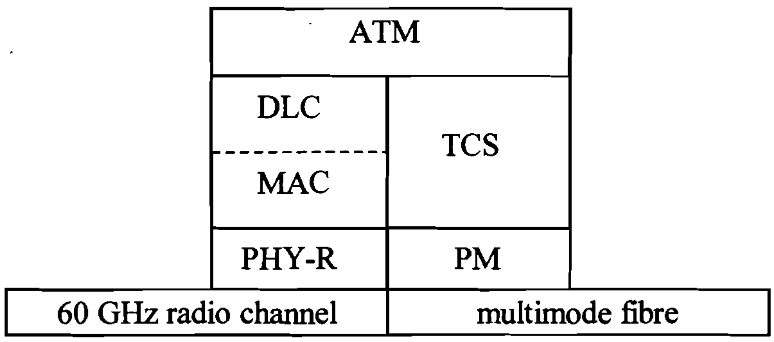


Figure 3-12: PRP user plane protocol stack

3.4.5MPS control plane protocol stack

Because of a non complete view of the functionality of the PRP concerning the call/connection control, the first suggestion was to implement a protocol stack similar to the protocol stack of the MSS (user and control plane). But after the writing of the protocol specification (see Chapter 4), it was obviously that this approach could be strongly simplified.
Figure 3-13 depicts the final protocol stack of the PRP.

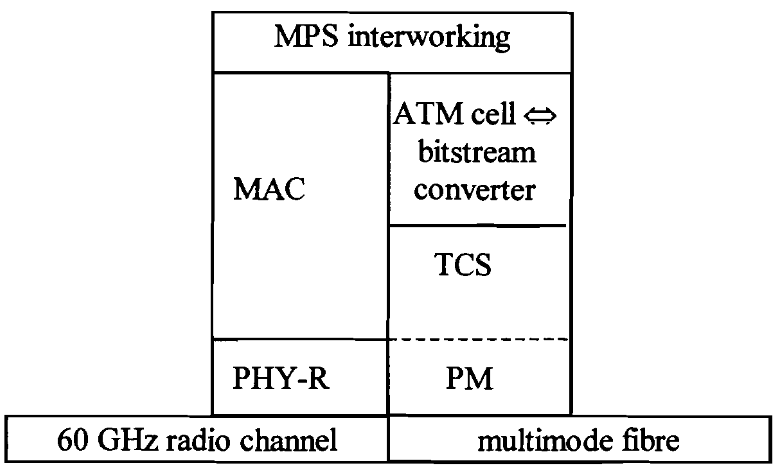


Figure 3-13: PRP control protocol stack

An additional sublayer, ATM cell ⇔ bitstream converter, is included to convert in the downlink direction the ATM cell stream into a bitstream. In the opposite direction an inverse action takes place.

3.4.6 MEDIAN user- and control plane protocol stack

The total network architector of the MEDIAN system connected to a public network (ATM switch) is illustrated in Figure 3-14.

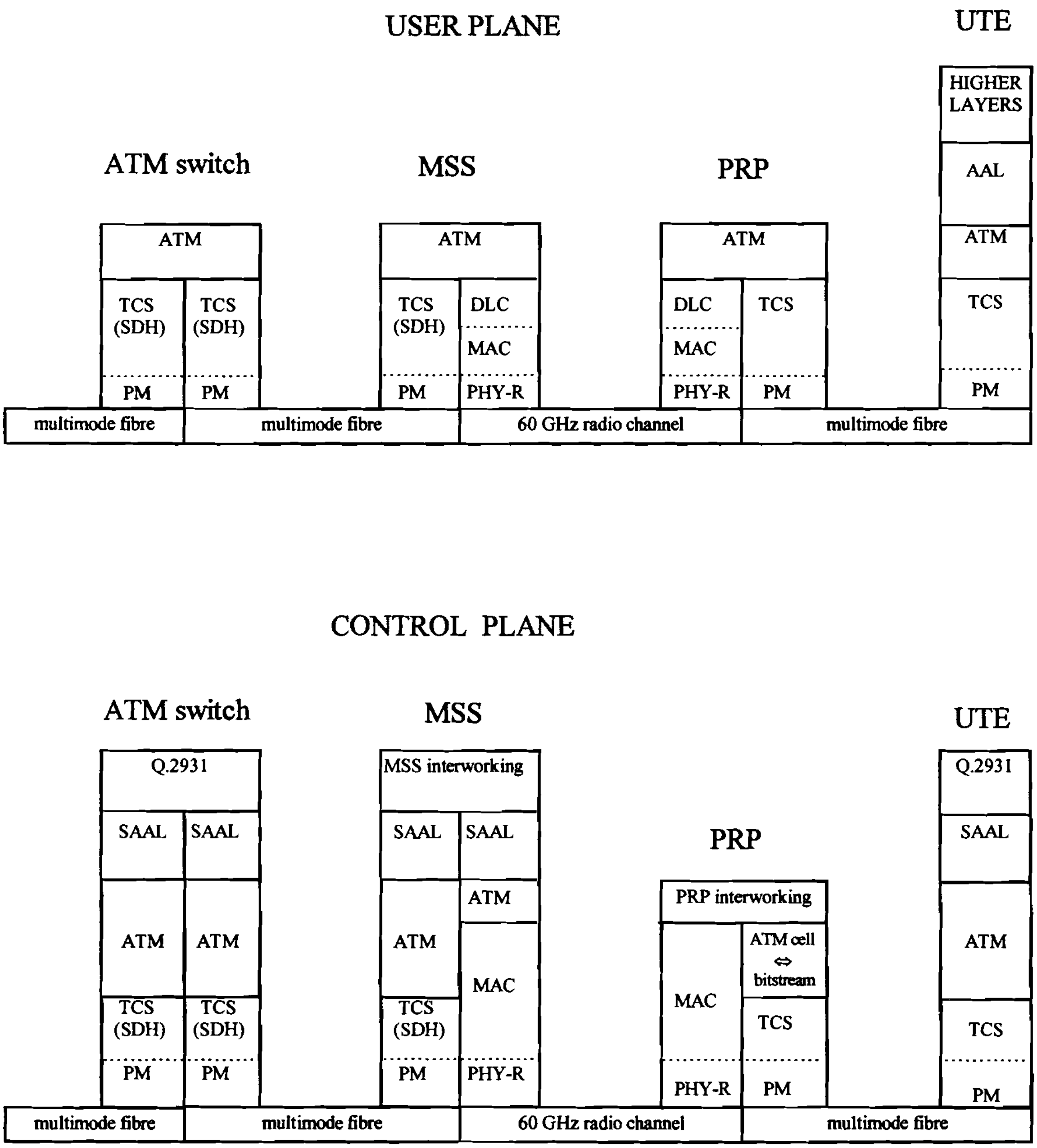


Figure 3-14: MEDIAN user- and control plane protocol stack

4. MEDIAN interworking layer in the MSS

4.1 Introduction

The interworking layer is only present in the control plane of the MEDIAN protocol stack. The SAAL in the control plane executes services that are used by the interworking layer of the MSS. The SAAL is the supplier of services and the interworking layer is the user of the service. In the following section the followed approach, the assumptions made and the final proposed protocol description of the interworking layer in the MSS are defined. Only the control plane of the MEDIAN system has been investigated because the user plane does not differ much from the user plane in a normal wired (optical) system. The only thing that is different is the presence of a data link control (DLC) layer on top of the MAC layer. This layer takes care of presenting a 'reliable' transmission (together with the MAC and the PHY-R layer) to the ATM-layer of user data.

4.2 Assumptions and Targets

Assumptions:

- because the MAC layer was developed to provide a maximum of one call per MPS, this is used as an assumption (non realistic assumption; the demonstrator needs to be able to support multimedia applications. These multimedia applications often use more then one connection. The proposed solution is easily adapted to more then one call per MPS)
- the users are only portable but not really mobile; meaning that the MPS does not move from the moment it's in call (no handover, no roaming etc.);
- the MSS is able to construct an MEDIAN internal address;
- the MPS is able to identify itself before requesting a call/connection. This can be done on a dynamic bases or static bases. A dynamic registration is assumed, because it has the benefit of a shorter internal MEDIAN address (only locally unique);
- the SAAL (SSCOP) is able to guarantee a reliable transportation of the control messages delivered/received (over the radio channel) within certain time limits;
- the DLC (retransmission) and the physical layer (line coding) are able to guarantee a reliable transportation of user data;
- mechanism available in MSS able to decide whether to accept or reject a call/connection establishment (able to use same mechanism as implemented in UTE);
- no modification of ATM-switch allowed (seen as a public switch);

Targets:

- keeping the complexity of MSS and MPS as low as possible (make use of already implemented functionalities in the UTE part of the MPS and the ATM-switch in the public network);
- detailed description of the needed functionalities in the MSS related to call/connection control in the MSS;
- maximum througput (sum of incoming and outgoing traffic) connection between MSS and the public network (ATM-switch) equals 155 Mbit/s (150 Mbit/s information)
- compatible with broadband-ISDN (B-ISDN);
- demonstrator system able to establish/maintain/release a interurban call/connection;
- demonstrator system able to establish/maintain/release a local call/connection;
- demonstrator able to support a radio connection of 155 Mbit/s and 30 Mbit/s;
- keeping overall complexity as low as possible;

4.3 Approach specification of interworking layer

Like mentioned previously, the functions the interworking layer needs to perform where not clear at the beginning. Therefore an approach was needed that yields in a specification of the interworking layer. The followed approach consists of the following steps:

- *step 1:* scenario definition
  define the different scenarios in which the MEDIAN system can be operating. In the call/connection establishment phase, call/connection maintenance phase, and in the call/ connection release phase;
- *step 2:* message flow derivation
  derive, from the different scenarios of the MEDIAN system, the message flow concentrating on the message flow in the MSS within the MEDIAN system;
- *step 3:* detailed description derivation of actions to be executed in the MSS
  derive, from the message flow of every scenario, a detailed description of the actions which have to be executed in the MSS. Concentrating on the actions of the interworking layer (including the SAAL and the MAC);
- *step 4:* protocol description derivation of actions to be executed in the MSS
  derive from the detailed description of the actions in the MSS a description of the protocol which needs to run in the MSS;

The numbers are equal to the order of execution of the step (first number one, ending with number 4). At the end of each step (except step number 1) a verification is executed to be certain that the derived step complies with the previous (source) step. This verification is also used to change small details in the previous step because, for example , the complexity can be decreased by aggregate certain actions.

Figure 4-1 depicts the work schedule.

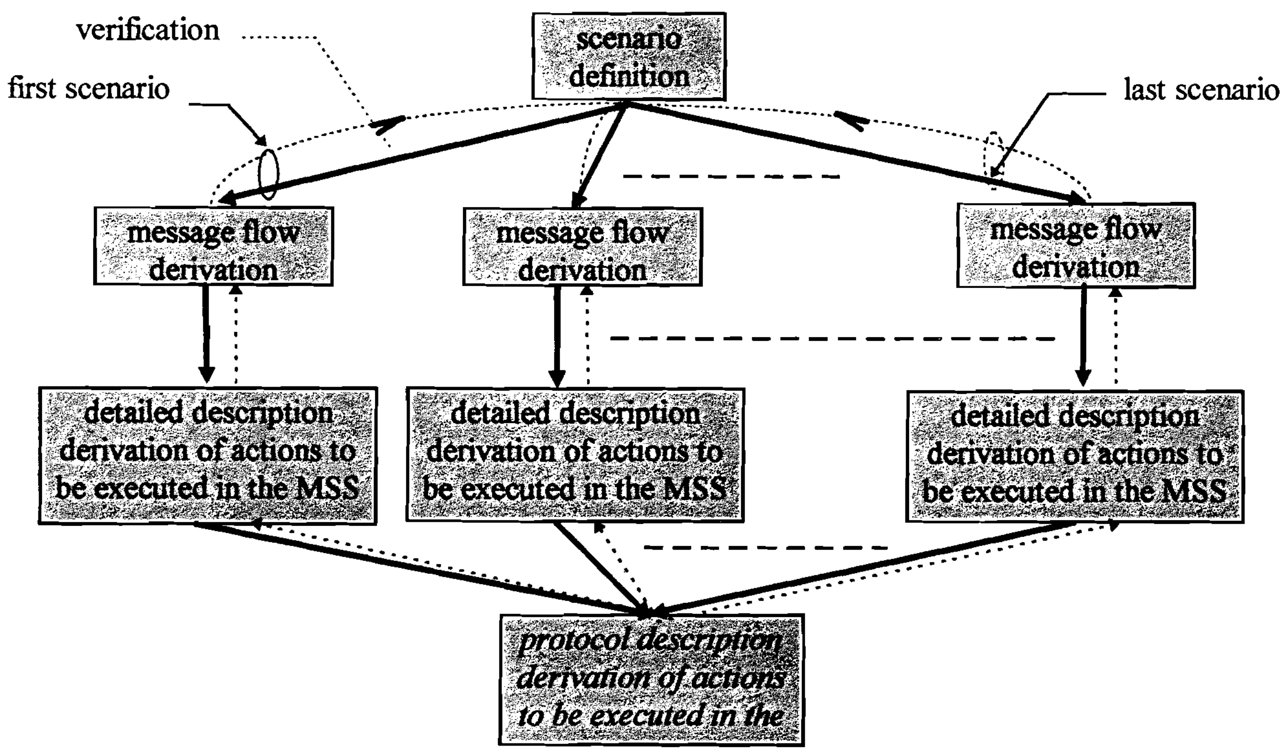


Figure 4-1: work schedule

4.4 Resulting outcome

4.4.1 Introduction

After a description of the followed approach, the protocol description of the MSS is derived using the defined approach. To clarify the results of the steps of the applied approach, an example is included which passes all the four steps. The example which is used is the scenario in which an MPS wants to establish an interurban call/connection but the call/connection establishment request is rejected by the public network (ATM-switch).

4.4.2 Results of Step 1

As noted in Section 2.5, three different call/connection types can be distinghuised in the MEDIAN system. These three call/connection types also apply for the MEDIAN demonstrator which is a subset of the MEDIAN system. Another distinction can be made when looking at the different phases of a call/connection, as discussed in Section 2.6.
Taking the above described distinctions into account, Step 1 (define scenarios) results in the following different scenarios related to a call/connection establishment:

- scenario I_Setup: a non MEDIAN user wants to establish an interurban call/connection with a MPS;
- scenario II_Setup: an MPS wants to establish an interurban call/connection;
- scenario III_Setup: an MPS wants to establish a local call/connection with another MPS;

After deriving the different scenarios related to a call/connection establishment, the different call/connection release scenarios are derived. The rejection of a call/connection establishment request is not described in this type of scenario but in the scenarios related to call/connection establishment (I_Setup, II_Setup, and III_Setup).
The possible places where a release of a call/connection can be initiated are:

- calling MPS;
- called MPS;
- the MSS;
- the public network (only in an interurban call/connection);
- non-MEDIAN user (only in an interurban call/connection).

The different call/connection release scenarios which are identified are:

- scenario IV_Release_a: an interurban call/connection release is requested by an non-MEDIAN end-user;
- scenario IV_Release_b: an interurban call/connection release is requested by an MPS
- scenario IV_Release_c: an interurban call/connection release is requested by the public network (ATM-switch);
- scenario IV_Release_d: an interurban call/connection release is requested by the MSS;
- scenario IV_Release_e: a local call/connection release is requested by an MPS (calling or called);
- scenario IV_Release_f: a local call/connection release is requested by the MSS.

Example

Scenario specification:

- a MPS wants to establish an interurban call/connection;
- the MSS can support the traffic specified and the requested quality of service and the broadband bearer service in the call/connection establishment request message (SETUP);
- the public network (ATM-switch) can not support the traffic specified and/or requested quality of service in the call/connection establishment request message (SETUP);
- public network initiates rejectance of the call/connection establishment request;
- co-ordinated releasing of made reservation during call/connection establishment request.

4.4.3 Results of Step 2

Before discussing the results of Step 2, first the different phases of a call/connection with the accompanying control messages is listed in Table 4-1.

Table 4-1: call/connection phases with the accompanying control messages

call/connection phase	accompanying messages
establishment	SETUP CALL PROCEEDING CONNECT CONNECT ACKNOWLEDGE RELEASE COMPLETE STATUS
maintenance	STATUS ENQUIRY STATUS
release	RELEASE COMPLETE RELEASE

When determining the message flow, keeping in mind that the complexity of the MSS should be on an acceptable complexity level and must be particle realizable, the MSS is developed in a way that it only generates control messages if really necessary.

Example

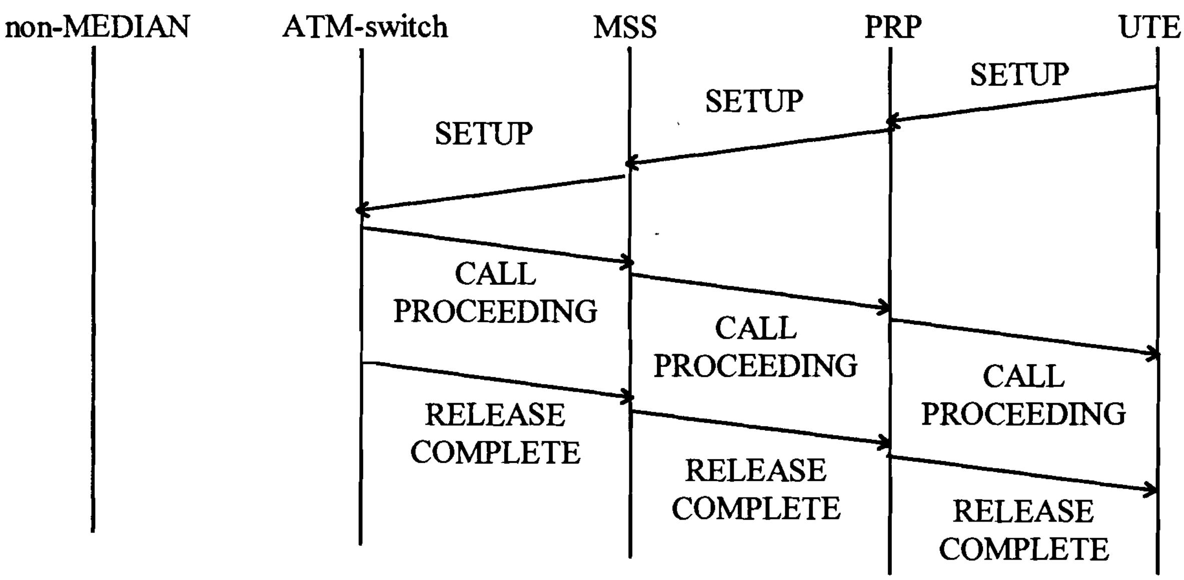


Figure 4-2: example results of Step 2

4.4.4 Results of Step 3

The result of Step 3 must be a detailed control protocol description of the interworking layer. Before being able to execute Step 3, first familiarity with the exact contents of all the messages is needed. A detailed description of the contents of the call/connection control messages is given in [ATM forum3.1], an abstract of the contents of the control messages can be found in Appendix H .
Not all of the functionality's (actions) which the MSS needs to possess can be derived from the in Step 2 obtained information flows. There are additional functionality's which have to be added to the functionality's derived from Step 2. One of the additional functionality's which needs to be included is: error control of messages, including functionality's to solve the control of errors like:

- protocol discriminator error;
- message to short (or to long);
- mandatory information element(s) missing;
- unexpected recognized information element;
- not expected call reference value.

Example
The total description of the message flow of scenario II-Setup-b is reduced to a shorter subset (27 of 108). This is done because it is the intention of the example to illustrate the meaning of the different steps and not to present the total outcome of the Step. See Appendix B for the total outcome of Step 3.
The list depicts a subset of the total outcome of Step 3.

1. the 'SETUP message' is transported to the MSS over the radio channel;.

2. at the MSS, the MSS-PHY layer delivers the constructed bitstream to the MSS-MAC layer;
3. the MSS-MAC layer delivers the 'signalling (SETUP) message to the MSS-ATM layer;
4. the MSS-ATM layer removes the ATM cell header (among other things) and delivers the ATM information cell to the MSS-SAAL;
5. the MSS-SAAL reassembles (among other things) the signalling message. The MSS-SAAL delivers the signalling message to the MSS-interworking layer;
6. the MSS-interworking layer receives the incoming signalling (SETUP) message and buffers the message;
7. the MSS-interworking layer extracts the Protocol discriminator Information Element:
    - must be 0000 1001 (Q.2931)
8. the MSS-interworking layer extracts the Call reference Information Element:
    - flag must be 0(message sent from the side that originates the call reference);
    - call reference value (23 bit) saved in static table;
9. the MSS-interworking layer extracts the Message type Information Element:
    - message type must be 0000 0101 (SETUP message);
    - save internal in 'originated' field in the static table;
    - change state of connection from Null(U0) into Call Initiated(U1) state in static table;
10. the MSS-interworking layer extracts the Message length Information Element:

- extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself);

11. the MSS-interworking layer extracts the ATM traffic descriptor Information Element:
    - deduce information about the forward/backward peak cell rate (CLP=0 and CLP=0+1) (if present);
    - deduce information about the forward/backward sustainable cell rate (CLP=0 and CLP=0+1) (if present);
    - deduce information about the forward/backward maximum burst size (CLP=0 and CLP=0+1) (if present);
    - save information in static data base;
12. the MSS-interworking layer extracts the broadband bearer capability Information Element:
    - bearer class (BCOB-A, BCOB-C, BCOB-X(most likely to be used));
    - traffic type (CBR, VBR);
    - timing requirements (no indication, end-to-end timing required, end-to-end timing not required;
    - susceptibility to clipping (yes, no);
    - user plane connection (point-to-point, point-to-multipoint);
    - save information in static data base;
13. the MSS-interworking layer extracts the Called party Number Information Element and deduces the ATM end system address;
14. the MSS-Interworking layer extracts the QoS Information element. The QoS information element can contain information concerning the:
    - cell error ratio;
    - severely-errored cell block ratio;
    - cell misinsertion ratio;
    - cell transfer delay;
    - mean cell transfer delay;
    - cell delay variation;
    - save information in static data base;
15. the MSS-interworking layer checks if message is errorless (Yes assumed);
16. the MSS-interworking layer decides if the end-user is:
    - an MEDIAN system user;
    - not an MEDIAN system user;
17. the MSS-interworking layer checks if requested QoS can be provided, using the extracted QoS information. There is additional information needed about the behaviour of the overall system for example BER, bursty characteristic of errors, propagation delay, transport delay of ATM cells in ATM layer, buffer capacity in Median Server Station etc. It should be possible to deduce the necessary information from the information maintained in the static data base;
18. the MSS-interworking layer checks if able to support combination of traffic parameters in the directions (MSS⇒ MPS and MPS⇒ MSS):
    - peak cell rate when CLP=0;
    - peak cell rate when CLP=0+1;
    - tagging (if requested);
    - sustainable cell rate CLP=0;
    - sustainable cell rate CLP=0+1;
    - maximum burst size CLP=0;
    - maximum burst size CLP=0+1.

The MSS-interworking layer maintains the information of the static list, therefore having the data to derive the free slots in the uplink and the downlink;

19. QoS check=okay AND traffic parameter check=okay AND non MEDIAN user (because this is scenario II_Setup_b);
20. MSS-interworking layer maps ATM end user address on an internal address;
21. MSS-interworking layer: deduce time on which cells, up- and downlink, expire and retain these values in the static list (or just do selective queuing);
22. MSS-interworking layer chooses and saves an unused RVCI for this call/connection;
23. MSS-interworking layer starts timer T303;
24. IF no reaction to the SETUP message before expiry of timer T303 THEN initiate call/connection release;
25. the signalling (SETUP) message is delivered it to the MSS-SAAL;
26. the MSS-SAAL delivers the segmented signalling (SETUP) message to the MSS-ATM layer;
27. the MSS-ATM layer delivers the bitstream to the MSS-PHY layer;

4.4.5 Results of step 4

4.4.5.1 Introduction

The results of Step 4 (derive protocol description) are distributed over three sections, each addressing a specific direction of the message flow. The three possibilities are:

1. arriving from the side of the public network (ATM switch or non-MEDIAN user);
2. arriving from the side of the MPS;
3. MSS initiates a call/connection changing procedure or requests a status of a call/connection;

The for Step 4 procedure of explaining the results on the basis of an example is not maintained. This is done because in Step 4 the results of Step 3 are aggregated. In contrast to the previous steps 2 and 3, where a discrimination took place between different scenarios, step 4 discriminates between the different call/connection control messages and the direction of the message flow. All of this makes it impossible to continue, on a logical bases, with using an example as an explanation.

4.4.5.2 ATM-switch ⇒ MSS protocol description

The protocol description developed is valid for the case that message is received by the MSS from the public network (ATM-switch) side of the MSS. The distinction between a message coming from the public network or from the MEDIAN system is made by the interworking layer. The interworking layer receives a data-receive indication (a primitive) from the SAAL layer of the public network side or a data-receive indication from the SAAL layer of the MEDIAN system side of the MSS (see control plane protocol stack of the MSS, Figure 3-14). Caused by the splitting, depending upon the direction of the message, only a subset of the total number of scenarios is applicable for the ATM-switch ⇒ MSS protocol description (set of interurban call/connection establishments or releasements).

Figure 4-2 depicts the flow diagram of the protocol description of the messages that arrive from the network (ATM-switch ⇒ MSS protocol description). The text in the boxes shortly describes the associated activities. The arrows show the direction of execution. If the last box does not contain a continue message then this implies that it is the end of the message flow within this specific protocol description.

It is conspicuous that all the messages in Figure 4-2 end in the box 'continue with message handling'. The message handler takes care of all the specific actions which have to take place, depending on the message type. The flow diagram of the message handler of the ATM-switch ⇒ MSS protocol description is given in Figure 4-2 and Figure 4-3. It is seen that the continuation is based, not only on the message type but also on a field called *initiator*. The field initiator differentiates between an internal call/connection or an external (interurban) call/connection. All the messages in this specific protocol arrive from the public network (message send by ATM-switch or non-MEDIAN user) implying that it is always an external (interurban) call/connection.

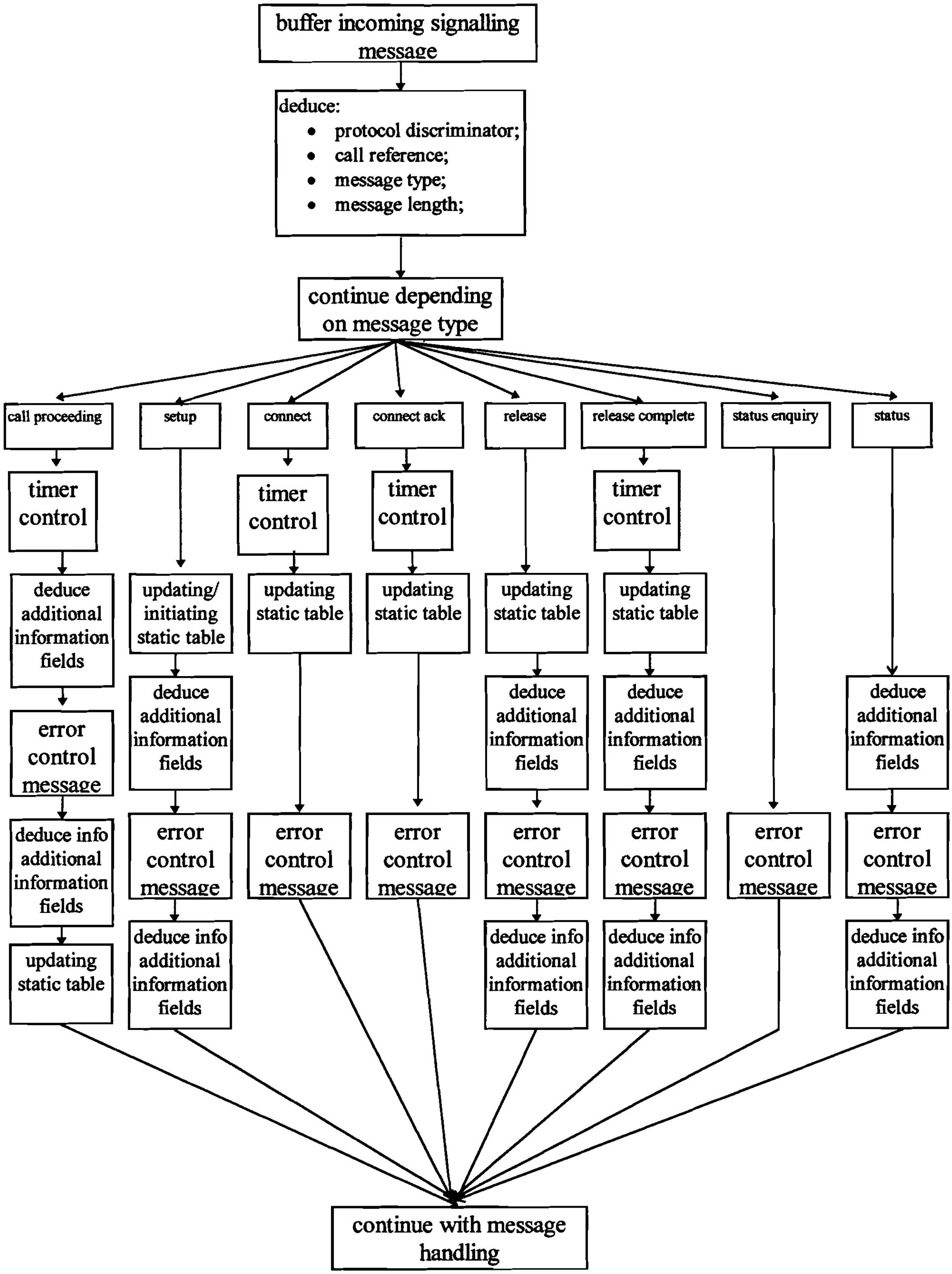


Figure 4-2: flowdiagram of the ATM-switch ⇒ MSS protocol description

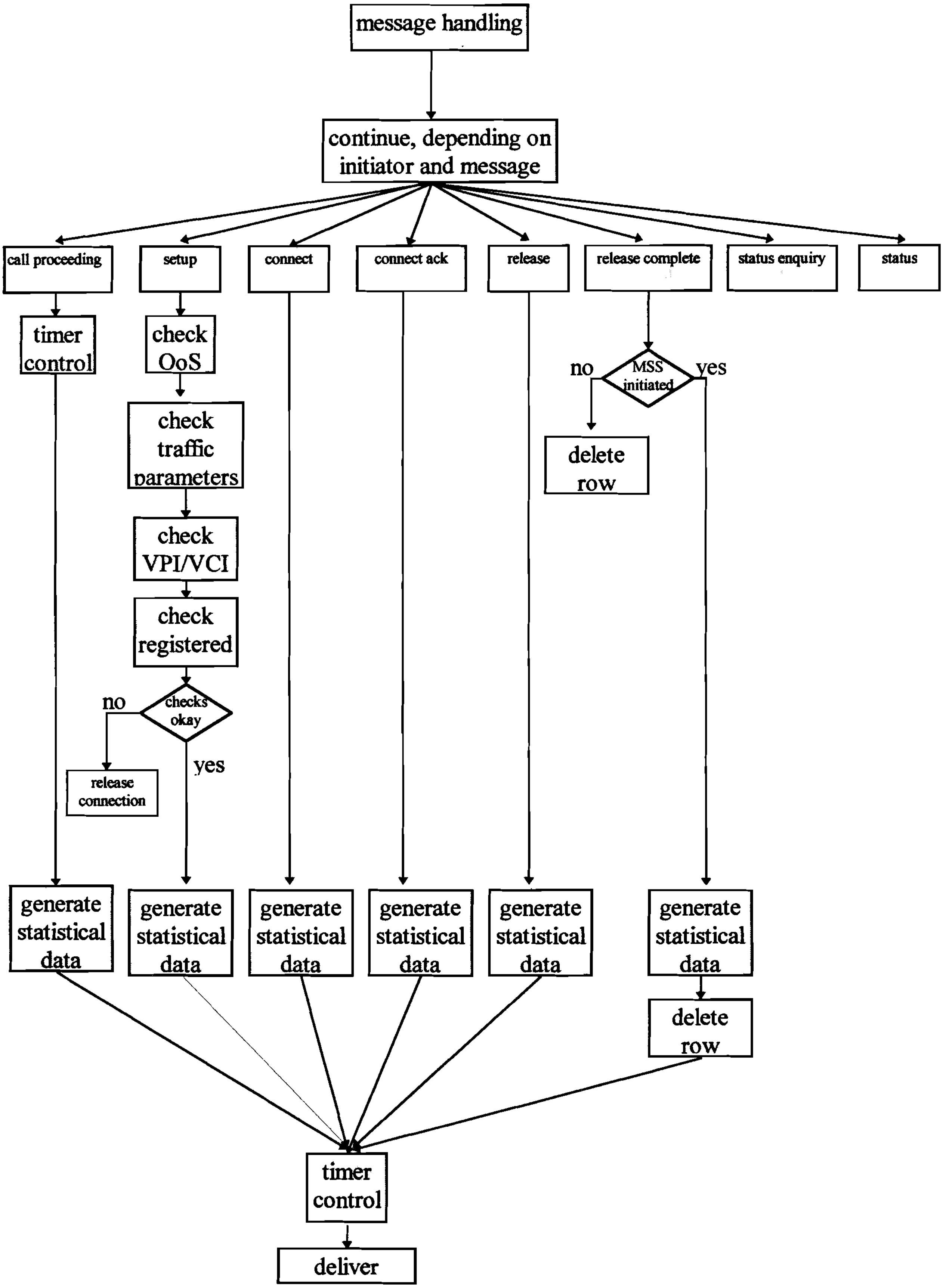


Figure 4-3: flow diagram of the message handler of the ATM-switch ⇒ MSS protocol description

4.4.5.3 PRP ⇒ MSS protocol description

When focusing on the message flow coming from the opposite direction, PRP (MPS) to MSS not only interurban calls/connections exist but also the local calls/connections.
The interworking layer receives a data-receive indication (a primitive) from the SAAL layer of the MEDIAN system side. The complexity of the PRP ⇒ MSS protocol description is higher when compared with the complexity of the previous described protocol description. Reason for this increase of complexity is the call/connection establishment of local calls/connections. In the local call/connection establishment the MSS is the initiator. With initiator is (among other things) meant that the MSS is responsible for:

- selecting a VPI/VCI combination for a local call/connection establishment;
- assigning a call reference value to the internal downlink calls/connections;
- generation of messages which normally are generated by the public network (read ATM-switch);
- traffic management of the aggregated traffic (the traffic coming from the public network is already properly ordered);
- discrimination between a local call/connection and an interurban call/connection;

The protocol description flow diagram of the messages received from the MPS (PRP) by the MSS (PRP ⇒ MSS protocol description) is depicted in Figure 4-4 and Figure 4-5.

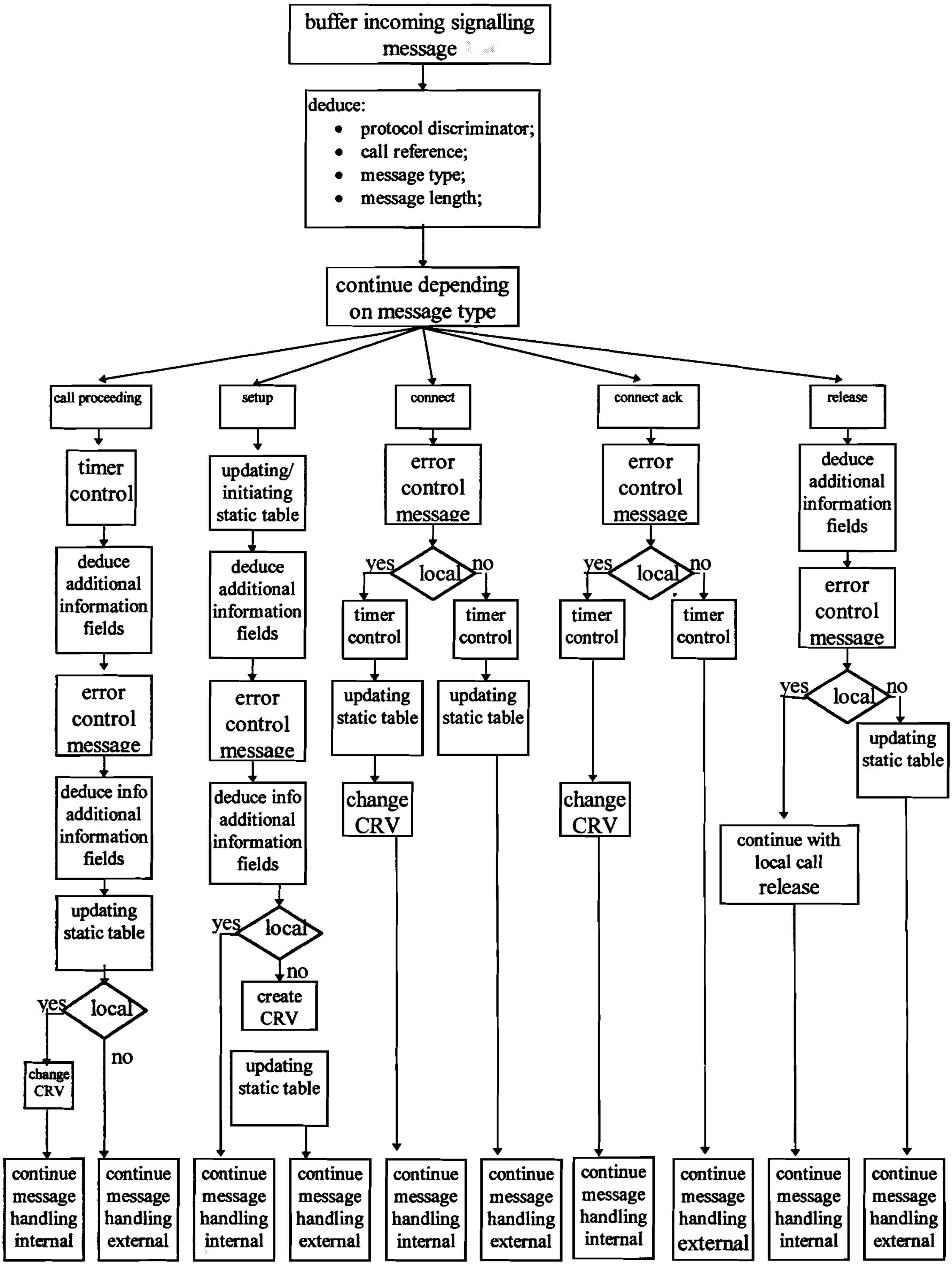


Figure 4-4:first subset of the total PRP ⇒ MSS protocol description

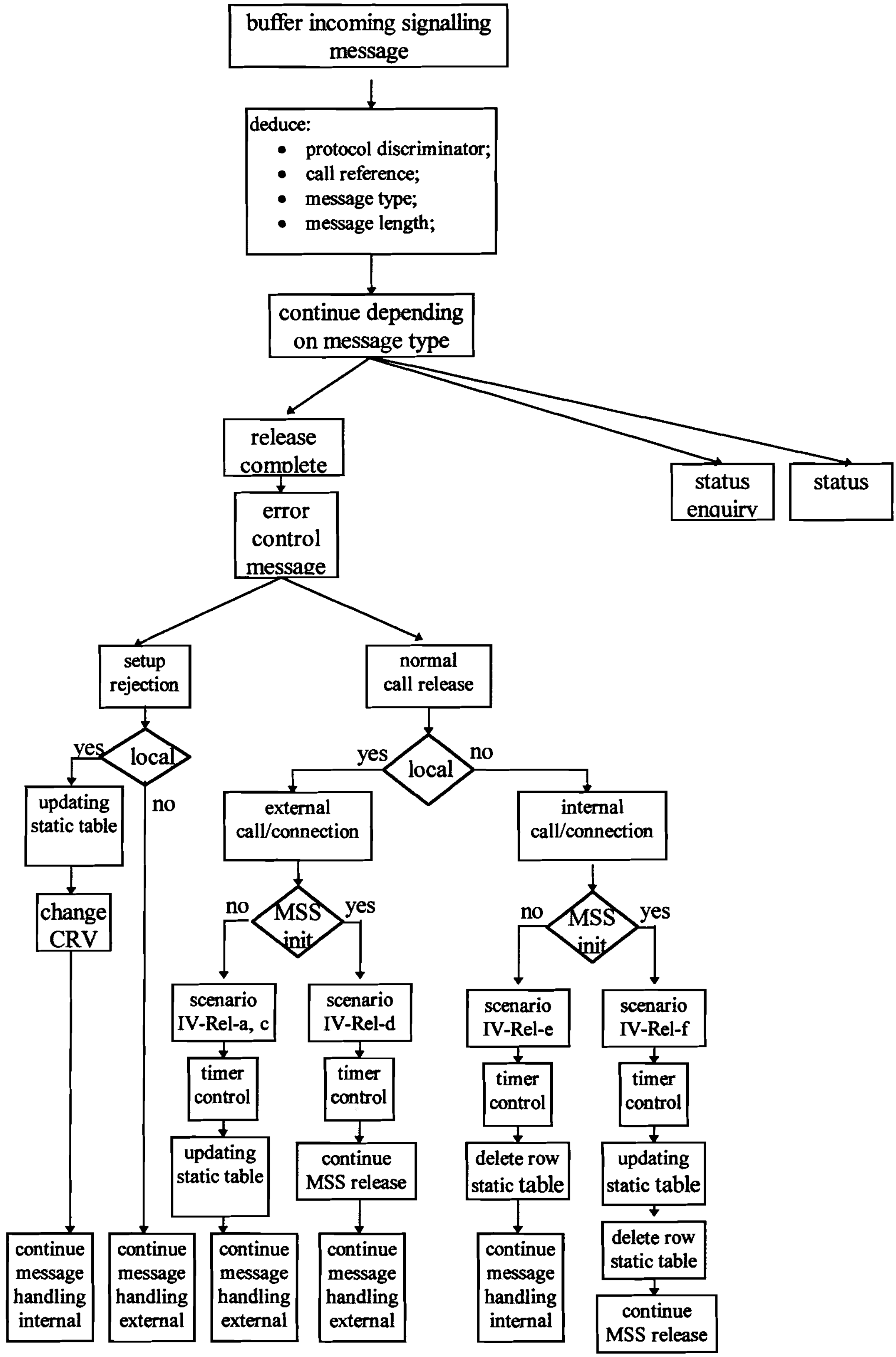


Figure 4-5: second subset of the total PRP Þ MSS protocol description

4.4.5.4 Error control of messages

In both the ATM-switch to MSS and the PRP to MSS protocol descriptions an error control mechanism is included which checks all the received messages on possible errors. This is done because when a message is received using protocol discriminator *Q.2931 user-network call control message* (valid for all the messages used in the protocol descriptions) then this message must first pass the checks which are listed.

Protocol discriminator error
When a message is received with a protocol discriminator coded other then *Q.2931 user-network call control message* then this message is ignored (deleted).

Message too short
The length of the message is indicated in the message length information element. The MSS calculates the actual received length of the message. If the measured message differs from the indicated message then the message shall be ignored.

Call reference error
If the received call reference information element is errored, for example an unrecognised call reference value, then the MSS will take the necessary actions. The action taken vary from sending a status message with the appropriate cause to totally releasing of the call/connection.

Message type or message sequence error
Whenever an unexpected message is received (for example a CONNECT message received in response to a RELEASE message), then a STATUS message shall be returned with appropriate cause number. The first four information elements: protocol discriminator, call reference, message type, and message length always have to appear in the order specified in this sentence (first protocol discriminator). Variable length information elements may appear in any order within a message.

Mandatory information element missing
Actions which will be taken if a mandatory information element is missing depends on the type of the message. In some cases (STATUS, STATUS ENQUIRY, CONNECT, CONNECT ACKNOWLEDGE) the total message is ignored and no state change should occur. In these cases a STATUS message is returned with the appropriate cause number. If the message type is one of the following message types: SETUP, RELEASE, RELEASE COMPLETE, and a mandatory information element is missing then a RELEASE COMPLETE message is returned with appropriate cause number.

Mandatory information element content error
When a message other than SETUP, RELEASE, or RELEASE COMPLETE, is received which has one or more mandatory information element(s) with invalid contents, no action will be taken on the message and no state change should occur. A status message is returned with appropriate cause number. If the message type is not one of the above specified, then a RELEASE COMPLETE message is returned with specific cause.

Unrecognized information element
When a message is received that has one or more unrecognized information elements, then the receiving entity shall proceed as follows. Actions shall be taken on the message and those information elements which are recognized and have valid content. When the receive message

is other than RELEASE or RELEASE COMPLETE, a STATUS message may be returned with accompanying cause number. If a RELEASE message is received then a RELEASE COMPLETE message is returned. If a RELEASE COMPLETE message is returned then no further actions are executed.

Unexpected recognized information element
When a message is received with a recognized information element that is not defined to be contained in that message, then the MSS interworking layer shall treat the information element as if never received.

4.4.5.5 MSS protocol description

After the previous two descriptions of the protocol description, splitted in the receiving direction of the message, the protocol description of the needed control functionality's when the MSS initiates a call/connection release is described. This possibility is taken into account because , for example, maintenance actions have to take place in a part of the MSS resulting in releasing of certain connections. Another example is when a subscriber generates traffic that does not correspond with the agreed (call/connection establishment) traffic contract, the MSS should then be able to release a connection (or connections). The last example refers to the unpredictable behaviour of the radio channel (for example a person standing in the direct line of sight). If the measured bit error rate (BER) exceeds the BER agreed during call/connection establishment (agreed QoS) then the MSS could decide to release that (or all) connection(s).

A differentiation is made between local calls/connections and interurban calls/connections in order to reduce the complexity of the MSS protocol description. Another step taken to reduce the complexity of the MSS protocol description (and in the end of course the protocol itself) is the spitting of the releasing procedure into two separated flows. One flow concerns the signalling messages belonging to the calling party (MPS or non-MEDIAN user) and the other the signalling messages belonging to the called party (MPS or non-MEDIAN user). An example of reduction of protocol description complexity concerning the previous step is the releasing of an internal connection (of course release initiated by the MSS). In this example first the connection to one of the MPS is released before taken any actions in the process of call/connection releasing towards the other MPS. This is depicted in Figure 4-6.

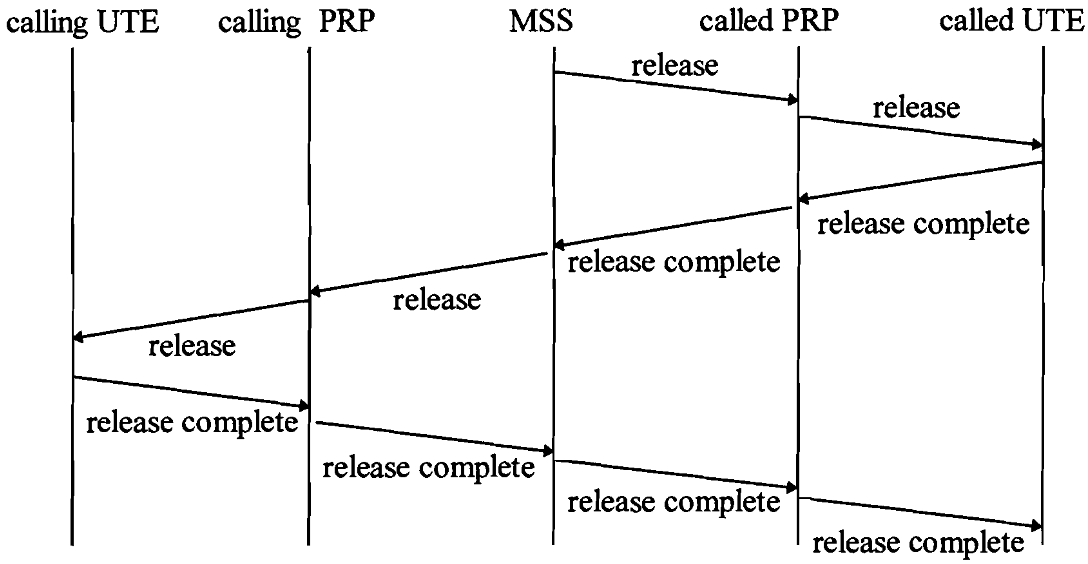


Figure 4-6: example internal call/connection release

By doing so the usage of the signalling timeslot in the uplink (and downlink) frame is allocated to only one MPS. There is no contention of the calling MPS and called MPS of the uplink signalling timeslot which causes a gain in release time. But on the other hand introduces the splitting of the two flows an increase of release time because first one connection needs to be totally released before the MSS is allowed to start with initiating of release towards the other connection. The time needed to release the call/connection is not very important because the signalling message are not that time critical (compared to user data). The main reason of this flow splitting is the decrease of protocol complexity.

Figure 4-7 shows the symmetric behaviour of the protocol description. This also contributes to a less complex implementation of the protocol.

The MSS protocol description flow diagram is depicted in Figure 4-7.

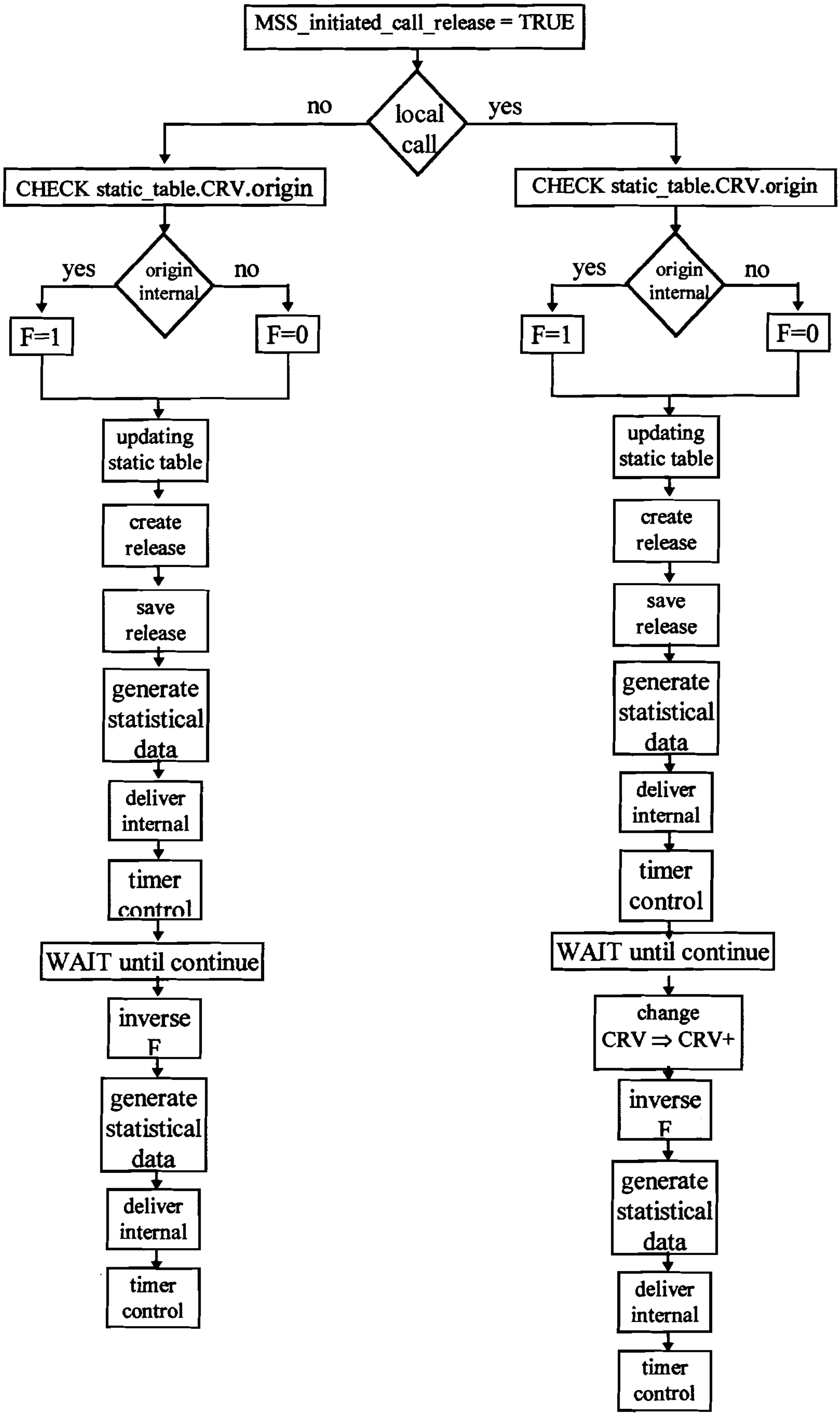


Figure 4-7: MSS protocol description flow diagram

4.4.5.6 Static table in the MSS

The static table in the MSS is used to maintain the information needed to execute the protocol on a proper way. The static table also maintains the information (maybe) needed by the MAC layer. Information which could be of interest to the MAC is:

- mapping of VPI/VCI on a radio VCI (RVCI);
- calculated delay: maximum delay which an ATM cell of a specific call/connection may undergo (in both the uplink and downlink direction).

The information maintained in the static list needed to execute the protocols is listed in Table 4-2.

Table 4-2: information fields static table used for MSS protocol description

static table field	description
CRV	Call Reference Value: unique local identification of the connection;
CRV+	Only needed in a local call/connection to identify the counterpart of the connection identified by the CRV;
VPI_VCI	indicating the VPI/VCI combination of the user-data flow belonging to the specific CRV;
RVCI	used to identify the reserved slot in the radio interface controlled by the MAC layer, constant during call/connection existence;
state	describing the state the connection momentary is in (important when implementing the protocol; finite state machine);
con_type	used to discriminate between a local and an interurban connection;
originated	needed when the MSS initiates sending of a message towards the calling respectively called party

4.4.5.7 Used timers

Table 4-3 depicts the used timers in the ATM-switch to MSS, PRP to MSS, and the MSS call release protocol. The timers are used in case anything unpredictable goes wrong and the addressed entity is not able to respond to the received message. If these timers are not included then deadlock could occur whenever a entity is not able to answer to the received message. For example if the MSS sends a SETUP message and the MPS does not respond to this message.

Table 4-3: Used timers in the MSS protocol descriptions

timer number	state of call	start	stop
T303	Call Initiated	SETUP sent	CONNECT CALL PROCEEDING RELEASE RELEASE COMPLETE received
T308	Release Request	RELEASE sent	RELEASE RELEASE COMPLETE received
T310	Outgoing Call Proceeding	CALL PROCEEDING received	CONNECT RELEASE received
T313	Connect Request	CONNECT sent	CONNECT ACKNOWLEDGE received

4.4.5.8 Problems originated from the protocol descriptions

In the following, several problems arising from the protocol descriptions are discussed. If necessary, a solution is given.

VPI/VCI selection

The selection of an unique VPI/VCI combination is needed whenever a local call/connection establishment request is initiated. In an interurban call/connection the choice of an unique VPI/VCI combination is done by the public network. In a local call/connection the MSS is responsible for the selection of an unique pair of VPI and VCI. The MSS is aware of all the VPI/VCI combinations which are active in a local/interurban call/connection with a MPS. The selected VPI/VCI combination (selected by MSS) will never pass the MSS in the direction towards the ATM-switch because it is only valid in a local call/connection. In ATM standardization the VPI concept is introduced to enhance the switching. All the VCIs with the same VPI are switched together. This means that only one (maybe two because bi-directional) VPI value is (are) used in the direction from public network (ATM-switch) towards the MEDIAN system (MSS).

Three different situations apply. Figure 4-8 depicts the different situations with their way of selecting the VPI/VCI combination.

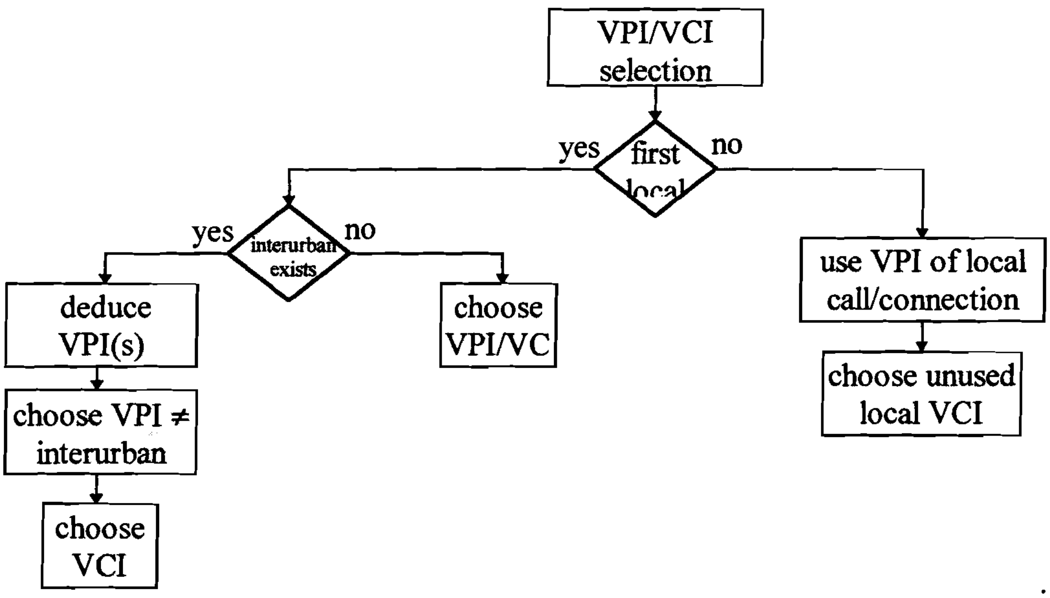


Figure 4-8: VPI/VCI selection

In the situation where the MSS randomly selects the VPI and the VCI (first connection with a MEDIAN user), restricted by range of available VPI/VCI combinations, a conflict can arise if the public network selects the same VPI/VCI combination upon another request to establish a call/connection. The public network is not acquainted with the VPI/VCI selection(s) of the MSS. The MSS checks if the requested VPI/VCI combination in the SETUP message is still available within the MEDIAN system. If the VPI/VCI combination is not available (because MEDIAN system uses this combination in a local call/connection) then the MSS will reject the call/connection establishment request enclosing a cause which indicates VPI/VCI not available. The public network can decide to try another VPI/VCI combination. Probably, in the near future, negotiation concerning the VPI/VCI will be also implemented (in addition to negotiation of QoS and traffic parameters).

Call reference value selection
The same problem occurs when the MSS needs to select a call reference value in a local call/connection. The same procedure, as applied for selecting a VPI/VCI combination, is used.

Internal address mapping
The MSS maps the large address specifying the ATM end-user on a shorter internal address (for example the MAC address of the addressed MPS). This identification of the ATM end-user is needed in the PRP to decide whether the signalling message is addressed to this particular PRP. If this internal address is not inserted, then the PRP feeds all the signalling information to the UTE. The UTE detects that the signalling information is addressed to the UTE, or not. In the latter case, the UTE will send a message to the MSS indicating that the signalling message was not addressed to it. The MSS needs to handle all these incoming message. Thus inclusion of the internal address in the broadcast cell has as disadvantage the procedure of internal address mapping and as advantage that the UTE only receives the signalling messages addressed to this particular UTE (MPS) and no additional message handling is needed.

Another solution would be the implementation of higher layers (SAAL, higher layer) for recognition of the ATM end-user address in the PRP. This would increase the complexity of the PRP dramatically. This increase of complexity of the PRP does not comply with one of our goals of the PRP (MPS), keeping the complexity as low as possible (by the way; a way of identification of the MPS is also needed when a MPS user wants toe establish a call/connection. The MPS then initiates a request towards the MSS with an identification of its identity.

Summarizing:

advantage internal mapping	**disadvantages internal mapping**
decrease complexity	unique internal address mapping needed
decreases computing time MPS	additional space needed in broadcast cell
less costs MPS	

Client registration
Generally two different types of client registration methods exists:
- static registration;
- dynamic registration;

If a static registration mechanism is used, then every MPS needs to be provided with a world-wide unique MEDIAN registration number. This registration is executed only ones per MPS (at an authorized company). In contrast to static registration, dynamic registration needs a registration procedure which needs to be executed every time an MPS enters a MEDIAN system. The MEDIAN registration number has only global significance. The length of this static MEDIAN registration number is much larger compared to the length of the dynamic MEDIAN registration number. It is not clear what the exact information is, inserted in the broadcast cell (see Section 3.3.2), but the length of the static MEDIAN registration number will most likely be to large. Therefore, it is suggested to use dynamic registration instead of static registration despite the increase of complexity (because of dynamic registration procedure).

5. MEDIAN interworking layer in the MPS

5.1 Introduction

After discussion the description of the MSS protocol, the MPS specific interworking layer is the focus of our attention. As mentioned before (in Chapter 4), the MPS consists of a unit needed to adapt the users, in an appropriate way, to the radio channel. The final connection between the MEDIAN system and the MEDIAN user is performed by the UTE. No actions are needed in the development of the protocol of the UTE (based on ATM forum 3.1) because this protocol is already implemented, for example on an ATM adapter card in a workstation.

5.2 Assumptions and targets

Assumptions:

- because the MAC layer was developed to provide a maximum of one call per MPS, this is used as an assumption (non realistic assumption, the demonstrator needs to be able to support multimedia applications. These multimedia applications often use more then one connection. The proposed solution is easily adapted to more calls per MPS)
- an MPS does not move during a call (no handover, no roaming etc.);
- the PRP knows the internal address (obtained in a dynamic registration procedure);
- PRP-interworking layer is able to retrieve an uplink timeslot from the MAC layer;
- PRP-interworking layer is able to deduce downlink ;
- PRP is able to deduce timeslot identified by a certain RVCI;
- MAC-layer and PHY-layer deliver a 'reliable' service concerning the transmission of the data and control information (BER $\approx 10^{-7}$);

Targets:

- keeping the complexity of the protocol description to a limit;
- no modification of the UTE is allowed;
- PRP functions as a transparent ATM-path between the MSS and the UTE;

5.3 Approach specification of interworking layer

The followed approach is the same as the one followed in the description of the interworking layer of the MSS. The approach was subdivided into 4 steps, which finally led to the solution of the functionality's (protocol description) of the interworking layer in the PRP of the MPS.

5.4 Resulting outcome

5.4.1 Introduction

One of the goals is to keep the complexity of the PRP as low as possible. By doing so, the price of the PRP can be minimized (important because number of PRPs in the MEDIAN system, not demonstrator, is always much larger compared to the number of MSSs). Most of the additional functionality's are centralized in (and controlled by) the MSS.
The PRP is ATM-cell oriented, meaning that the operations concern the receiving and transmitting of the ATM-cells do not exceed the ATM-layer. Therefore no ATM adaptation functionality's are needed in the PRP, which decreases the complexity of the PRP (goal of the PRP). Because of the ATM-cell oriented nature of the PRP, dependency on the behaviour of

the medium access control (MAC) layer is increased. The design/operation of the MAC is still under investigation of the responsible workpackage. Therefore, the interworking layer of the PRP is developed in a way that it is easily adapted to a change of the applied MAC design/operation. Another advantage of the ATM-cell oriented PRP is the decrease of throughput time when compared with a message oriented structure.

5.4.2 Results of Step 1

The different scenarios as defined in Step 1 of Section 4.4.2 also apply in case when we want to describe the MEDIAN interworking layer in the MPS (specifically in the PRP). There was no need to define additional scenarios. This is obvious, because the PRP is mainly responsible for the adaptation of the reliable optical channel to the unreliable radio channel and does not take actions concerning issues as call/connection acceptance or rejectance.

5.4.3 Results of Step 2

The message flow derived in Section 4.4.3 reflects the message flow of the total end-to-end connection. This means that the message flow of the PRP is also taken into account. Like described earlier, the main task of the PRP is to adapt the radio channel to the optical channel, and vice versa. No messages are derived from the incoming/outgoing flow (obvious because the PRP is ATM-cell oriented). The PRP is not involved in actions concerning the signalling within the MEDIAN demonstrator (or MEDIAN system).

5.4.4 Results of step 3

Step 3 of this section clearly differs from Step 3 of Section 4.4.4, because it focuses on the PRP only when compared with the previous steps (1 and 2) which focus on the overall system (demonstrator). To keep a clear view of the steps taken, the same example is used as compared to Section 4 (II-Setup-b). Only two different flows can be distinghuised:

- from the UTE to the PRP (outgoing flow);
- from the MSS to the PRP (incoming flow),

like depicted in Figure 5-1.

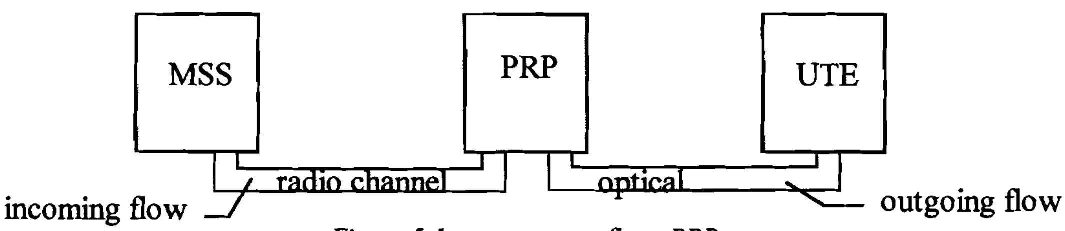


Figure 5-1: two message flows PRP

An example of the incoming and the outgoing flow is listed.

1. the PRP-PHY layer delivers the received bitstream to the PRP-ATM layer;
2. the PRP-ATM layer only deserialises the bitstream into ATM cells, and the ATM cells with VPI/VCI=0/5 are delivered to the PRP-Interworking layer;
3. the PRP-Interworking layer buffers (in a receive buffer) the incoming ATM cells. Buffering is needed because the PRP-MAC ,maybe , first has to content for a free (signalling) slot; As soon as the PRP-MAC delivers the broadcast cell with the internal address in the signalling slot, then the PRP-Interworking layer can deduce the allocated timeslots from the broadcast slot;
4. the PRP-Interworking layer delivers the ATM cells to the PRP-MAC;
   the PRP-MAC layer delivers the bitstream to the PRP-PHY layer;

5. the 'SETUP message' is transported to the MSS over the radio channel;.

   .
   .
   .
   .
   .

29.'CALL PROCEEDING message' is transported to the MPS using the radiochannel;

30.the PRP-PHY layer delivers the constructed bitstream to the PRP-MAC layer;
31.the PRP-MAC delivers the broadcast cell to the PRP-Interworking layer;
32.the PRP-MAC delivers the signalling information cell(s) to the PRP-Interworking layer ;
33.the PRP-Interworking layer deduces the RVCI and the internal address (or ATM end system address or E.164 address) from the signalling cell in the broadcast cell;
34.the PRP-Interworking layer checks if the internal address is the address of the PRP (UTE)
35.if not then do not proceed;
36.if yes then proceed (yes is assumed);
37.the PRP-Interworking layer saves the RVCI in the static table;
38.the PRP-Interworking layer buffers the signalling information cell(s) (of the different timeslot(s)) (s) belonging to the SRVCI;
39.the PRP-Interworking layer starts a timer for releasing the reservation of the SRVCI;
40.the PRP-Interworking layer feeds the buffered ATM (signalling) cells to the ATM-layer (where, among other things, the ATM cells are serialised into a bitstream);
41.PRP-ATM layer delivers bitstream to the PRP-PHY layer;

42.CALL PROCEEDING message is transported to the calling UTE, using the optical channel;

5.4.5 Results of Step 4

5.4.5.1 Introduction

The results of Step 4 (derive protocol description) are distributed over two sections. The results are split depending upon the receiving direction of the message. The possibilities are:

1. arriving from the side of the MSS;
2. arriving from the side of the UTE.

As the results of Step 4 of the previous chapter, the applied procedure of explaining the results of Step 4 on the basis of an example is not maintained. In contrast to the previous Steps 2 and 3, where a discrimination took place between different scenarios, now a discrimination takes place between the two different directions (described above).

5.4.5.2 MSS ⇒ PRP protocol description

This protocol runs valid whenever a message is received from the MSS (coming from another MPS or from the ATM-switch.
Caused by the splitting, depending upon the receiving direction of the message, only a subset of the total number of scenarios is applicable for the MSS ⇒ PRP protocol description.

Before continuing, first a brief description of the broadcast cell is given. The broadcast cell is one of the 128 timeslots of the time division duplex (TDD) frame. It is used in the downlink frame of the TDD frame. Information in the broadcast cell is addressed to all the MPS (active and idle) and it contains information concerning the timeslot allocation of user and control (signalling) data.
Figure 5-2 illustrates an example of how the broadcast cell could be structured (this structure is used in the protocol description).

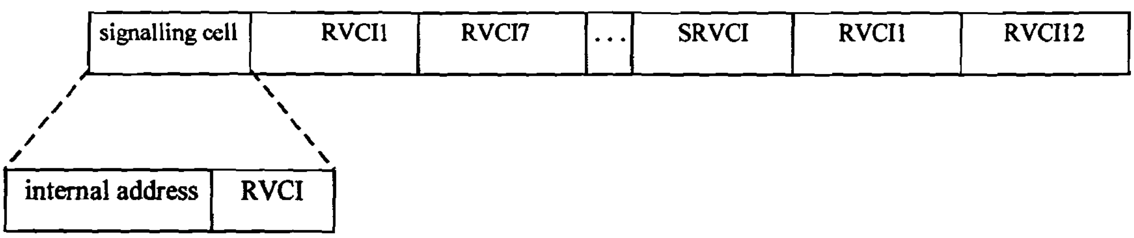


Figure 5-2: broadcast cell

For the execution of the call/connection control procedures the signalling RVCI (SRVCI) is important. The SRVCI has an unique value, for example 00000.
The MAC layer delivers the broadcast cell and the signalling cell(s) (if any) to the interworking layer. Like depicted in Figure 5-2, the SRVCI contains two subfields. One subfield contains the internal address of the destination of the signalling control message. The other field contains the RVCI belonging to the data connection of this signalling control message. Like described above, is the value of the RVCI of the signalling cells always the same (for example the suggested 00000). This means that whenever in the broadcast cell the RVCI value 00000 is used to allocate a timeslot, this timeslot is used to carry the signalling message (or a part of the message). The internal address is added to make it possible to the PRPs to examine if the signalling control message is addressed to the UTE (of this MPS) or not. The internal address

is the mapping of the world-wide unique address of the end-user (MPS) on a local unique (and much shorter) address (for example the MAC address). The usage of this internal address is not the only solution to the recognition problem of the PRP of signalling messages received form the MSS. Every signalling message is uniquely identified by its call reference value which corresponds (on an unique way) to a certain channel (VPI/VCI combination), like described earlier in the protocol description of the MSS. The PRP could use this identification, but then higher layer functionality's (SAAL) need to be implemented in the PRP.
The other subfield is the RVCI data path of the associated signalling message. This subfield is specially included in the SRVCI to inform the PRP (of a certain MPS) about which timeslot(s) is associated with the transportation of the user data. The PRP extracts all the timeslots identified by this RVCI. The PRP does not have to distinghuis between different calls/connections in progress. The PRP only has to pass all the ATM cells belonging to this MPS to the UTE. The UTE is able to distinghuis between the different calls/connections based on the VPI/VCI combination of the ATM cells. Despite the fact that one of the assumptions of the MAC was that only one call could be handled per MPS (rather strange assumption if you want to be able to execute multimedia applications), is this restriction not carried through by the PRP. My opinion is that the MAC must discard this one call assumption if we want to make the demonstrator successful (and not only working). The RVCI subfield only needs to be included in the call/connection establishment phase, because ones the call/connection has been setup all the needed information is present at the various locations.

Figure 5-3 depicts the flow diagram of the MSS ⇒ PRP protocol description. The text in the boxes shortly describes the associated results of Step 3. The arrows indicate the direction of execution. If the last box does not contain a continue message then this implies that it is the end of the message flow within this specific protocol description.

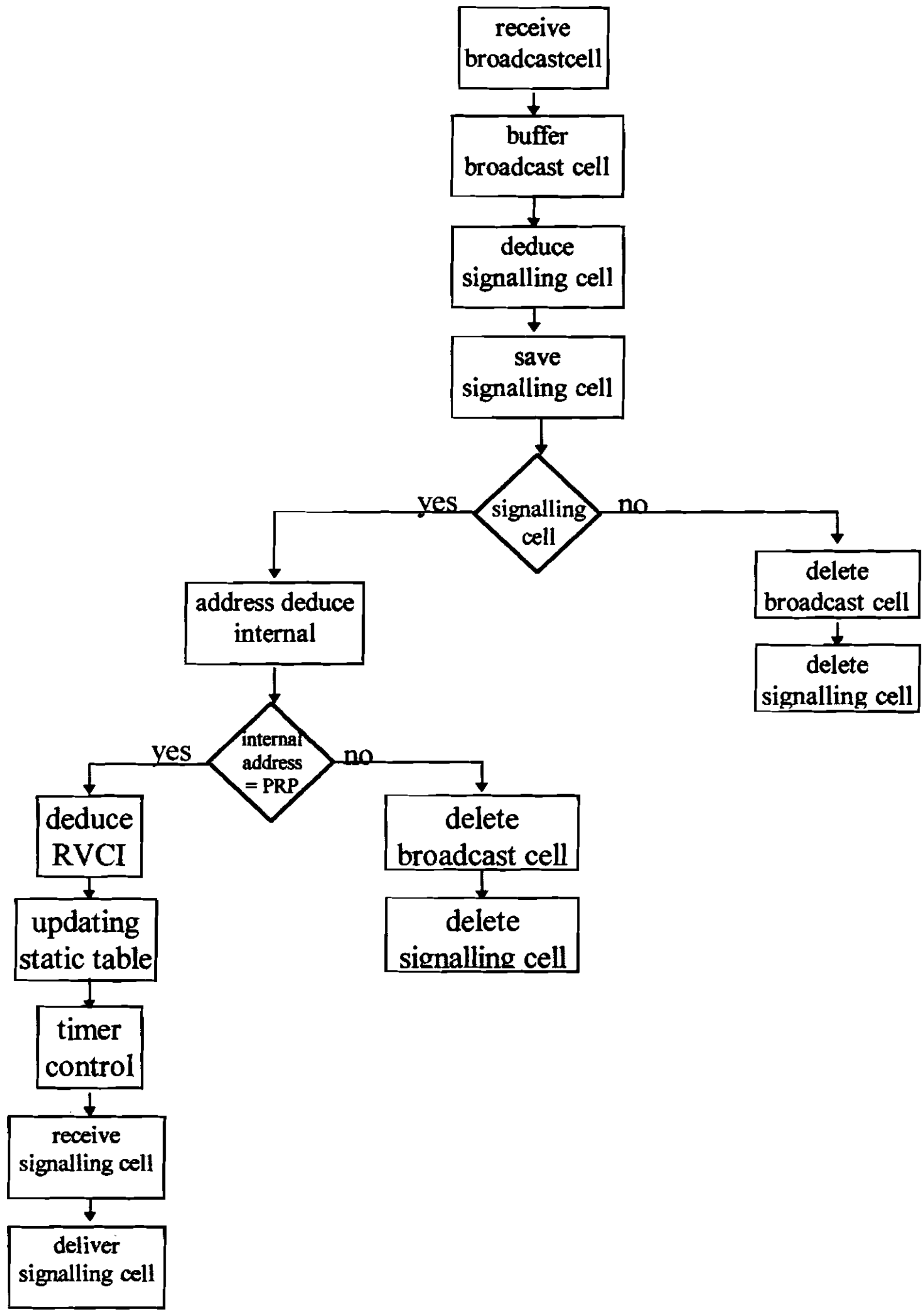


Figure 5-3: MSS-PRP protocol flow diagram

5.4.5.3 UTE ⇒ PRP protocol description

This protocol runs whenever the PRP receives a message from the UTE.
Caused by the splitting, depending upon the receiving direction of the message, only a subset of the total number of scenarios is applicable for the UTE ⇒ PRP protocol description.

The protocol description discriminates between the first signalling cell and the consecutive signalling cells. This is done because when the first signalling cell is received a distinction has to be made between two situations. The distinction is based on whether or not a timeslot is reserved for this signalling cell (indicated by the field inuse (see Section 5.4.5.4) in the static table). If no timeslot is reserved then the MAC layer needs to content for a timeslot in the uplink frame. Ones the first signalling cell has been transmitted, the reservation stays active for the duration of the signalling message. Two problems occur:

1. what is the duration of the signalling message (depends on kind of signalling message);
2. what to do when no timeslot is free for this signalling message.

To solve the above mentioned problems two different timers are introduced. The first one is timer Tzzz. Timer Tzzz is introduced to measure the time between two successive signalling cells (if applicable). If the UTE sends a signalling message to the PRP, then this is a continuous flow of ATM cells (identified by VPI/VCI= 0/5). The timer Tzzz is started on the arrival of an ATM-signalling cell. If the timer Tzzz exceeds a default value then the PRP decides that the whole signalling message has been transported and releases the signalling channel in the uplink frame. The default value of Tzzz (theoretical duration of ATM cell) is set to a short period, depending on the throughput time of the lower layers (de-serialiser and physical layer). By introducing this reservation depending on expiry of timer Tzzz, the number of contentions for a signalling timeslot in the uplink frame is (under normal circumstances) reduced to only one contention.

The Figure 5-4 illustrates the usage of timer Tzzz.

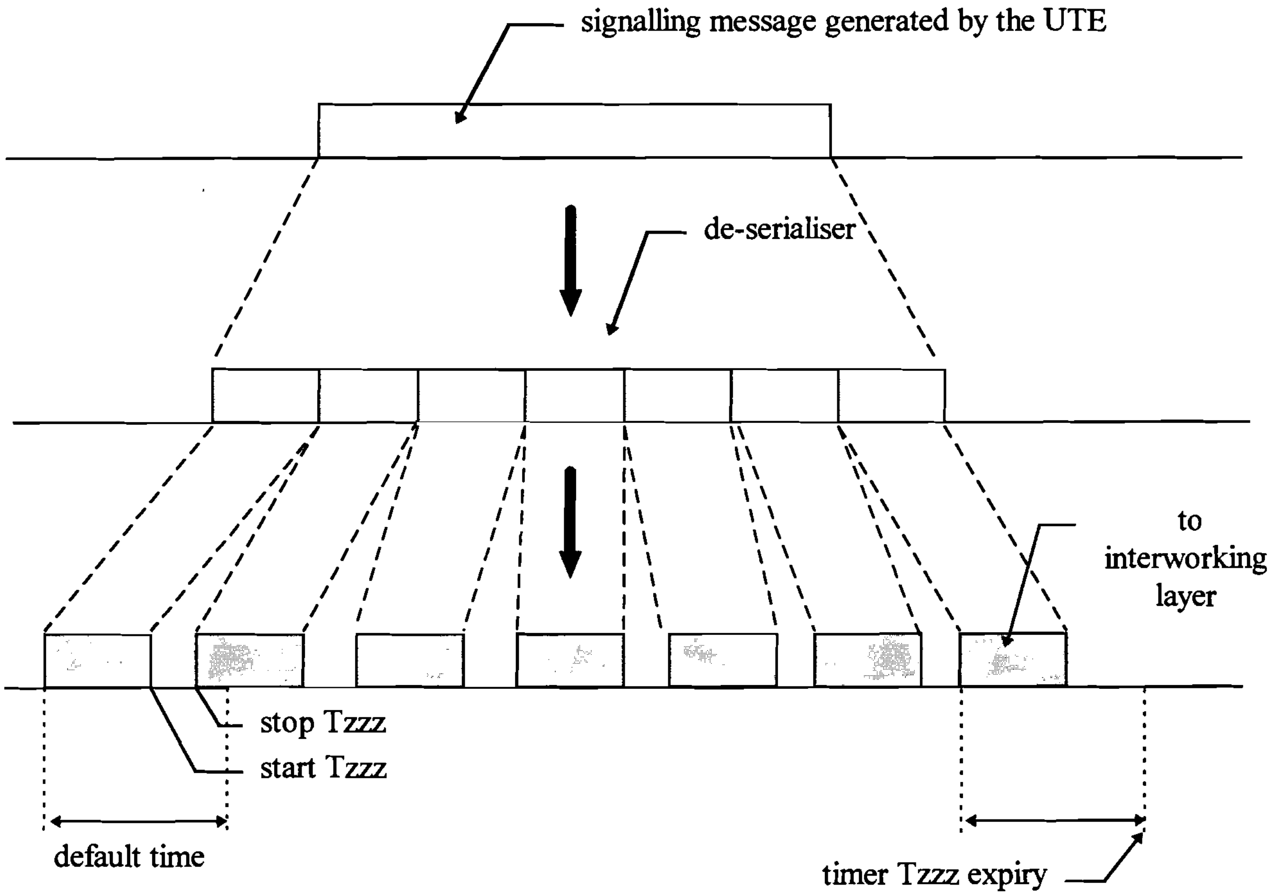


Figure 5-4: explanation timer Tzzz

The second problem (what to do when no timeslot is free for this signalling message) arises whenever another MPS is already using the uplink signalling timeslot. A timer Tyyy, is used to limit the time given to the MAC to content for an uplink signalling timeslot to prevent deadlock.

Figure 5-5 depicts the flow diagram of the UTE ⇒ PRP protocol description. The text in the boxes shortly describes the associated activities. The arrows show the direction of execution. If the last box does not contain a continue message then this implies that it is the end of the message flow within this specific protocol description.

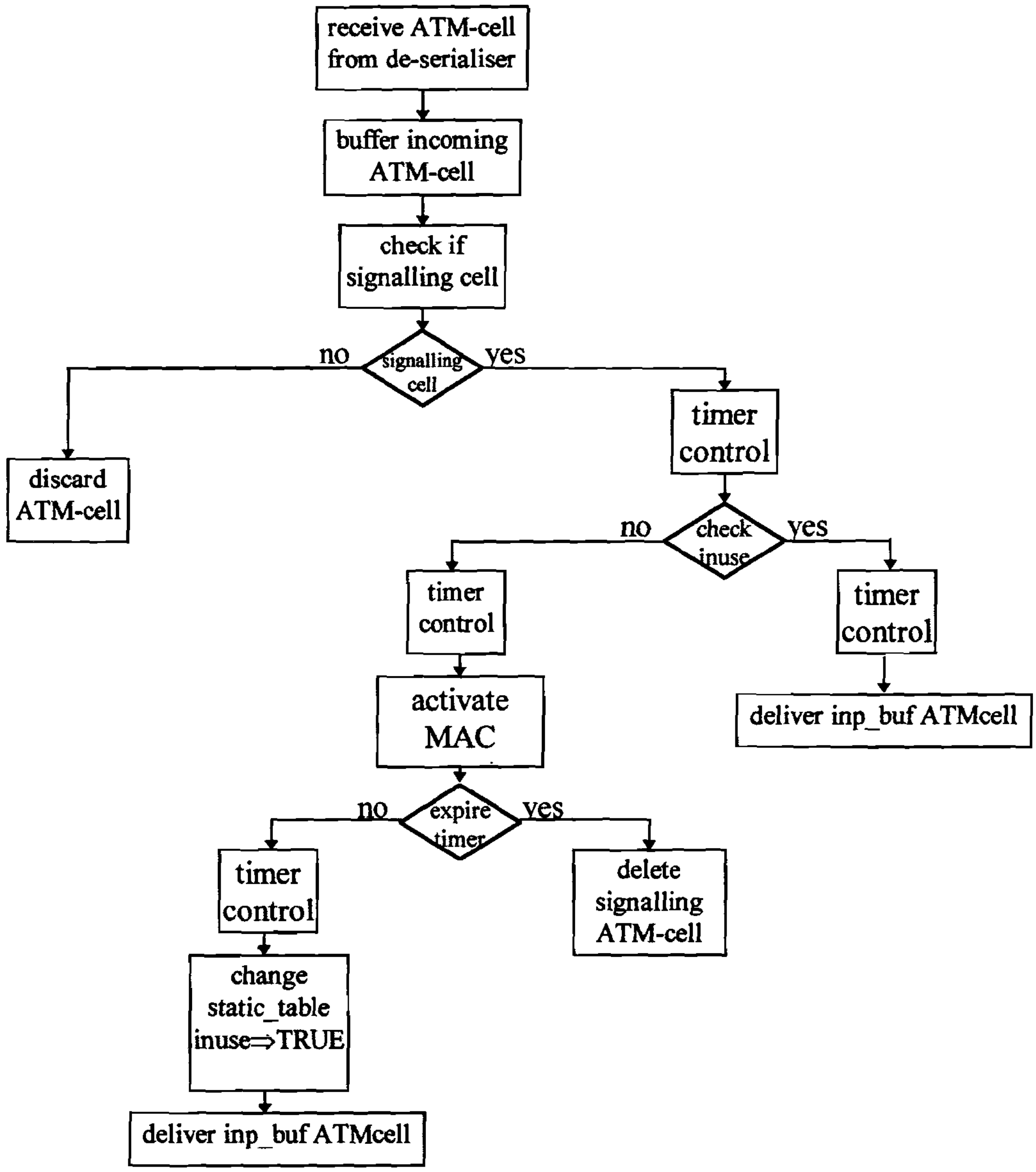


Figure 5-5: protocol description UTE⇒ PRP

5.4.5.4 Static table needed in the PRP protocol

The static table in the PRP is used to maintain the information needed to execute the protocol in a proper way. The static table maintains the information (that may be) needed by the MAC layer. Information which could be of interest to the MAC is:

- mapping of VPI/VCI on a radio VCI (RVCI);
- calculated delay: maximum delay which an ATM cell of a specific call/connection may undergo (in both the uplink and downlink direction);
- in-use field, indicating whether or not a timeslot is reserved.

5.4.5.5 Timers

The only timers that are being used in the PRP are the previously described Tyyy and Tzzz.

5.4.5.6 Problems originated from the protocol descriptions

Whenever the MAC layer in the MPS is not able to reserve a signalling timeslot before expiry of timer Tuuu, then the accompanying signalling ATM cell is deleted. Meaning that a part of the signalling message is deleted. It is possible, on the other hand, that the MAC is able to reserve a signalling uplink timeslot for the next signalling ATM cell of the same message. The interworking layer of the MSS receives this incorrect signalling message. After receiptance, the MSS checks if the message is received (transmitted) correctly (error_control_of_message). If an error has occurred (missing of part of signalling message) then the MSS can decide to ask the MPS to retransmit the signalling message.
Because of the length of the address of the MPS, a mapping is introduced in the MSS of the MPS address on a local unique internal address. This mapping needs a kind of user registration, dynamically or statically (chosen for a dynamical approach).

6. Conclusions and recommendations

The protocol descriptions derived provide a good view of the expected complexity of the interworking layer concerning call/connection handling.

The derived protocol descriptions of the MSS and the MPS describe (in detail) the actions that the interworking layer needs to carry out. These actions concern the call/connection establishment and call/connection release. Based on the high level protocol descriptions, it should be possible to implement the protocols in software.

Before being able to solve the problems that occur when determining interworking functions, a large amount of knowledge of ATM networks is required. Because of the many currently executed actions in standardization of ATM signalling (executed by ETSI and ATM Forum), it was difficult to choose from several different options. For example, it is not clear what will be the status of ATM signalling by the time the demonstrator will be presented. The current status of todays ATM backbones, concerning the signalling, is that the network is not able to support features as: QoS surveillance, VPI/VCI negotiation, traffic contract negotiation, VCI switching.

One of the difficulties of the MSS protocol descriptions derived is the checking of the QoS included in the SETUP message, it is likely that in the near future (within 5 years) no ATM based-backbone will be able to support the QoS surveillance.

Because of the dependence of the interworking layer on the MAC layer, one point to be addressed is to define a clear service access point between these layers. These should be independent of future changes of the MAC protocols. By doing so, the interworking layer can be developed only depending on the pre-defined service access point.

The high level protocol specification is developed in such a way that it holds all the information that is needed by the MAC layer (seen from authors view).

Further actions concerning the interworking layer in the demonstrator can take two possible directions. The first direction describes the actions concerning the enhancement of the interworking layer (not only call/connection establishment en release handling is needed). This will finally lead to implementation of the interworking layer. An alternative approach is to examine the possibility to use an existing network unit (for example an adapted ATM-switch) and modify it to carry out the additional functionality's needed. Important issues concerning the final selection of the followed approach may depend on terms like: complexity; costs; feasibility; and needed implementation time.

It is very important to have good connections with the suppliers of ATM components. These are aware of the future ATM developments and can advise in taking certain actions that depend on expectations concerning ATM developments.

Certain actions that are executed in the interworking layer of the MSS are already developed in the Q.2931 protocol. In order to reduce implementation effort it is possible to achieve a source code of the Q.2931. In order to adapt it to the needs of the MEDIAN system.

An additional feature of the protocol description is that it also includes the specification of additional timers. These timers make it possible to solve certain temporary regression of the radio channel.

References

[1] The ATM Forum,
ATM User-Network Interface (UNI) specification, version 3.1
London: Prentice-Hall, 1995
ISBN: 0-13-393828-X

[2] Martin De Prycker,
Asynchronous Transfer Mode, Solution for Broadband ISDN, third edition
United Kingdom: Prentice-Hall, 1995
ISBN: 0-13-342171-6

[3] William Stallings,
ISDN and Broadband ISDN, second edition
New York: Macmillan Publishing Company,1992
ISBN: 0-02-415475-X

[4] Gilbert Held,
Internetworking LAN's and WAN's: concepts, techniques and methods
Chichester: Wiley, 1993
ISBN: 0-471-93568-9

[5] Harrell J. Van Norman,
LAN-WAN optimization techniques
London: Artech House, 1992
ISBN: 0-89006-617-5

[6] Esaki Hiroshi, Tsuda Y.,
Evaluation of High Speed Multimedia Communication Architecture in ATM networks
IEICE Trans. Commun., Vol. E77-B, 1994

[7] D. Raychaudhuri
ATM Based Transport Architecture for Multiservices Wireless Personal Communication Networks
Princeton: IEEE, 1994
ISBN: 0-7803-1825-0

[8] D. Raychaudhuri, Wilson N.
Multimedia Transport in Next Generation Personal Communication Networks
Princeton: IEEE, 1993

[9] B.G. Marchent
Performance Evaluation of Signalling Procedures for 3rd Generation Mobile Systems
Hampshire: IEEE, 1195
ISBN: 0-8186-6902-0

[10] S. Baldisseri
An Integrated Model of the ATM Switch for Control and Management Purpose
Rome: IEEE, 1994
ISBN: 0-7803-1825-0

FACULTEIT DER ELEKTROTECHNIEK
TECHNISCHE UNIVERSITEIT EINDHOVEN
VAKGROEP TELECOMMUNICATIE EC

INTERWORKING FUNCTIONS IN A WIRELESS BROADBAND LOCAL AREA NETWORK (PART 2)

BY

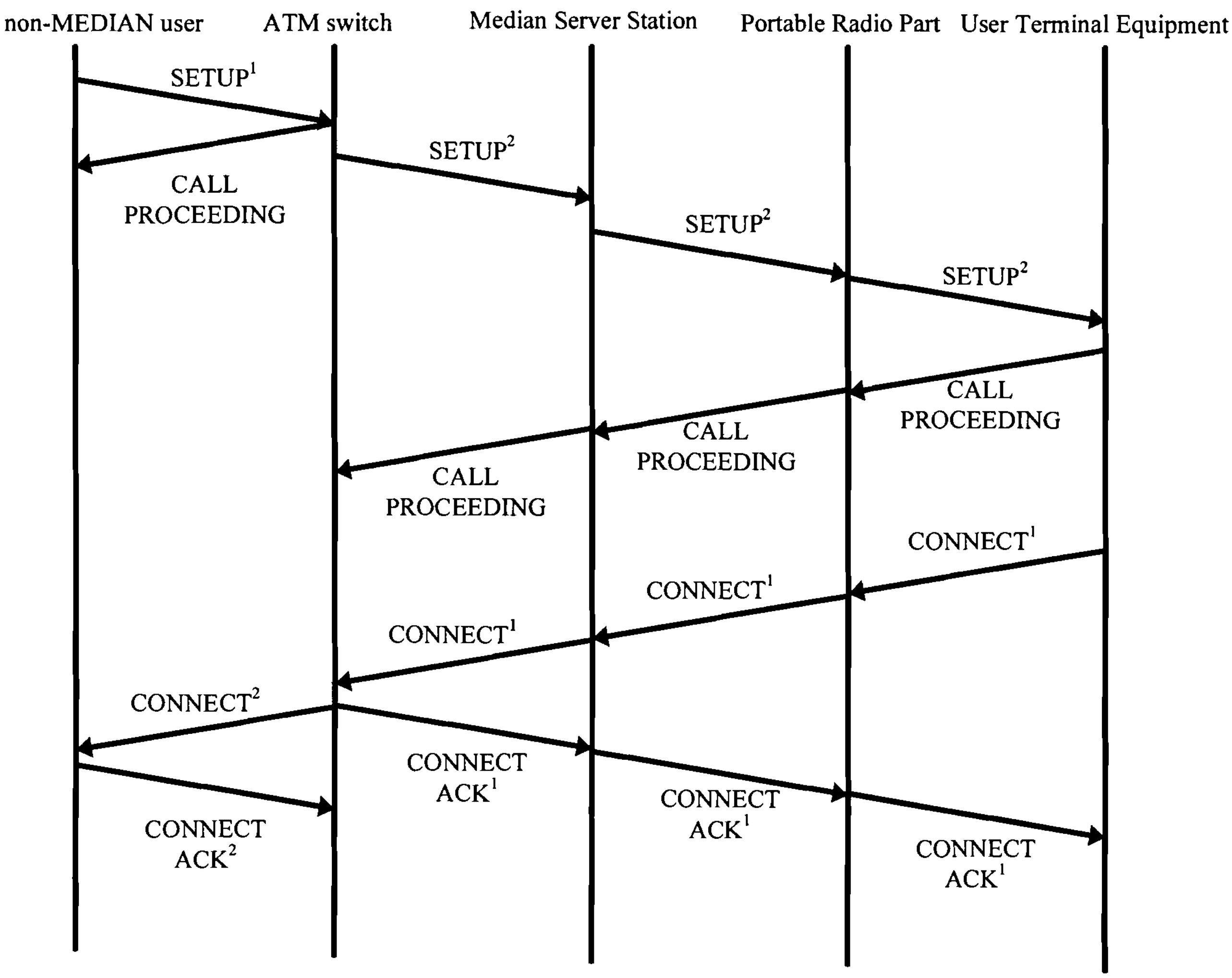
ING. M.J.H. PETERS

Verslag van het afstudeerwerk
uitgevoerd van 15-11-1995 tot 27-6-1996
Afstudeerhoogleraar: prof.dr.ir. G. Brussaard
Begeleider: dr.ir. P.F.M. Smulders

I_Setup_a

Scenario specification:

- the MSS is able to support the traffic (specified in the ATM traffic descriptor) and the requested QoS, included in the SETUP message;
- the UTE is able to support the traffic (specified in the ATM traffic descriptor) and the requested QoS, included in the SETUP message;
- connection can be established between the calling non-MEDIAN user and the UTE.

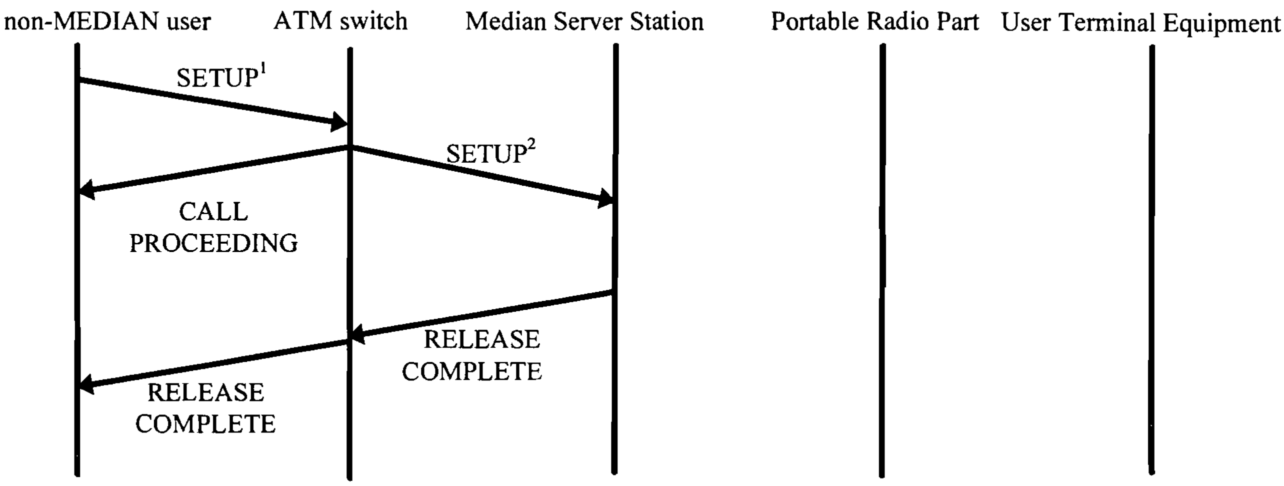


A change in indices of a specific message means that the contents of the information elements of the specific message are changed (this is also applied in all the other scenarios).

I_Setup_b

Scenario specification:

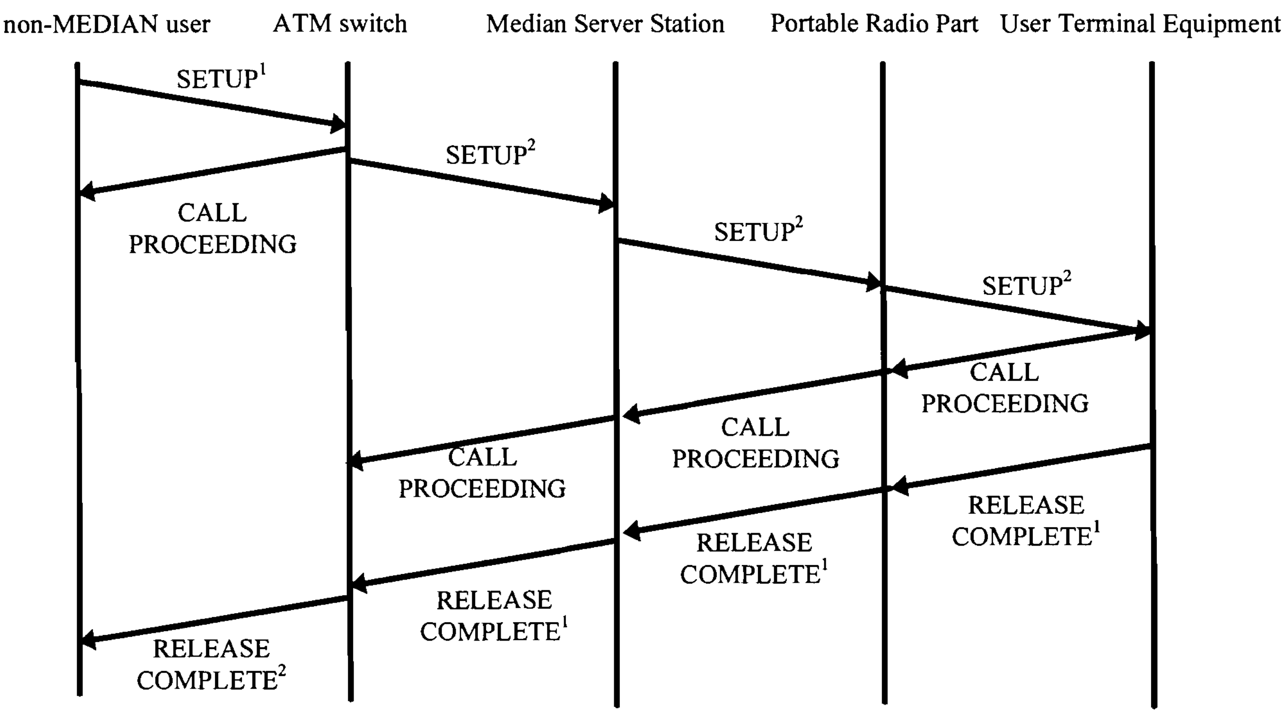
- the MSS cannot support the traffic (specified in the ATM traffic descriptor) and/or the requested QoS, included in the SETUP message;
- connection cannot be established between the calling non-MEDIAN user and the UTE.



I_Setup_c

Scenario specification:

- the MSS can support the traffic (specified in the ATM traffic descriptor) and the requested QoS, included in the SETUP message;
- the UTE cannot support the traffic (specified in the ATM traffic descriptor) and/or the requested QoS, included in the SETUP message;
- connection cannot be established between the calling non-MEDIAN user and the UTE.

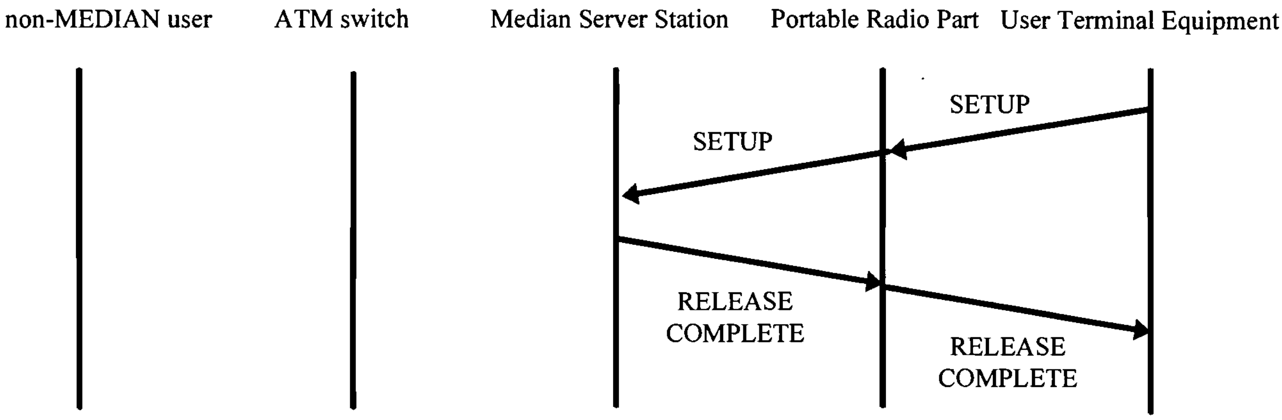


Appendix A

II_Setup_a

Scenario specification:

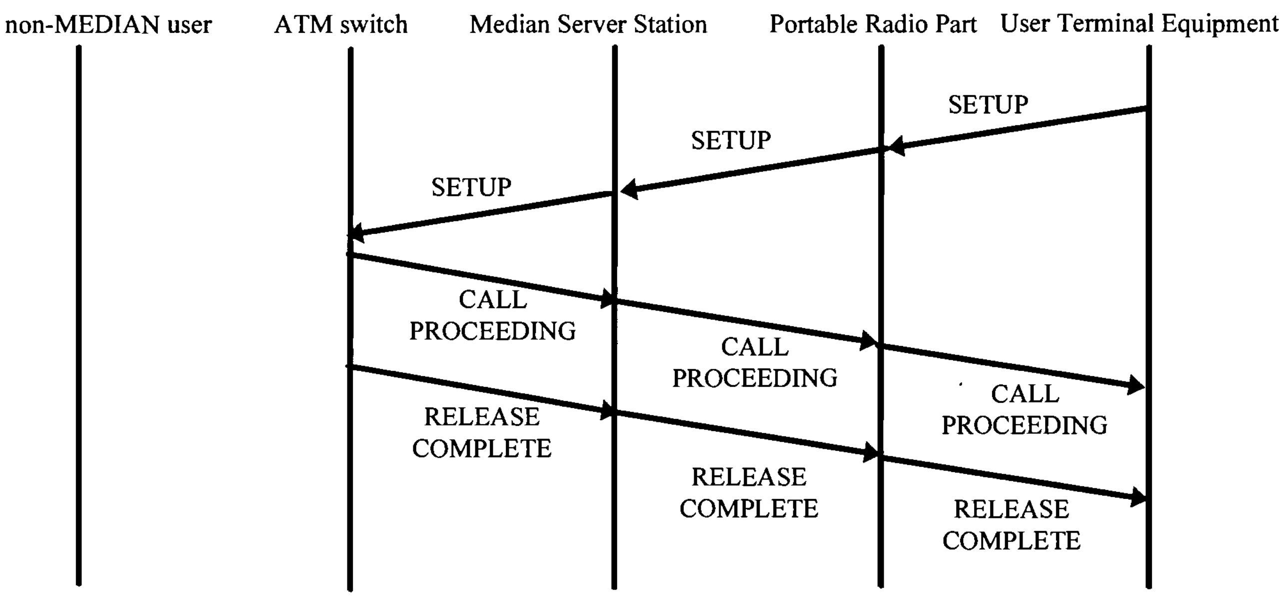
- the MSS cannot support the traffic (specified in the ATM traffic descriptor) and/or the requested QoS, included in the SETUP message initiate by the UTE;
- connection cannot be established between the calling UTE and the non-MEDIAN user.



II_Setup_b

Scenario specification:

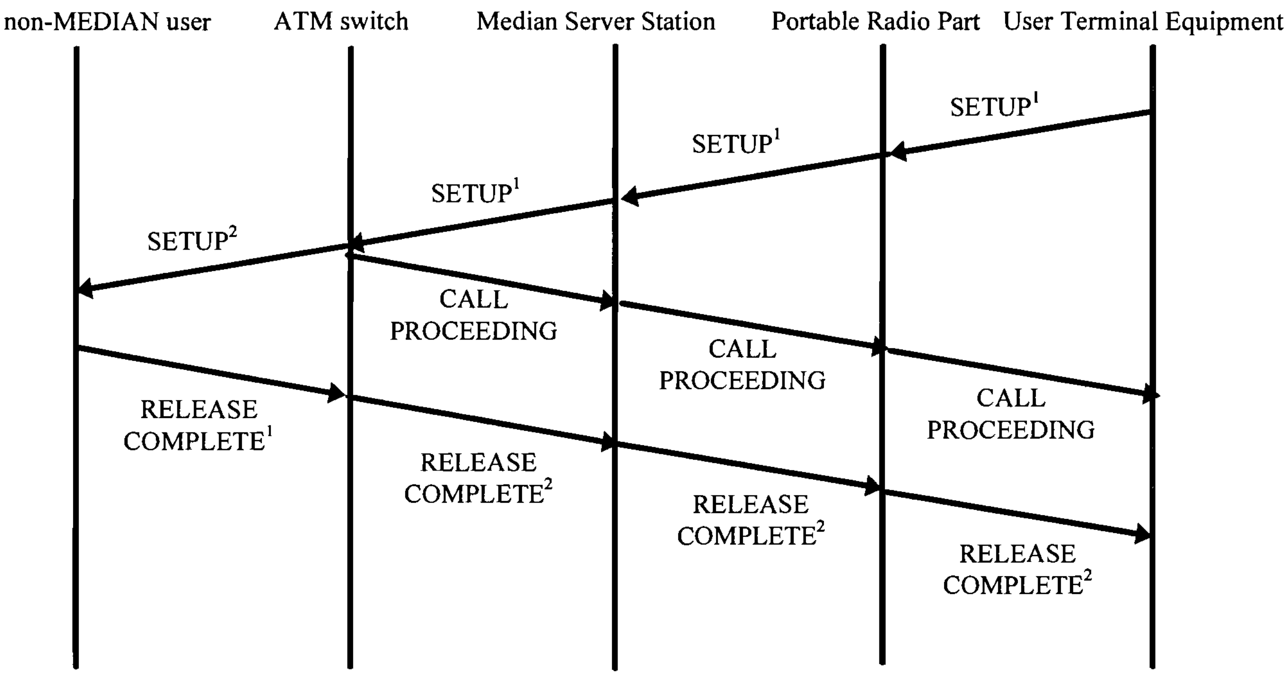
- the MSS can support the traffic (specified in the ATM traffic descriptor) and the requested QoS, included in the SETUP message initiate by the UTE;
- the public network (ATM switch) cannot support the traffic (specified in the ATM traffic descriptor) and/or the requested QoS, included in the SETUP message initiate by the UTE;
- connection cannot be established between the calling UTE and the non-MEDIAN user;



II_Setup_c

Scenario specification:

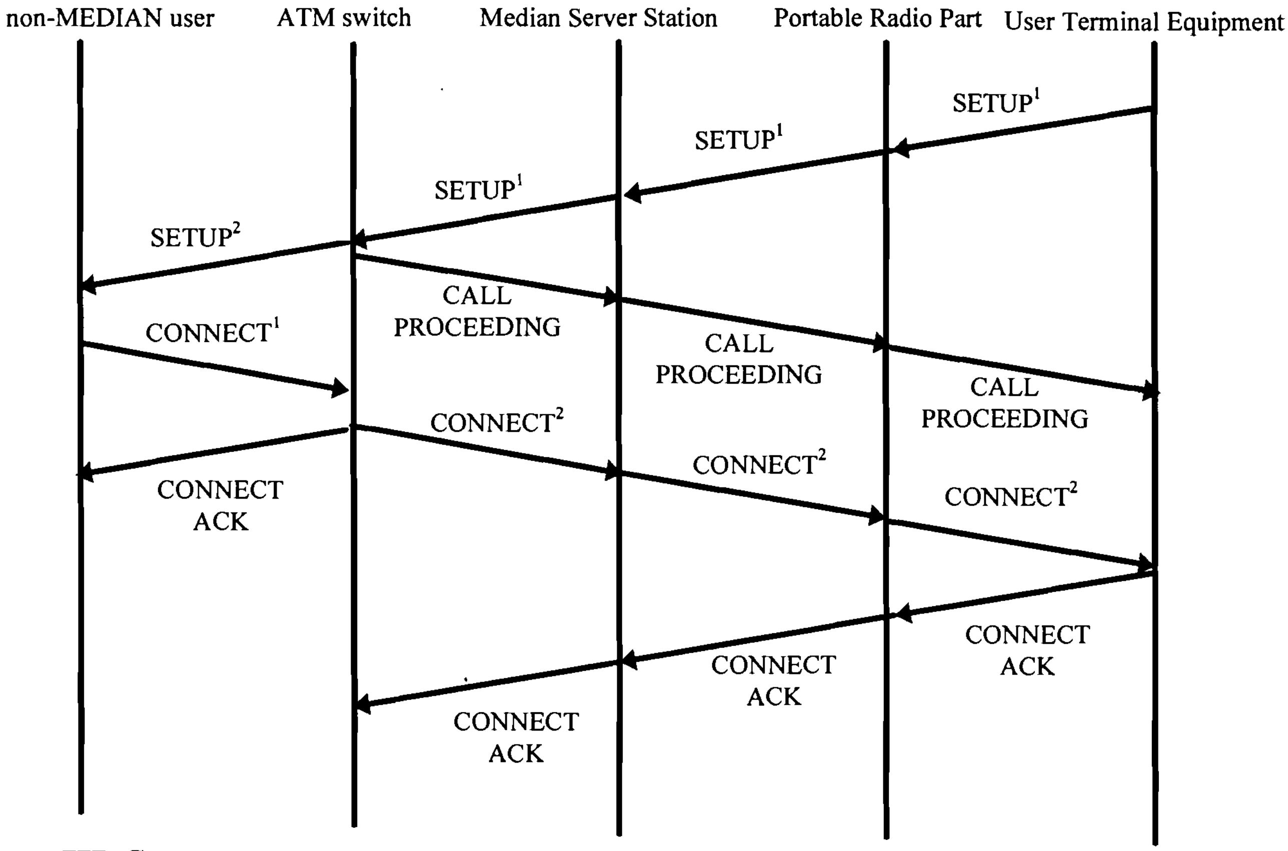
- the MSS and the public network (ATM switch) can support the traffic (specified in the ATM traffic descriptor) and the requested QoS;
- the called non-MEDIAN user cannot support the traffic (specified in the ATM traffic descriptor) and/or the requested QoS, included in the SETUP message initiate by the UTE;
- connection cannot be established between the calling UTE and the called non-MEDIAN user.



II_Setup_d

Scenario specification:

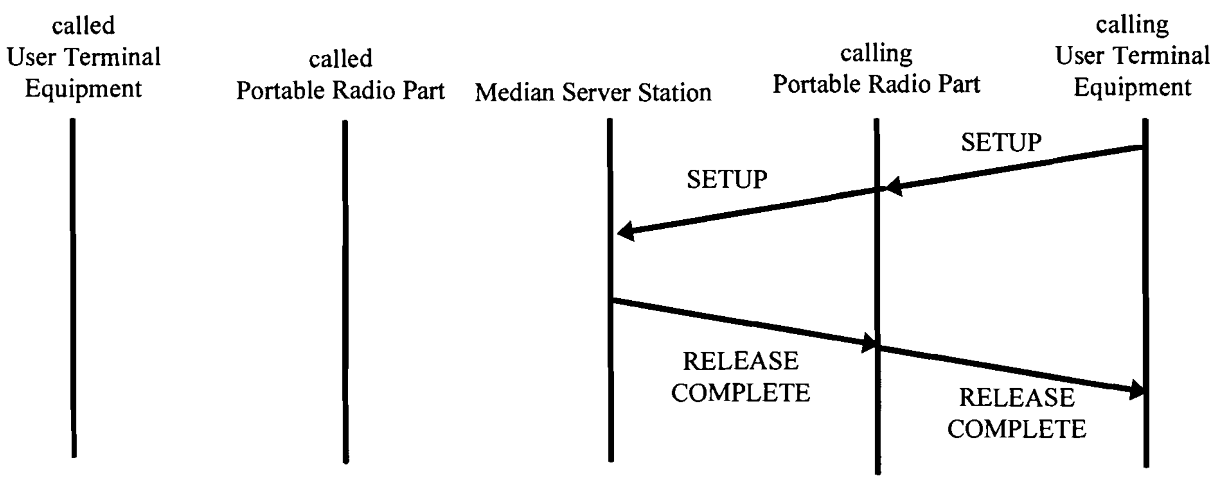
- the MSS, the public network (ATM switch), and non-MEDIAN user can support the traffic (specified in the ATM traffic descriptor) and the requested QoS;
- connection can be established between the calling UTE and the called non-MEDIAN user;



III_Setup_a

Scenario specification:

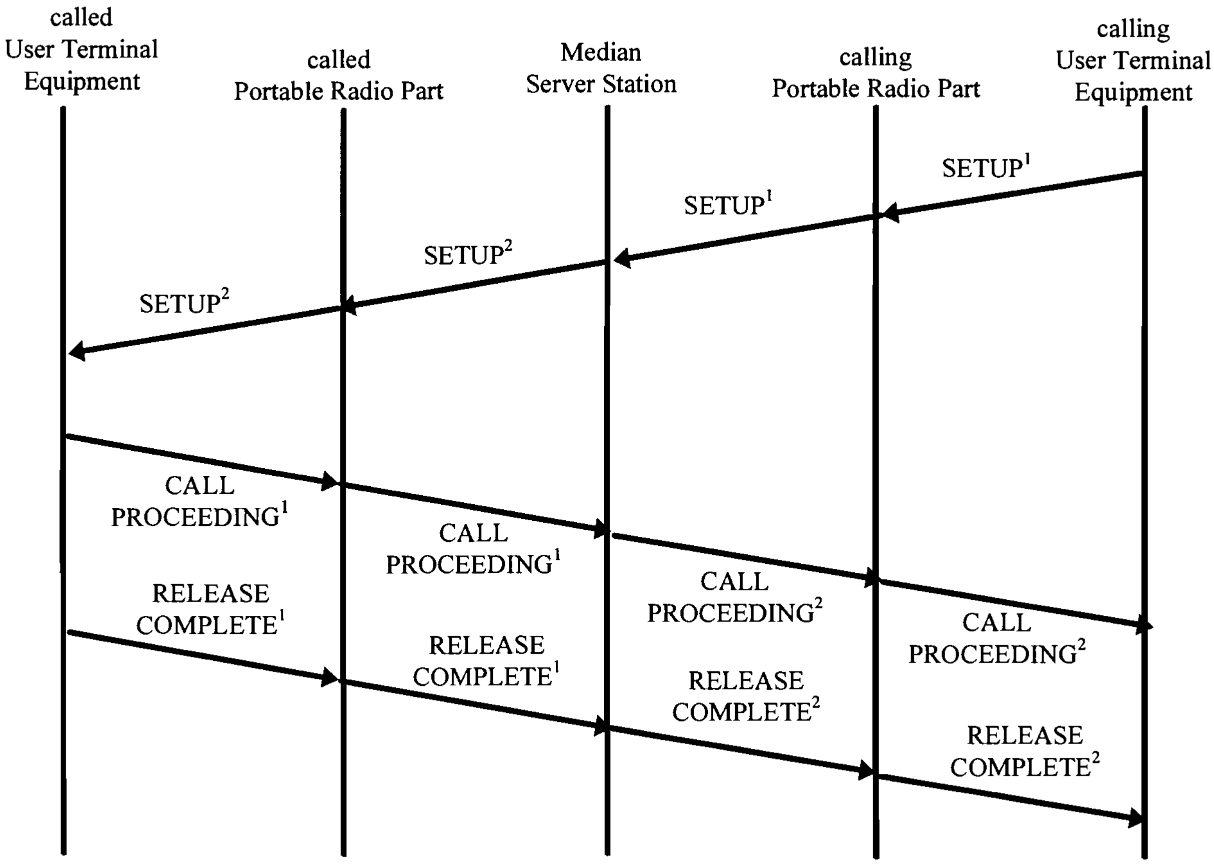
- the MSS can not support the traffic specified in the ATM traffic descriptor and/or the requested QoS;
- connection can not be established between the calling UTE and the called UTE;



III_Setup_b

Scenario specification:

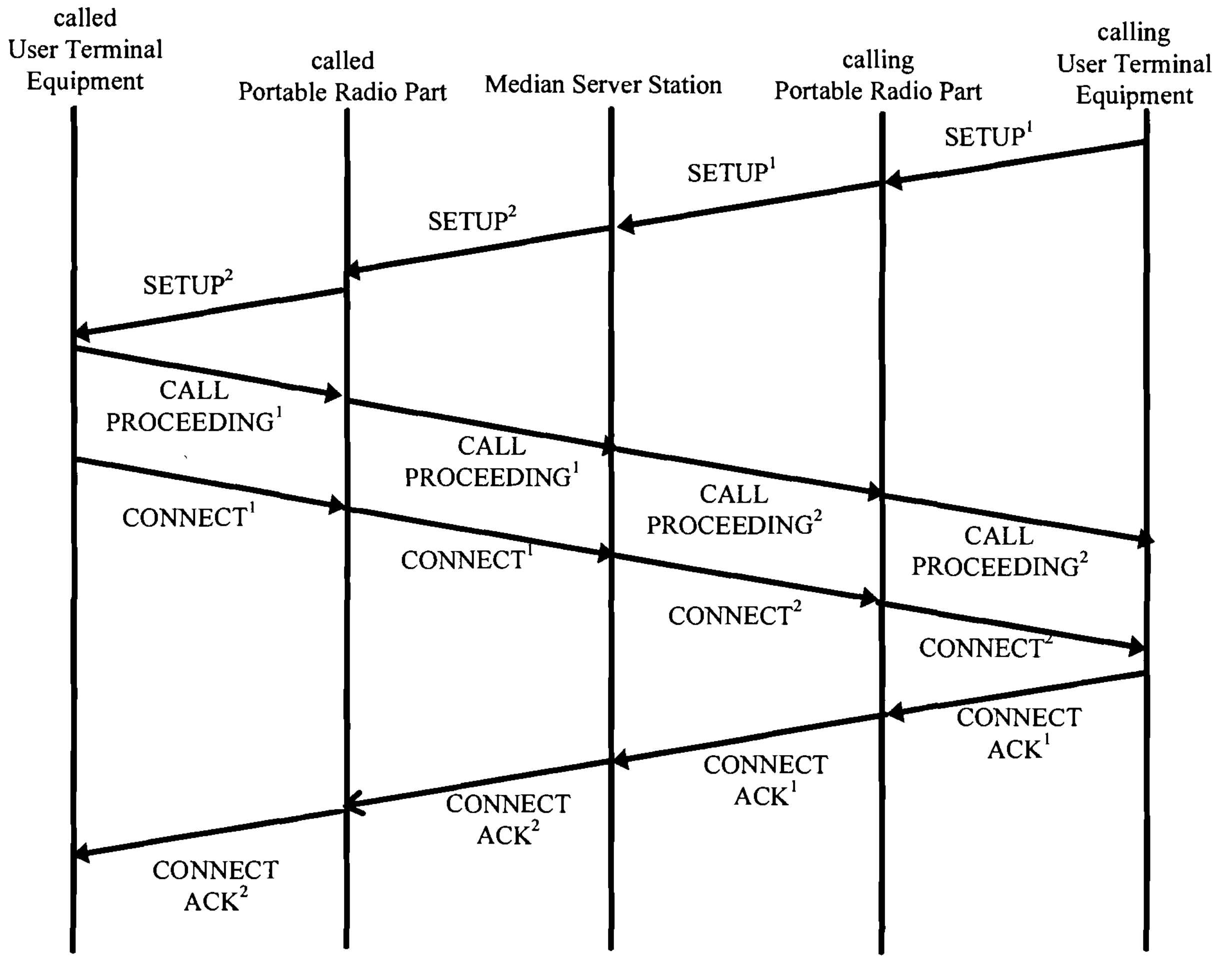
- the MSS can support the traffic (specified in the ATM traffic descriptor) and the requested QoS, included in the SETUP message initiate
- the called UTE rejects the requested call establishment;
- connection cannot be established between the calling UTE and the called UTE.



III_Setup_c

Scenario specification:

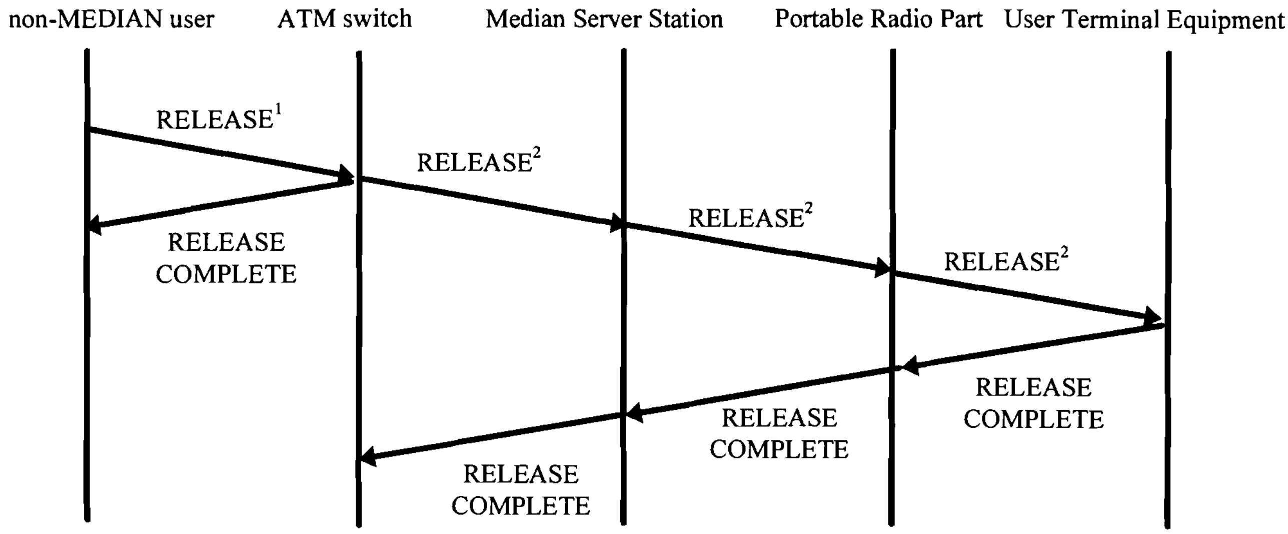
- the MSS and the UTE of the called party can support the traffic (specified in the ATM traffic descriptor) and the requested QoS;
- connection can be established between the calling UTE and the called UTE;



IV_Release_a

Scenario specification:

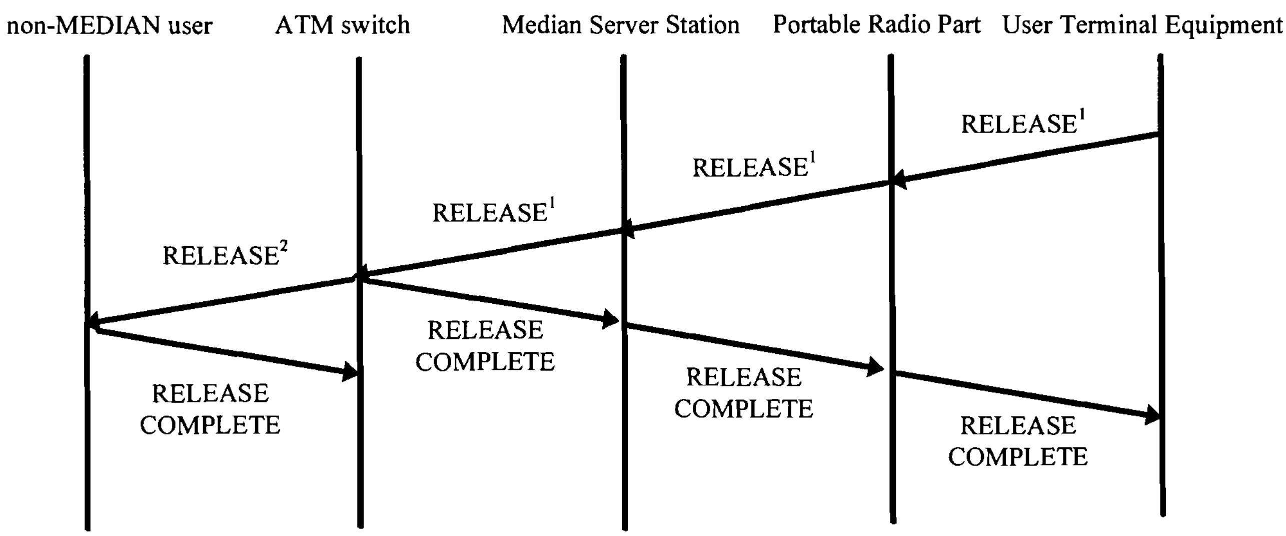
- non-MEDIAN user initiates call release of connection with MPS;
- connection will be released.



IV_Release_b

Scenario specification:

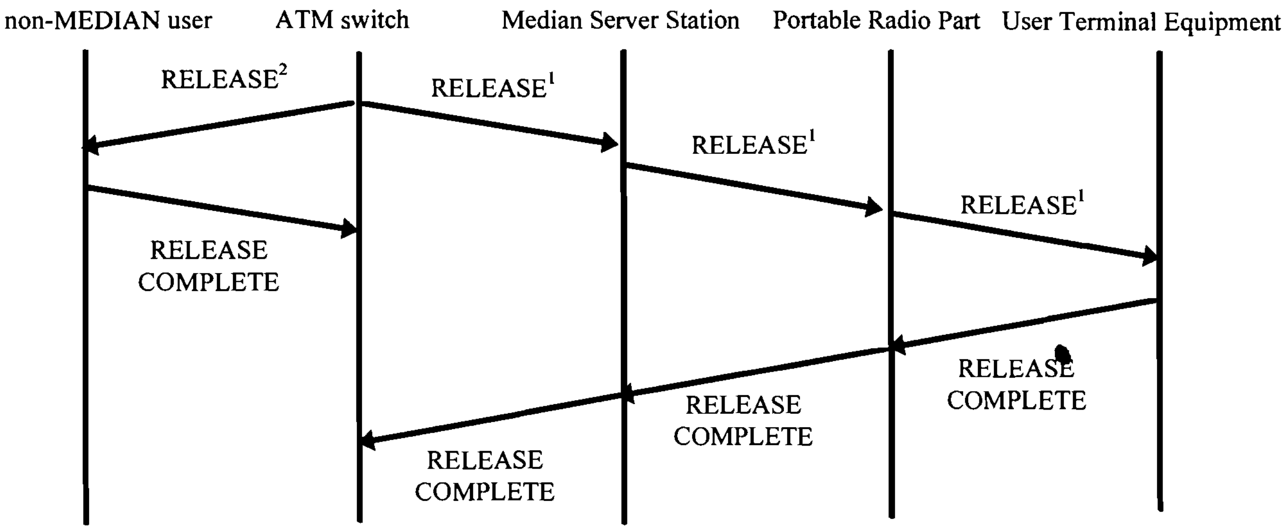
- MPS initiates call release of connection with ATM end-user;
- connection will be released.



IV_Release_c

Scenario specification:

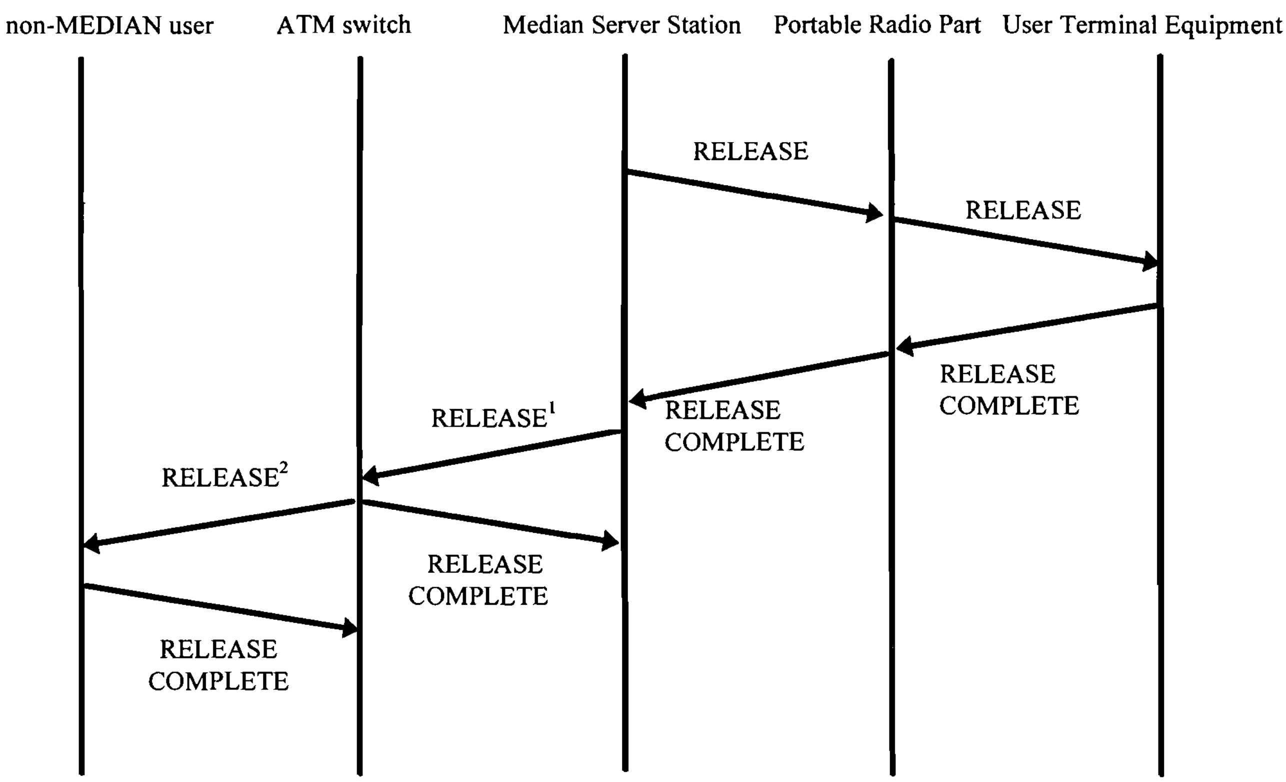
- Public network (ATM switch) initiates call release of connection between a MEDIAN user and a non-MEDIAN user;
- connection will be released.



IV_Release_d

Scenario specification:

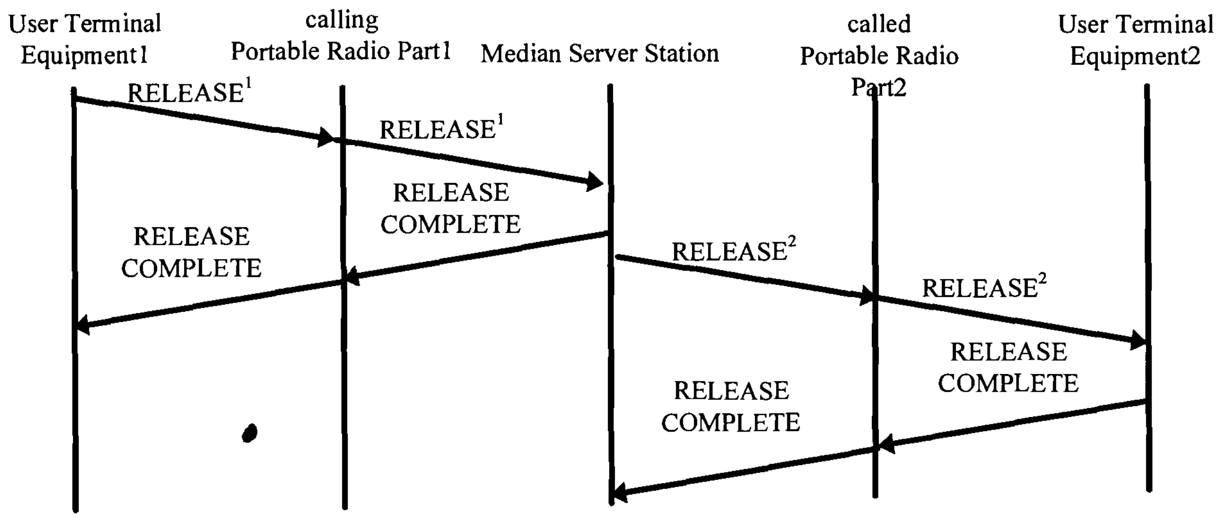
- MSS initiates call release of connection between a MEDIAN user and a non-MEDIAN user;
- connection will be released.



IV_Release_e

Scenario specification:

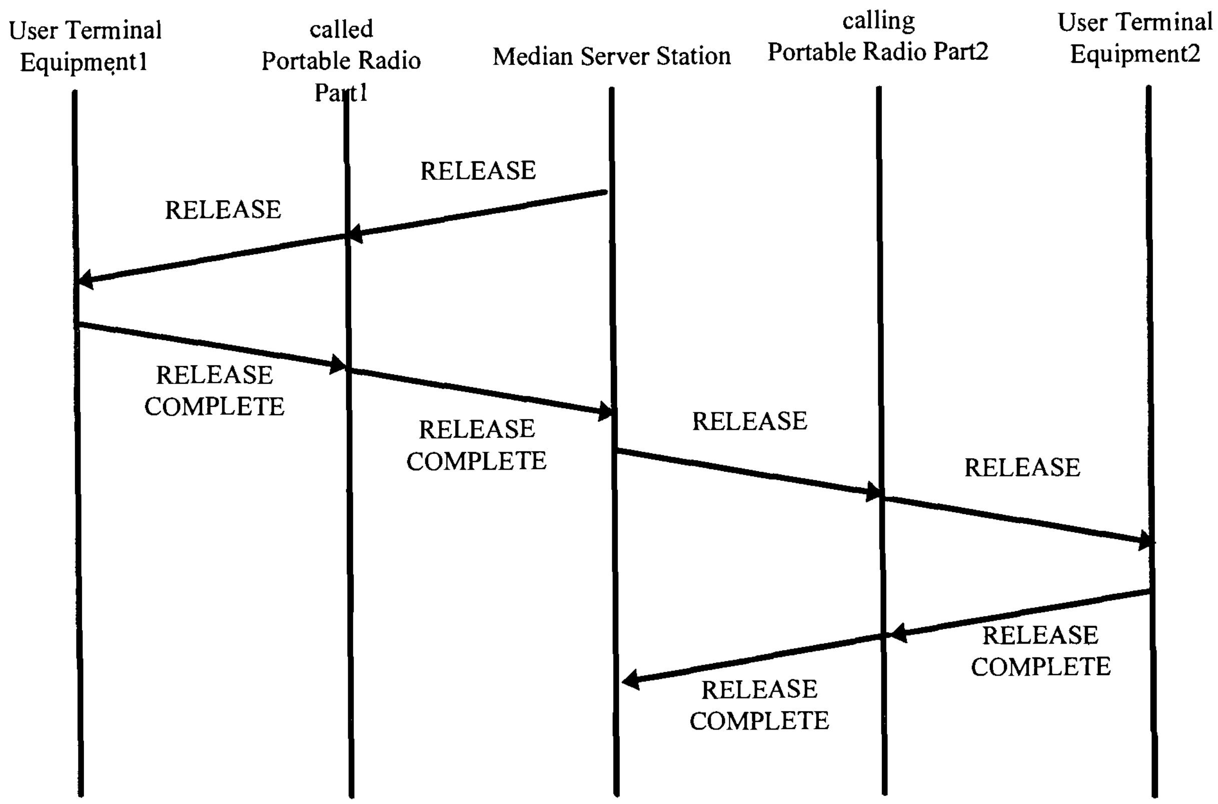
- MPS initiates call release of connection with another MPS;
- connection will be released.



IV_Release_f

Scenario specification:

- MSS initiates call release of connection between two MEDIAN users;
- connection will be released.



I_Setup_a
Scenario specification:
- the MSS is able to support the traffic (specified in the ATM traffic descriptor) and the requested QoS, included in the SETUP message;
- the UTE is able to support the traffic (specified in the ATM traffic descriptor) and the requested QoS, included in the SETUP message;
- connection can be established between the calling non-MEDIAN user and the UTE.

I_Setup_b
Scenario specification:
- the MSS cannot support the traffic (specified in the ATM traffic descriptor) and/or the requested QoS, included in the SETUP message;
- connection cannot be established between the calling non-MEDIAN user and the UTE.

I_Setup_c
Scenario specification:
- the MSS can support the traffic (specified in the ATM traffic descriptor) and the requested QoS, included in the SETUP message;
- the UTE cannot support the traffic (specified in the ATM traffic descriptor) and/or the requested QoS, included in the SETUP message;
- connection cannot be established between the calling non-MEDIAN user and the UTE.

II_Setup_a
Scenario specification:
- the MSS cannot support the traffic (specified in the ATM traffic descriptor) and/or the requested QoS, included in the SETUP message initiate by the UTE;
- connection cannot be established between the calling UTE and the non-MEDIAN user.

II_Setup_b
Scenario specification:
- the MSS can support the traffic (specified in the ATM traffic descriptor) and the requested QoS, included in the SETUP message initiate by the UTE;
- the public network (ATM switch) cannot support the traffic (specified in the ATM traffic descriptor) and/or the requested QoS, included in the SETUP message initiate by the UTE;
- connection cannot be established between the calling UTE and the non-MEDIAN user;

II_Setup_c
Scenario specification:
- the MSS and the public network (ATM switch) can support the traffic (specified in the ATM traffic descriptor) and the requested QoS;
- the called non-MEDIAN user cannot support the traffic (specified in the ATM traffic descriptor) and/or the requested QoS, included in the SETUP message initiate by the UTE;
- connection cannot be established between the calling UTE and the called non-MEDIAN user.

II_Setup_d
Scenario specification:
- the MSS, the public network (ATM switch), and non-MEDIAN user can support the traffic (specified in the ATM traffic descriptor) and the requested QoS;
- connection can be established between the calling UTE and the called non-MEDIAN user;

III_Setup_a
Scenario specification:
- the MSS can not support the traffic specified in the ATM traffic descriptor and/or the requested QoS;
- connection can not be established between the calling UTE and the called UTE;

III_Setup_b
Scenario specification:
- the MSS can support the traffic (specified in the ATM traffic descriptor) and the requested QoS, included in the SETUP message initiate
- the called UTE rejects the requested call establishment;
- connection cannot be established between the calling UTE and the called UTE.

III_Setup_c
Scenario specification:
- the MSS and the UTE of the called party can support the traffic (specified in the ATM traffic descriptor) and the requested QoS;
- connection can be established between the calling UTE and the called UTE;

IV_Release_a
Scenario specification:
- non-MEDIAN user initiates call release of connection with MPS;
- connection will be released.

IV_Release_b
Scenario specification:
- MPS initiates call release of connection with ATM end-user;
- connection will be released.

IV_Release_c
Scenario specification:
- Public network (ATM switch) initiates call release of connection between a MEDIAN user and a non-MEDIAN user;
- connection will be released.

IV_Release_d
Scenario specification:
- MSS initiates call release of connection between a MEDIAN user and a non-MEDIAN user;
- connection will be released.

IV_Release_e
Scenario specification:
- MPS initiates call release of connection with another MPS;
- connection will be released.

IV_Release_f
Scenario specification:
- MSS initiates call release of connection between two MEDIAN users;
- connection will be released.

Appendix B

I_Setup_a

1. calling ATM end-user wants to establish a connection with a MPS and sends a SETUP message to the called MPS (through the ATM-switch, MSS, PRP and finally UTE);
2. ATM-switch accepts call and sends a CALL PROCEEDING message to the ATM end-user and a SETUP message (with VPI/VCI indicated in the Connection Identifier Information Element) to the called MPS (through the MSS ,PRP and UTE);

3. MSS-PHY layer receives information from ATM-switch, and delivers the bit stream to the MSS-ATM layer;
4. MSS-ATM layer delivers the signalling ATM cells without the header to the MSS-SAAL;
5. MSS-SAAL reassembles the signalling ATM cells and delivers the signalling message (in particular SETUP message) to the MSS-Interworking layer;
6. the MSS-Interworking layer receives the incoming signalling (SETUP) message and buffers the message;
7. the MSS-Interworking layer extracts the Protocol discriminator Information Element:
   - must be 0000 1001 (Q.2931);
8. the MSS-Interworking layer extracts the Call reference Information Element:
   - flag must be 0 (message sent from the side that originates the call reference);
   - call reference value (23 bit) saved in static table;
9. the MSS-Interworking layer extracts the Message type Information Element:
   - message type must be 0000 0101 (SETUP message);
   - change state of connection from Null(U0) into Call Present (U6) state in static table;
   - save external in 'originated' field in the static table;
10. the MSS-Interworking layer extracts the Message length Information Element:
    - extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself)
11. the MSS-Interworking layer extracts the ATM traffic descriptor Information Element:
    - deduce information about the forward/backward peak cell rate (CLP=0 and CLP=0+1) (if present);
    - deduce information about the forward/backward sustainable cell rate (CLP=0 and CLP=0+1) (if present);
    - deduce information about the forward/backward maximum burst size (CLP=0 and CLP=0+1) (if present);
    - save information in static data base;
12. the MSS-Interworking layer extracts the Broadband bearer capability Information Element:
    - bearer class (BCOB-A, BCOB-C, BCOB-X(most likely to be used));
    - traffic type (CBR, VBR);
    - timing requirements (no indication, end-to-end timing required, end-to-end timing not required;
    - susceptibility to clipping (yes, no);
    - user plane connection (point-to-point, point-to-multipoint);
    - save information in static data base;
13. the MSS-Interworking layer extracts the Called Party number Information element and deduces the ATM end system address or E.164 address;
14. the MSS-Interworking layer extracts the Connection Identifier Information Element:
    - extract VPI/VCI;
    - VPI/VCI is maintained in the static table;
15. the MSS-Interworking layer extracts the QoS information element, which can contain information concerning the:
    - cell error ratio;
    - severly-errored cell block ratio;
    - cell misinsertion ratio;
    - cell transfer delay;
    - mean cell transfer delay;
    - cell delay variation;
16. save information in static data base;
17. the MSS-Interworking layer checks if message is errorless (Yes assumed);

18. the MSS-Interworking layer checks if requested QoS can be provided, using the saved QoS information. There is additional information needed about the behaviour of the overall system for example BER, bursty characteristic of errors, propagation delay, transport delay of ATM cells in ATM layer, buffer capacity in MSS etc. It should be possible to deduce the necessary information from the information maintained in the static data base;
19. the MSS-Interworking layer checks if able to support combination of traffic parameters in both the directions (MSS⇒ MPS and MPS⇒ MSS). The MSS-Interworking layer maintains the information of the static list, therefore having the data to derive the free slots (and RVCI's) in the uplink and the downlink;
20. the MSS-Interworking layer checks if VPI/VCI is acceptable (available);
21. the MSS-Interworking layer decides if the end-user is:
    - an MEDIAN system end-user (is valid in this scenario);
    - not an MEDIAN system end-user (initiate call release);

    Check can be done because the MSS has an address table supplied by the address registration procedures of the management system containing, among other things, ATM end system address or E.164 address (which, within the MEDIAN system, is mapped on an internal shorter address (int_adress);
22. QoS check=okay AND traffic parameter check=okay AND VPI/VCI available=okay AND address check=okay (because this is scenario I_Setup_a);
23. MSS-Interworking layer translates the ATM end system address in an internal address;
24. MSS-Interworking layer saves internal address in static table;
25. MSS-Interworking layer unambiguous maps incoming VPI/VCI on a RVCI for user-data transport. This information is maintained in the static list (max. # of RVCI =32);
26. MSS-Interworking layer deduces time on which cells belonging to the selected RVCI, up- and downlink expire and retain these values in the static list (or just do selective queuing);
27. MSS-Interworking layer deduces statistical data (what ever this maybe) for static database;

    Parameters which could be of interest:
    - cell error ratio;
        - severly-errored cell block ratio;
        - cell misinsertion ratio;
        - cell transfer delay;
        - mean cell transfer delay;
    - cell delay variation;
    - total information flow uplink;
    - total information flow downlink;
    - uplink and downlink information flow per radio connection;
    - state a certain RVCI is in (null/call initiated/outgoing call proceeding/.../call abort/restart/restart request);
    - number of rejected calls on an average;
28. the MSS-Interworking layer starts timer T303;
29. IF no reaction to the SETUP message before expiry of timer T303 THEN initiate call/connection release;
30. no modification of the SETUP message, the Median Server Station seems to be transparent to the SETUP message in this scenario (only reading and checking of the SETUP message). The buffered Setup message is delivered to the MSS-SAAL layer;
31. after the MSS-SAAL layer has done the necessary actions and segments the signalling (SETUP) message into ATM-information containers, it delivers the ATM-information containers to the MSS-ATM layer;
32. the MSS-ATM layer adds the ATM-header to the ATM-information container;
33. the ATM cells and all the necessary information being of importance to the MSS-MAC are send to the MSS-MAC;
34. MSS-MAC delivers the bitstream to the MSS-PHY layer (SAAL, MAC and PHY take care of the 'reliable' transportation of the SETUP message to the PRP);

35. the 'signalling (SETUP) message' is transported to the PRP using the radiochannel;

36. the PRP-PHY layer delivers the constructed bitstream to the PRP-MAC layer;
37. the PRP-MAC delivers the broadcast cell to the PRP-Interworking layer;
38. the PRP-MAC delivers (if any) signalling cell(s) (VPI/VCI=0/5) to the Interworking layer;
39. the PRP-Interworking layer buffers the broadcast cell;
40. the PRP-Interworking layer buffers the signalling cell(s);
41. the PRP-Interworking layer deduces signalling broadcast cell from the broadcast cell;

42. the PRP-Interworking layer checks if signalling broadcast cell is nil;
43. if nil then delete broadcast cell and signalling broadcast cell, else continue (continue is assumed);
44. the PRP-Interworking layer deduces the RVCI and the internal address (or ATM end system address or E.164 address) from the signalling broadcast cell;
45. the PRP-Interworking layer checks if the internal address is the address of the PRP (MPS)
    - if unequal then do not proceed and delete the buffered ATM signalling cells (if any) and the broadcast cell;
    - if yes then proceed (yes is assumed);
46. the PRP-Interworking layer saves the RVCI in the static table;
47. the PRP-Interworking layer sets the in-use field in the static table to yes;
48. the PRP-Interworking layer starts timer concerning reaction of the UTE;
49. if timer expiry before answer of UTE then set the in-use field to no;
50. the PRP-Interworking layer feeds the buffered ATM (signalling) cells to the ATM-layer (where, among other things, the ATM cells are serialised into a bitstream);
51. PRP-ATM layer delivers bitstream to the PRP-PHY layer;

52. 'SETUP message' is transported to the UTE layer, using the optical channel;
53. how the UTE exactly reacts is not of importance (because these functionality's are already implemented on the interface card mounted in the UTE);
54. the UTE decides to send a CALL PROCEEDING message (because this is scenario I_Setup_a) to the PRP (which finally has to arrive at the ATM switch) when the requested call establishment has been initiated and no more call establishment information will be accepted;

55. 'CALL PROCEEDING message' is transported to the PRP using the optical channel;

56. the PRP-PHY layer delivers the received bitstream to the PRP-ATM layer;
57. the PRP-ATM layer deserialises the bitstream into ATM cells;
58. the PRP-ATM layer delivers signalling ATM cell (if any) to the PRP-Interworking layer;
59. the PRP-Interworking layer buffers (in a receive buffer) the incoming signalling ATM cells (buffering is needed because the PRP-MAC ,maybe , first has to content for a free (signalling) slot);
60. the PRP-Interworking layer starts a timer to measure the time between two successive signalling cells;
61. on expiry of the timer, set the static table inuse value to false;
62. the PRP-Interworking layer checks if the static table inuse value is true or false;
63. if inuse is false then
    - ask the MAC layer to content for an uplink (signalling) timeslot and start timer for contention time;
    - on expiry of timer delete buffered signalling ATM cell
    - the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the PRP-Interworking layer stops the timer;
    - the PRP-Interworking layer changes the static table field inuse to true;
64. if inuse is true then the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
65. the PRP-Interworking layer stops the timer;
66. the PRP-Interworking layer checks if there is another signalling ATM cell in the de-serialiser (or better, waits until the PRP-ATM layer delivers a signalling cell before timer expiry);
67. if no signalling ATM cell available in de-serialiser then change static table field inuse to false (end);
68. if ATM cell available in de-serialiser then
    - the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the PRP-Interworking layer continues this procedure until there is no signalling ATM cell in the de-serialiser;
69. the PRP-MAC layer delivers the bitstream to the PRP-PHY layer;

70. the 'CALL PROCEEDING message' is transported to the MSS over the radio channel;

71. at the MSS the MSS-PHY layer delivers the constructed bitstream to the MSS-MAC layer;
72. the MSS-MAC layer delivers the signalling message to the MSS-ATM layer;
73. the MSS-ATM layer removes the ATM cell header and delivers the ATM information cell to the MSS-SAAL;
74. the MSS-SAAL reassembles (among other things) the signalling message. The MSS-SAAL delivers the signalling message to the MSS-Interworking layer;

75. the MSS-Interworking layer receives the incoming signalling (CALL PROCEEDING) message and buffers the message;
76. the MSS-Interworking layer extracts the Protocol discriminator Information Element:
    - must be 000010001 (Q.2931)
77. the MSS-Interworking layer extracts the Call reference Information Element:
    - flag must be 1 (message sent to side that originates the call reference (calling party));
    - deduce call reference value
78. the deduced call reference value is also maintained in static table, and uniquely identifies the call the signalling message refers to;
79. the MSS-Interworking layer extracts the Message type Information Element:
    - message type must be 0000 0010 (CALL PROCEEDING message);
    - change state of connection from Call Present (U6) into Incoming Call Proceeding (U3) state in static table;
    - stops T303;
    - start T310;
80. IF no reaction to the CALL PROCEEDING message before expiry of timer T310 THEN initiate call/connection release;
81. the MSS-Interworking layer extracts the Message length Information Element;
82. extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself);
83. the MSS-Interworking layer checks if message is errorless (Yes assumed);
84. MSS-Interworking layer deduces statistical data (what ever this maybe) for static database;
85. the MSS-Interworking layer delivers the received (buffered) message to the MSS-SAAL;
86. the MSS-SAAL delivers the segmented CALL PROCEEDING message to the MSS-ATM layer (synchronised);
87. the MSS-ATM layer delivers the bitstream to the MSS-PHY layer;
88. 'signalling (CALL PROCEEDING) message' is delivered to the ATM-switch (actions concerning CALL PROCEEDING message within the MEDIAN system are finished);

89. after the UTE has checked whether is wants to accept the call (Yes assumed because I_Setup_a), it sends a CONNECT message to the PRP (which finally has to arrive at the ATM switch)(I assume that the MSS reserves a RVCI for the signalling and for the user-data transport);

90. the PRP-PHY layer delivers the received bitstream to the PRP-ATM layer;
91. the PRP-ATM layer deserialises the bitstream into ATM cells;
92. the PRP-ATM layer delivers signalling ATM cell (if any) to the PRP-Interworking layer;
93. the PRP-Interworking layer buffers (in a receive buffer) the incoming signalling ATM cells (buffering is needed because the PRP-MAC ,maybe , first has to content for a free (signalling) slot);
94. the PRP-Interworking layer starts a timer to measure the time between two successive signalling cells;
95. on expiry of the timer, set the static table inuse value to false;
96. the PRP-Interworking layer checks if the static table inuse value is true or false;
97. if inuse is false then
    - ask the MAC layer to content for an uplink (signalling) timeslot and start timer for contention time;
    - on expiry of timer delete buffered signalling ATM cell
    - the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the PRP-Interworking layer stops the timer;
    - the PRP-Interworking layer changes the static table field inuse to true;
98. if inuse is true then the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
99. the PRP-Interworking layer stops the timer;
100. the PRP-Interworking layer checks if there is another signalling ATM cell in the de-serialiser (or better, waits until the PRP-ATM layer delivers a signalling cell before timer expiry);
101. if no signalling ATM cell available in de-serialiser then change static table field inuse to false (end);

102.if ATM cell available in de-serialiser then
- the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
- the PRP-Interworking layer continues this procedure until there is no signalling ATM cell in the de-serialiser;

103.the PRP-MAC layer delivers the bitstream to the PRP-PHY layer;
104.the 'CONNECT message' is transported to the MSS over the radio channel;

105.the MSS-PHY layer delivers the constructed bitstream to the MSS-MAC layer;
106.the MSS-MAC layer delivers the signalling (CONNECT message) to the MSS-ATM layer;
107.the MSS-ATM layer removes the ATM cell header and delivers the ATM information cell to the MSS-SAAL;
108.the MSS-SAAL reassembles (among other things) the received 'signalling message'. The SAAL delivers the signalling (CONNECT) message to the MSS-Interworking layer;
109.the MSS-Interworking layer receives the incoming signalling (CONNECT) message and buffers the message;
110.the MSS-Interworking layer extracts the Protocol discriminator Information Element:
- must be 000010001 (Q.2931)

111.the MSS-Interworking layer extracts the Call reference Information Element:
- flag must be 1 (message sent to side that originates the call reference (calling party));

112.the deduced call reference value is also maintained in the static table, and uniquely identifies the call the signalling message refers to;
113.the MSS-Interworking layer extracts the Message type Information Element:
- message type must be 0000 0111 (CONNECT message);
- change state of connection from Incoming Call Proceeding(U9) state into Connect Request(U8) in static table;
- stop timer T310;
- start timer T313;

114.IF no reaction to the CONNECT message before expiry of timer T313 THEN initiate call release;
115.the MSS-Interworking layer extracts the Message length Information Element:
116.extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself);
117.the MSS-Interworking layer checks if message is errorless (Yes assumed);
118.MSS-Interworking layer deduces statistical data (what ever this maybe) for static database;
119.the saved CONNECT message is delivered it to the MSS-SAAL;
120.the MSS-SAAL delivers the segmented signalling (CONNECT) message to the MSS-ATM layer;
121.the MSS-ATM layer delivers the bitstream to the MSS-PHY layer (actions concerning CONNECT message within the MEDIAN system are finished);

122.the ATM-switch receives the CONNECT message from the MPS and sends a CONNECT ACKNOWLEDGE message to the MPS and a CONNECT message to the calling ATM end-user;
123.CONNECT ACKNOWLEDGE message is transported to the MPS;

124.MSS-PHY layer delivers received bit stream to ATM layer;
125.MSS-ATM layer delivers the signalling ATM cells without the header to the MSS-SAAL;
126.MSS-SAAL reassembles the signalling ATM cells and delivers the signalling message (CONNECT ACKNOWLEDGE message) to the MSS-Interworking layer;
127.the MSS-Interworking layer receives the incoming signalling (CONNECT ACKNOWLEDGE) message and buffers the message;
128.the MSS-Interworking layer extracts the Protocol discriminator Information Element:
- must be 000010001 (Q.2931)

129.the MSS-Interworking layer extracts the Call reference Information Element:
- flag must be 0 (message sent from side that originates the call reference (calling party));

130.the deduced call reference value is also saved in static table, and identifies the call the signalling message refers to;

131.the MSS-Interworking layer extracts the Message type Information Element:
- message type must be 0000 1111 (CONNECT ACKNOWLEDGE message);
- change state of connection from Outgoing Connect Request(U8) state into Active(U10) in the static table;
- stop timer T313;

132.the MSS-Interworking layer extracts the Message length Information Element;
133.the MSS-Interworking layer extracts the length of the message (excluding protocol discriminator, call reference, message type, message length indication itself);
134.the MSS-Interworking layer checks if message is errorless (Yes assumed);
135.no modification of the CONNECT ACKNOWLEDGE message, the MSS seems to be transparent to the CONNECT ACKNOWLEDGE message in this scenario;
136.MSS-Interworking layer deduces statistical data (what ever this maybe) for static database;
137.the buffered CONNECT ACKNOWLEDGE message is delivered to the MSS-SAAL layer;
138.after the MSS-SAAL layer has done the necessary actions and segmented the signalling message into ATM-information containers. The flow of ATM information containers is delivered to the MSS-ATM layer;
139.the MSS-ATM layer adds the ATM-header to the ATM-information container resulting in the same ATM cell corresponding to the arrived ATM-cell at the beginning (meaning that the signalling (CONNECT ACKNOWLEDGE) message is transparently transported through the MSS-Interworking layer);
140.the ATM cells and all the necessary information being of importance to the MSS-MAC are send to the MSS-MAC.;
141.MSS-MAC delivers the bitstream to the MSS-PHY layer (MAC and PHY take care of the 'reliable' transportation of the signalling message to the PRP);

142.'signalling (CONNECT ACKNOWLEDGE) message' is transported to the PRP using the radiochannel;

143.the PRP-PHY layer delivers the constructed bitstream to the PRP-MAC layer;
144.the PRP-MAC delivers the broadcast cell to the PRP-Interworking layer;
145.the PRP-MAC delivers (if any) signalling cell(s) (VPI/VCI=0/5) to the Interworking layer;
146.the PRP-Interworking layer buffers the broadcast cell;
147.the PRP-Interworking layer buffers the signalling cell(s);
148.the PRP-Interworking layer deduces signalling broadcast cell from the broadcast cell;
149.the PRP-Interworking layer checks if signalling broadcast cell is nil;
150.if nil then delete broadcast cell and signalling broadcast cell, else continue (continue is assumed);
151.the PRP-Interworking layer deduces the RVCI and the internal address (or ATM end system address or E.164 address) from the signalling broadcast cell;
152.the PRP-Interworking layer checks if the internal address is the address of the PRP (MPS)
- if unequal then do not proceed and delete the buffered ATM signalling cells (if any) and the broadcast cell;
- if yes then proceed (yes is assumed);

153.the PRP-Interworking layer saves the RVCI in the static table;
154.the PRP-Interworking layer sets the in-use field in the static table to yes;
155.the PRP-Interworking layer starts timer concerning reaction of the UTE;
156.if timer expiry before answer of UTE then set the in-use field to no;
157.the PRP-Interworking layer feeds the buffered ATM (signalling) cells to the ATM-layer (where, among other things, the ATM cells are serialised into a bitstream);
PRP-ATM layer delivers bitstream to the PRP-PHY layer;

158.CONNECT ACKNOWLEDGE message is transported to the called UTE, using the optical channel;
159.the 'CONNECT ACKNOWLEDGE message' arrives at the destination, UTE;
160.connection is established and user-data can be transmitted using the reserved RVCI;

Appendix B

I_Setup_b

The MSS needs to:

1. calling ATM end-user wants to establish a connection with a MPS and sends a SETUP message to the called MPS;
2. ATM switch accepts call and sends a CALL PROCEEDING message to the ATM end-user and a SETUP message (with VPI/VCI indicated in the Connection Identifier Information Element) to the called MPS;

3. MSS-PHY layer receives information from ATM-switch, and delivers the bit stream to the MSS-ATM layer;
4. MSS-ATM layer delivers the signalling ATM cells without the header to the MSS-SAAL;
5. MSS-SAAL reassembles the signalling ATM cells and delivers the signalling message (in particular SETUP message) to the MSS-Interworking layer;
6. the MSS-Interworking layer receives the incoming signalling (SETUP) message and buffers the message;
7. the MSS-Interworking layer extracts the Protocol discriminator Information Element:
   - must be 0000 1001 (Q.2931);
8. the MSS-Interworking layer extracts the Call reference Information Element:
   - flag must be 0 (message sent from the side that originates the call reference)
   - call reference value (23 bit) saved in static table;
9. the MSS-Interworking layer extracts the Message type Information Element:
   - message type must be 0000 0101 (SETUP message);
   - change state of connection from Null(U0) into Call Present (U6) state in static table;
   - save external in 'originated' field in the static table;
10. the MSS-Interworking layer extracts the Message length Information Element:
    - extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself)
11. the MSS-Interworking layer extracts the ATM traffic descriptor Information Element:
    - deduce information about the forward/backward peak cell rate (CLP=0 and CLP=0+1) (if present);
    - deduce information about the forward/backward sustainable cell rate (CLP=0 and CLP=0+1) (if present);
    - deduce information about the forward/backward maximum burst size (CLP=0 and CLP=0+1) (if present);
    - save information in static data base;
12. the MSS-Interworking layer extracts the Broadband bearer capability Information Element:
    - bearer class (BCOB-A, BCOB-C, BCOB-X(most likely to be used));
    - traffic type (CBR, VBR);
    - timing requirements (no indication, end-to-end timing required, end-to-end timing not required;
    - susceptibility to clipping (yes,no);
    - user plane connection (point-to-point, point-to-multipoint);
    - the MSS-Interworking layer extracts the Called party Number Information Element end;
    - save information in static data base;
13. the MSS-Interworking layer extracts the Called Party number Information element and deduces the ATM end system address;
14. the MSS-Interworking layer extracts the Connection Identifier Information Element:
    - extract VPI/VCI;
    - VPI/VCI is maintained in the static table;
15. the MSS-Interworking layer extracts the QoS information element, which can contain information concerning the:
    - cell error ratio;
    - severly-errored cell block ratio;
    - cell misinsertion ratio;
    - cell transfer delay;
    - mean cell transfer delay;
    - cell delay variation.
    - save information in static data base;
16. the MSS-Interworking layer checks if message is errorless (Yes assumed);

17. the MSS-Interworking layer checks if requested QoS can be provided, using the extracted QoS information. There is additional information needed about the behaviour of the overall system for example BER, bursty characteristic of errors, propagation delay, transport delay of ATM cells in ATM layer, buffer capacity in Median Server Station etc. It should be possible to deduce the necessary information from the information maintained in the static data base;
18. the MSS-Interworking layer checks if able to support combination of traffic parameters in both the directions (MSS⇒ MPS and MPS⇒ MSS). The MSS-Interworking layer maintains the information of the static list, therefore having the data to derive the free slots in the uplink and the downlink;
19. the MSS-Interworking layer checks if VPI/VCI is acceptable (available);
20. the MSS-Interworking layer decides if the end-user is:
    - an MEDIAN system end-user (is valid in this scenario);
    - not an MEDIAN system end-user;
21. at least one of the above mentioned checks is not okay (because this is scenario I_Setup_b);
22. if QoS check not okay then cause#49 = QoS unavailable;
23. if traffic parameter check not okay then cause#37 = resource unavailable, unspecified;
24. if traffic parameter check contains a non supported set of traffic parameters then cause#73 = unsupported combination of traffic parameters;
25. if VPI/VCI not available then cause#35 = requested VPI/VCI not available;
26. if non MEDIAN address then cause#89 = invalid endpoint reference;
27. MSS-Interworking layer changes the state of the call/connection from Call Present(U6) into Null(U0);
28. MSS-Interworking layer terminates call establishment by not delivering the SETUP message to the called UTE and by initiating sending of RELEASE COMPLETE message to the ATM switch;
29. before the MSS-Interworking is able to send the RELEASE COMPLETE message, it first has to extract additional information to generate the cause field. The RELEASE COMPLETE message consists of:
    - protocol discriminator ⇒ 0000 1001 (same as in the SETUP message);
    - call reference ⇒ same as in the SETUP message;
    - message type ⇒ 0101 1010 (RELEASE COMPLETE message);
    - message length ⇒ number of octets of the message contents, excluding the octets used for 'protocol discriminator', ',message type', and message length indication itself
    - cause ⇒ depending on the reason which caused the rejection of the call;
30. MSS-Interworking layer generates cause field subfields:
    - length of cause information contents;
    - location (describing the location of call rejection);
    - cause value (describing the reason of rejection);
    - diagnostic(s)(if any);
31. MSS-Interworking layer calculates the message length;
32. MSS-Interworking layer deduces statistical data (what ever this maybe) for static database (static list);
33. MSS-Interworking layer constructs the RELEASE COMPLETE message;
34. MSS-Interworking layer releases all the reservations made concerning this call/connection;
35. MSS-Interworking layer delivers the RELEASE COMPLETE message to the SAAL layer;
36. after the MSS-SAAL layer has done the necessary actions and segmented the signalling message into ATM-information containers, it delivers the ATM-information containers to the MSS-ATM layer;
37. the MSS-ATM layer adds the ATM-header to the ATM-information container;
38. the MSS-ATM layer delivers the bitstream to the MSS-PHY layer;

39. signalling (RELEASE COMPLETE) message is transported to the ATM switch using the opticalchannel;
40. ATM switch receives the signalling (RELEASE COMPLETE) message and knows that call establishment has been rejected;
41. call establishment is rejected by the MSS;

Appendix B

I_Setup_c

1. calling ATM end-user wants to establish a connection with a MPS and sends a SETUP message to the called MPS (through the ATM-switch, MSS, PRP and finally UTE);
2. ATM-switch accepts call and sends a CALL PROCEEDING message to the ATM end-user and a SETUP message (with VPI/VCI indicated in the Connection Identifier Information Element) to the called MPS (through the MSS ,PRP and UTE);

3. MSS-PHY layer receives information from ATM-switch, and delivers the bit stream to the MSS-ATM layer;
4. MSS-ATM layer delivers the signalling ATM cells without the header to the MSS-SAAL;
5. MSS-SAAL reassembles the signalling ATM cells and delivers the signalling message (in particular SETUP message) to the MSS-Interworking layer;
6. the MSS-Interworking layer receives the incoming signalling (SETUP) message and buffers the message;
7. the MSS-Interworking layer extracts the Protocol discriminator Information Element:
   - must be 0000 1001 (Q.2931);
8. the MSS-Interworking layer extracts the Call reference Information Element:
   - flag must be 0 (message sent from the side that originates the call reference);
   - call reference value (23 bit) saved in static table;
9. the MSS-Interworking layer extracts the Message type Information Element:
   - message type must be 0000 0101 (SETUP message);
   - change state of connection from Null(U0) into Call Present (U6) state in static table;
   - save external in 'originated' field in the static table;
10. the MSS-Interworking layer extracts the Message length Information Element:
    - extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself)
11. the MSS-Interworking layer extracts the ATM traffic descriptor Information Element:
    - deduce information about the forward/backward peak cell rate (CLP=0 and CLP=0+1) (if present);
    - deduce information about the forward/backward sustainable cell rate (CLP=0 and CLP=0+1) (if present);
    - deduce information about the forward/backward maximum burst size (CLP=0 and CLP=0+1) (if present);
    - save information in static data base;
12. the MSS-Interworking layer extracts the Broadband bearer capability Information Element:
    - bearer class (BCOB-A, BCOB-C, BCOB-X(most likely to be used));
    - traffic type (CBR, VBR);
    - timing requirements (no indication, end-to-end timing required, end-to-end timing not required;
    - susceptibility to clipping (yes, no);
    - user plane connection (point-to-point, point-to-multipoint);
    - save information in static data base;
13. the MSS-Interworking layer extracts the Called Party number Information element and deduces the ATM end system address or E.164 address;
14. the MSS-Interworking layer extracts the Connection Identifier Information Element:
    - extract VPI/VCI;
    - VPI/VCI is maintained in the static table;
15. the MSS-Interworking layer extracts the QoS information element, which can contain information concerning the:
    - cell error ratio;
    - severly-errored cell block ratio;
    - cell misinsertion ratio;
    - cell transfer delay;
    - mean cell transfer delay;
    - cell delay variation;
    - save information in static data base;
16. the MSS-Interworking layer checks if message is errorless (Yes assumed);

17. the MSS-Interworking layer checks if requested QoS can be provided, using the saved QoS information. There is additional information needed about the behaviour of the overall system for example BER, bursty characteristic of errors, propagation delay, transport delay of ATM cells in ATM layer, buffer capacity in MSS etc. It should be possible to deduce the necessary information from the information maintained in the static data base;
18. the MSS-Interworking layer checks if able to support combination of traffic parameters in both the directions (MSS⇒ MPS and MPS⇒ MSS). The MSS-Interworking layer maintains the information of the static list, therefore having the data to derive the free slots (and RVCI's) in the uplink and the downlink;
19. the MSS-Interworking layer checks if VPI/VCI is acceptable (available);
20. the MSS-Interworking layer decides if the end-user is:
    - an MEDIAN system end-user (is valid in this scenario);
    - not an MEDIAN system end-user (initiate call release);

    Check can be done because the MSS has an address table supplied by the address registration procedures of the management system containing, among other things, ATM end system address or E.164 address (which, within the MEDIAN system, is mapped on an internal shorter address (int_adress);
21. QoS check=okay AND traffic parameter check=okay AND VPI/VCI available=okay AND address check=okay (because this is scenario I_Setup_a);
22. MSS-Interworking layer translates the ATM end system address in an internal address;
23. MSS-Interworking layer saves internal address in static table;
24. MSS-Interworking layer unambiguous maps incoming VPI/VCI on a RVCI for user-data transport. This information is maintained in the static list (max. # of RVCI =32);
25. MSS-Interworking layer deduces time on which cells belonging to the selected RVCI, up- and downlink expire and retain these values in the static list (or just do selective queuing);
26. MSS-Interworking layer deduces statistical data (what ever this maybe) for static database;
    Parameters which could be of interest:
    - cell error ratio;
    - severly-errored cell block ratio;
    - cell misinsertion ratio;
    - cell transfer delay;
    - mean cell transfer delay;
    - cell delay variation;
    - total information flow uplink;
    - total information flow downlink;
    - uplink and downlink information flow per radio connection;
    - state a certain RVCI is in (null/call initiated/outgoing call proceeding/.../call abort/restart/restart request);
    - number of rejected calls on an average;
27. the MSS-Interworking layer starts timer T303;
28. IF no reaction to the SETUP message before expiry of timer T303 THEN initiate call/connection release;
29. no modification of the SETUP message, the Median Server Station seems to be transparent to the SETUP message in this scenario (only reading and checking of the SETUP message). The buffered Setup message is delivered to the MSS-SAAL layer;
30. after the MSS-SAAL layer has done the necessary actions and segments the signalling (SETUP) message into ATM-information containers, it delivers the ATM-information containers to the MSS-ATM layer;
31. the MSS-ATM layer adds the ATM-header to the ATM-information container;
32. the ATM cells and all the necessary information being of importance to the MSS-MAC are send to the MSS-MAC;
33. MSS-MAC delivers the bitstream to the MSS-PHY layer (SAAL, MAC and PHY take care of the 'reliable' transportation of the SETUP message to the PRP);

34. the 'signalling (SETUP) message' is transported to the PRP using the radiochannel;

35. the PRP-PHY layer delivers the constructed bitstream to the PRP-MAC layer;
36. the PRP-MAC delivers the broadcast cell to the PRP-Interworking layer;
37. the PRP-MAC delivers (if any) signalling cell(s) (VPI/VCI=0/5) to the Interworking layer;
38. the PRP-Interworking layer buffers the broadcast cell;
39. the PRP-Interworking layer buffers the signalling cell(s);
40. the PRP-Interworking layer deduces signalling broadcast cell from the broadcast cell;

41. the PRP-Interworking layer checks if signalling broadcast cell is nil;
42. if nil then delete broadcast cell and signalling broadcast cell, else continue (continue is assumed);
43. the PRP-Interworking layer deduces the RVCI and the internal address (or ATM end system address or E.164 address) from the signalling broadcast cell;
44. the PRP-Interworking layer checks if the internal address is the address of the PRP (MPS)
    - if unequal then do not proceed and delete the buffered ATM signalling cells (if any) and the broadcast cell;
    - if yes then proceed (yes is assumed);
45. the PRP-Interworking layer saves the RVCI in the static table;
46. the PRP-Interworking layer sets the in-use field in the static table to yes;
47. the PRP-Interworking layer starts timer concerning reaction of the UTE;
48. if timer expiry before answer of UTE then set the in-use field to no;
49. the PRP-Interworking layer feeds the buffered ATM (signalling) cells to the ATM-layer (where, among other things, the ATM cells are serialised into a bitstream);
    PRP-ATM layer delivers bitstream to the PRP-PHY layer;

50. 'SETUP message' is transported to the UTE layer, using the optical channel;
51. how the UTE exactly reacts is not of importance (because these functionality's are already implemented on the interface card mounted in the UTE);
52. the UTE decides to send a CALL PROCEEDING message (because this is scenario I_Setup_a) to the PRP (which finally has to arrive at the ATM switch) when the requested call establishment has been initiated and no more call establishment information will be accepted;

53. 'CALL PROCEEDING message' is transported to the PRP using the optical channel;

54. the PRP-PHY layer delivers the received bitstream to the PRP-ATM layer;
55. the PRP-ATM layer deserialises the bitstream into ATM cells;
56. the PRP-ATM layer delivers signalling ATM cell (if any) to the PRP-Interworking layer;
57. the PRP-Interworking layer buffers (in a receive buffer) the incoming signalling ATM cells (buffering is needed because the PRP-MAC ,maybe , first has to content for a free (signalling) slot);
58. the PRP-Interworking layer starts a timer to measure the time between two successive signalling cells;
59. on expiry of the timer, set the static table inuse value to false;
60. the PRP-Interworking layer checks if the static table inuse value is true or false;
61. if inuse is false then
    - ask the MAC layer to content for an uplink (signalling) timeslot and start timer for contention time;
    - on expiry of timer delete buffered signalling ATM cell
    - the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the PRP-Interworking layer stops the timer;
    - the PRP-Interworking layer changes the static table field inuse to true;
62. if inuse is true then the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
63. the PRP-Interworking layer stops the timer;
64. the PRP-Interworking layer checks if there is another signalling ATM cell in the de-serialiser (or better, waits until the PRP-ATM layer delivers a signalling cell before timer expiry);
65. if no signalling ATM cell available in de-serialiser then change static table field inuse to false (end);
66. if ATM cell available in de-serialiser then
    - the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the PRP-Interworking layer continues this procedure until there is no signalling ATM cell in the de-serialiser;
67. the PRP-MAC layer delivers the bitstream to the PRP-PHY layer;

68. the 'CALL PROCEEDING message' is transported to the MSS over the radio channel;

69. at the MSS the MSS-PHY layer delivers the constructed bitstream to the MSS-MAC layer;
70. the MSS-MAC layer delivers the signalling message to the MSS-ATM layer;
71. the MSS-ATM layer removes the ATM cell header and delivers the ATM information cell to the MSS-SAAL;
72. the MSS-SAAL reassembles (among other things) the signalling message. The MSS-SAAL delivers the signalling message to the MSS-Interworking layer;

73. the MSS-Interworking layer receives the incoming signalling (CALL PROCEEDING) message and buffers the message;
74. the MSS-Interworking layer extracts the Protocol discriminator Information Element:
    - must be 000010001 (Q.2931)
75. the MSS-Interworking layer extracts the Call reference Information Element:
    - flag must be 1 (message sent to side that originates the call reference (calling party));
    - deduce call reference value
76. the deduced call reference value is also maintained in static table, and uniquely identifies the call the signalling message refers to;
77. the MSS-Interworking layer extracts the Message type Information Element:
    - message type must be 0000 0010 (CALL PROCEEDING message);
    - change state of connection from Call Present (U6) into Incoming Call Proceeding (U3) state in static table;
    - stops T303;
    - start T310;
78. IF no reaction to the CALL PROCEEDING message before expiry of timer T310 THEN initiate call/connection release;
79. the MSS-Interworking layer extracts the Message length Information Element;
80. extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself);
81. the MSS-Interworking layer checks if message is errorless (Yes assumed);
82. MSS-Interworking layer deduces statistical data (what ever this maybe) for static database;
83. the MSS-Interworking layer delivers the received (buffered) message to the MSS-SAAL;
84. the MSS-SAAL delivers the segmented CALL PROCEEDING message to the MSS-ATM layer (synchronised);
85. the MSS-ATM layer delivers the bitstream to the MSS-PHY layer;
86. 'signalling (CALL PROCEEDING) message' is delivered to the ATM-switch (actions concerning CALL PROCEEDING message within the MEDIAN system are finished);

87. the MPS decides to send a RELEASE COMPLETE message to the ATM-switch;
88. 'RELEASE COMPLETE message' is transported to the PRP using the optical channel;

89. the PRP-PHY layer delivers the received bitstream to the PRP-ATM layer;
90. the PRP-ATM layer deserialises the bitstream into ATM cells;
91. the PRP-ATM layer delivers signalling ATM cell (if any) to the PRP-Interworking layer;
92. the PRP-Interworking layer buffers (in a receive buffer) the incoming signalling ATM cells (buffering is needed because the PRP-MAC ,maybe , first has to content for a free (signalling) slot);
93. the PRP-Interworking layer starts a timer to measure the time between two successive signalling cells;
94. on expiry of the timer, set the static table inuse value to false;
95. the PRP-Interworking layer checks if the static table inuse value is true or false;
96. if inuse is false then
    - ask the MAC layer to content for an uplink (signalling) timeslot and start timer for contention time;
    - on expiry of timer delete buffered signalling ATM cell
    - the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the PRP-Interworking layer stops the timer;
    - the PRP-Interworking layer changes the static table field inuse to true;
97. if inuse is true then the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
98. the PRP-Interworking layer stops the timer;
99. the PRP-Interworking layer checks if there is another signalling ATM cell in the de-serialiser (or better, waits until the PRP-ATM layer delivers a signalling cell before timer expiry);
100. if no signalling ATM cell available in de-serialiser then change static table field inuse to false (end);
101. if ATM cell available in de-serialiser then
    - the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the PRP-Interworking layer continues this procedure until there is no signalling ATM cell in the de-serialiser;
102. the PRP-MAC layer delivers the bitstream to the PRP-PHY layer;

103. at the MSS the MSS-PHY layer delivers the constructed bitstream to the MSS-MAC layer;

104.the MSS-MAC layer delivers the 'signalling (RELEASE COMPLETE) message' to the MSS-ATM layer;
105.the MSS-ATM layer removes the ATM cell header and delivers the signalling ATM information cells to the MSS-SAAL;
106.the MSS-SAAL reassembles (among other things) the received signalling ATM information cell containers. The MSS-SAAL delivers the signalling (RELEASE COMPLETE) message to the MSS-Interworking layer;
107.the MSS-Interworking layer receives the incoming signalling (RELEASE COMPLETE) message and buffers the message;
108.the MSS-Interworking layer extracts the Protocol discriminator Information Element:
- must be 000010001 (Q.2931)

109.the MSS-Interworking layer extracts the Call reference Information Element:
- flag must be 1 (message sent to side that originates the call reference (calling party));

110.the deduced call reference value has been saved in the static table, and uniquely identifies the call the signalling message refers to;
111.the MSS-Interworking layer extracts the Message type Information Element:
- message type must be 0101 1010 (RELEASE COMPLETE message);
- the MSS-Interworking layer stop T310;
- change state of connection from Incoming Call Proceeding (U9) into Null(U0) state in static table;

112.the MSS-Interworking layer extracts the Message length Information Element:
113.extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself);
114.the MSS-Interworking layer checks if message is errorless (Yes assumed);
115.deduce information needed to construct the static data base (optional);
116.the MSS-Interworking layer locally releases all the reservations made concerning the connection (connection is identified by the call reference Information element);
117.MSS-Interworking layer does not change the RELEASE COMPLETE message (only reading), and the RELEASE COMPLETE message is delivered to the MSS-SAAL;
118.the MSS-SAAL delivers the segmented signalling (RELEASE COMPLETE) message to the MSS-ATM layer;
119.the MSS-ATM layer delivers the bitstream to the MSS-PHY layer
120.ATM switch receives the 'RELEASE COMPLETE message' and decides to send a RELEASE COMPLETE message to the ATM end-user;
121.call establishment is rejected;

II_Setup_a

1. calling MPS wants to establish a connection with a ATM end-user;
2. calling MPS initiates call establishment by sending SETUP message to the ATM-switch;
3. calling MPS shall not include the Connection Identifier information element;

4. UTE transmits a 'SETUP message' to the PRP using the opticalchannel;

5. the PRP-PHY layer delivers the received bitstream to the PRP-ATM layer;
6. the PRP-ATM layer deserialises the bitstream into ATM cells;
7. the PRP-ATM layer delivers signalling ATM cell (if any) to the PRP-Interworking layer;
8. the PRP-Interworking layer buffers (in a receive buffer) the incoming signalling ATM cells (buffering is needed because the PRP-MAC ,maybe , first has to content for a free (signalling) slot);
9. the PRP-Interworking layer starts a timer to measure the time between two successive signalling cells;
10. on expiry of the timer, set the static table inuse value to false;
11. the PRP-Interworking layer checks if the static table inuse value is true or false;
12. if inuse is false then
    - ask the MAC layer to content for an uplink (signalling) timeslot and start timer for contention time;
    - on expiry of timer delete buffered signalling ATM cell
    - the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the PRP-Interworking layer stops the timer;
    - the PRP-Interworking layer changes the static table field inuse to true;
13. if inuse is true then the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
14. the PRP-Interworking layer stops the timer;
15. the PRP-Interworking layer checks if there is another signalling ATM cell in the de-serialiser (or better, waits until the PRP-ATM layer delivers a signalling cell before timer expiry);
16. if no signalling ATM cell available in de-serialiser then change static table field inuse to false (end);
17. if ATM cell available in de-serialiser then
    - the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the PRP-Interworking layer continues this procedure until there is no signalling ATM cell in the de-serialiser;
18. the PRP-MAC layer delivers the bitstream to the PRP-PHY layer;

19. the 'SETUP message' is transported to the MSS over the radio channel;.

20. at the MSS, the MSS-PHY layer delivers the constructed bitstream to the MSS-MAC layer;
21. the MSS-MAC layer delivers the 'signalling (SETUP) message to the MSS-ATM layer;
22. the MSS-ATM layer removes the ATM cell header (among other things) and delivers the ATM information cell to the MSS-SAAL;
23. the MSS-SAAL reassembles (among other things) the signalling message. The MSS-SAAL delivers the signalling message to the MSS-Interworking layer;
24. the MSS-Interworking layer receives the incoming signalling (SETUP) message and buffers the message;
25. the MSS-Interworking layer extracts the Protocol discriminator Information Element:
    - must be 0000 1001 (Q.2931)
26. the MSS-Interworking layer extracts the Call reference Information Element:
    - flag must be 0 (message sent from the side that originates the call reference);
    - call reference value (23 bit) saved in static table;
27. the MSS-Interworking layer extracts the Message type Information Element:
    - message type must be 0000 0101 (SETUP message);
    - save internal in 'originated' field in the static table;
    - change state of connection from Null(U0) into Call Initiated(U1) state in static table;
28. the MSS-Interworking layer extracts the Message length Information Element:
    - extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself);

29. the MSS-Interworking layer extracts the ATM traffic descriptor Information Element:
    - deduce information about the forward/backward peak cell rate (CLP=0 and CLP=0+1) (if present);
    - deduce information about the forward/backward sustainable cell rate (CLP=0 and CLP=0+1) (if present);
    - deduce information about the forward/backward maximum burst size (CLP=0 and CLP=0+1) (if present);
    - save information in static data base;
30. the MSS-Interworking layer extracts the Broadband bearer capability Information Element:
    - bearer class (BCOB-A, BCOB-C, BCOB-X(most likely to be used));
    - traffic type (CBR, VBR);
    - timing requirements (no indication, end-to-end timing required, end-to-end timing not required;
    - susceptibility to clipping (yes,no);
    - user plane connection (point-to-point, point-to-multipoint);
    - save information in static data base;
31. the MSS-Interworking layer extracts the Called party Number Information Element and deduces the ATM end system address;
32. the MSS-Interworking layer extracts the QoS Information element. The QoS information element can contain information concerning the:
    - cell error ratio;
    - severly-errored cell block ratio;
    - cell misinsertion ratio;
    - cell transfer delay;
    - mean cell transfer delay;
    - cell delay variation;
    - save information in static data base;
33. the MSS-Interworking layer checks if message is errorless (Yes assumed);
34. the MSS-Interworking layer decides if the end-user is:
    - an MEDIAN system user;
    - not an MEDIAN system user;
35. the MSS-Interworking layer checks if requested QoS can be provided, using the extracted QoS information. There is additional information needed about the behaviour of the overall system for example BER, bursty characteristic of errors, propagation delay, transport delay of ATM cells in ATM layer, buffer capacity in Median Server Station etc. It should be possible to deduce the necessary information from the information maintained in the static data base;
36. the MSS-Interworking layer checks if able to support combination of traffic parameters in the directions (MSS⇒ MPS and MPS⇒ MSS):
    - peak cell rate when CLP=0;
    - peak cell rate when CLP=0+1;
    - tagging (if requested);
    - sustainable cell rate CLP=0;
    - sustainable cell rate CLP=0+1;
    - maximum burst size CLP=0;
    - maximum burst size CLP=0+1.

    The MSS-Interworking layer maintains the information of the static list, therefore having the data to derive the free slots in the uplink and the downlink;
37. because the MSS-interworking layer rejects the call establishment (because this is scenario II_setup_a), at least one of the previous checks is not okay;
38. if QoS unavailable then cause#49 = QoS unavailable;
39. if traffic parameters not okay then cause#51 = user cell rate unavailable;
40. if traffic parameters contains a non supported set of traffic parameters then cause#73 = unsupported combination of traffic parameters;
41. MSS-interworking layer terminates call establishment by not sending the SETUP message to the called UTE and by initiating sending of RELEASE COMPLETE message to the calling UTE;
42. MSS-interworking layer changes the state from Call Initiated (U1) to Null(U0) in the static table;

43. before the MSS-Interworking is able to send the RELEASE COMPLETE message, it has to extract additional information to generate the cause field.
    The RELEASE COMPLETE message consists of:
    - protocol discriminator ⇒ 0000 1001 (same as in the SETUP message);
    - call reference ⇒ same as in the SETUP message;
    - message type ⇒ 0101 1010 (RELEASE COMPLETE message);
    - message length ⇒ number of octets of the message contents, excluding the octets used for 'protocol discriminator', ',message type', and message length indication itself
    - cause ⇒ depending on the reason which caused the rejection of the call;
44. MSS-Interworking layer generates cause field subfields:
    - length of cause information contents;
    - location (describing the location of call rejection);
    - cause value (describing the reason of rejection);
    - diagnostic(s)(if any);
45. MSS-Interworking layer calculates the message length;
46. MSS-Interworking layer constructs the RELEASE COMPLETE message;
47. MSS-Interworking layer deduces statistical data (what ever this maybe) for static database
48. MSS-Interworking layer releases the reservations made for this connection establishment;
49. MSS-Interworking layer delivers the RELEASE COMPLETE message to the SAAL layer;
50. after the MSS-SAAL layer has done the necessary actions and segmented the signalling (RELEASE COMPLETE) message into ATM-information containers, it delivers the ATM-information containers to the MSS-ATM layer;
51. the MSS-ATM layer adds the ATM-header to the ATM-information container;
52. the ATM cells and all the necessary information being of importance to the MSS-MAC are send to the MSS-MAC;
53. MSS-MAC delivers the bitstream to the MSS-PHY layer (MAC and PHY take care of the 'reliable' transportation of the RELEASE message to the MPS);

54. 'RELEASE COMPLETE 'message is transported to the MPS using the radiochannel;

55. the PRP-PHY layer delivers the constructed bitstream to the PRP-MAC layer;
56. the PRP-MAC delivers the broadcast cell to the PRP-Interworking layer;
57. the PRP-MAC delivers (if any) signalling cell(s) (VPI/VCI=0/5) to the Interworking layer;
58. the PRP-Interworking layer buffers the broadcast cell;
59. the PRP-Interworking layer buffers the signalling cell(s);
60. the PRP-Interworking layer deduces signalling broadcast cell from the broadcast cell;
61. the PRP-Interworking layer checks if signalling broadcast cell is nil;
62. if nil then delete broadcast cell and signalling broadcast cell, else continue (continue is assumed);
63. the PRP-Interworking layer deduces the RVCI and the internal address (or ATM end system address or E.164 address) from the signalling broadcast cell;
64. the PRP-Interworking layer checks if the internal address is the address of the PRP (MPS)
    - if unequal then do not proceed and delete the buffered ATM signalling cells (if any) and the broadcast cell;
    - if yes then proceed (yes is assumed);
65. the PRP-Interworking layer saves the RVCI in the static table;
66. the PRP-Interworking layer sets the in-use field in the static table to yes;
67. the PRP-Interworking layer starts timer concerning reaction of the UTE;
68. if timer expiry before answer of UTE then set the in-use field to no;
69. the PRP-Interworking layer feeds the buffered ATM (signalling) cells to the ATM-layer (where, among other things, the ATM cells are serialised into a bitstream);
70. PRP-ATM layer delivers bitstream to the PRP-PHY layer;

71. 'RELEASE COMPLETE message' is transported to the UTE, using the optical channel;
72. calling MPS (UTE) knows that the call establishment has been rejected;

II_Setup_b

1. calling UTE wants to establish a connection with a ATM end-user;
2. calling UTE initiates call establishment by sending SETUP message to the ATM switch (through the calling PRP and the MSS);

3. UTE transmits a 'SETUP message' to the PRP using the opticalchannel;
4. calling MPS shall not include the Connection Identifier information element;

5. the PRP-PHY layer delivers the received bitstream to the PRP-ATM layer;
6. the PRP-ATM layer deserialises the bitstream into ATM cells;
7. the PRP-ATM layer delivers signalling ATM cell (if any) to the PRP-Interworking layer;
8. the PRP-Interworking layer buffers (in a receive buffer) the incoming signalling ATM cells (buffering is needed because the PRP-MAC ,maybe , first has to content for a free (signalling) slot);
9. the PRP-Interworking layer starts a timer to measure the time between two successive signalling cells;
10. on expiry of the timer, set the static table inuse value to false;
11. the PRP-Interworking layer checks if the static table inuse value is true or false;
12. if inuse is false then
    - ask the MAC layer to content for an uplink (signalling) timeslot and start timer for contention time;
    - on expiry of timer delete buffered signalling ATM cell
    - the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the PRP-Interworking layer stops the timer;
    - the PRP-Interworking layer changes the static table field inuse to true;
13. if inuse is true then the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
14. the PRP-Interworking layer stops the timer;
15. the PRP-Interworking layer checks if there is another signalling ATM cell in the de-serialiser (or better, waits until the PRP-ATM layer delivers a signalling cell before timer expiry);
16. if no signalling ATM cell available in de-serialiser then change static table field inuse to false (end);
17. if ATM cell available in de-serialiser then
    - the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the PRP-Interworking layer continues this procedure until there is no signalling ATM cell in the de-serialiser;
18. the PRP-MAC layer delivers the bitstream to the PRP-PHY layer;

19. the 'SETUP message' is transported to the MSS over the radio channel;.

20. at the MSS, the MSS-PHY layer delivers the constructed bitstream to the MSS-MAC layer;
21. the MSS-MAC layer delivers the 'signalling (SETUP) message to the MSS-ATM layer;
22. the MSS-ATM layer removes the ATM cell header (among other things) and delivers the ATM information cell to the MSS-SAAL;
23. the MSS-SAAL reassembles (among other things) the signalling message. The MSS-SAAL delivers the signalling message to the MSS-Interworking layer;
24. the MSS-Interworking layer receives the incoming signalling (SETUP) message and buffers the message;
25. the MSS-Interworking layer extracts the Protocol discriminator Information Element:
    - must be 0000 1001 (Q.2931)
26. the MSS-Interworking layer extracts the Call reference Information Element:
    - flag must be 0(message sent from the side that originates the call reference);
    - call reference value (23 bit) saved in static table;
27. the MSS-Interworking layer extracts the Message type Information Element:
    - message type must be 0000 0101 (SETUP message);
    - save internal in 'originated' field in the static table;
    - change state of connection from Null(U0) into Call Initiated(U1) state in static table;
28. the MSS-Interworking layer extracts the Message length Information Element:
    - extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself);

29. the MSS-Interworking layer extracts the ATM traffic descriptor Information Element:
    - deduce information about the forward/backward peak cell rate (CLP=0 and CLP=0+1) (if present);
    - deduce information about the forward/backward sustainable cell rate (CLP=0 and CLP=0+1) (if present);
    - deduce information about the forward/backward maximum burst size (CLP=0 and CLP=0+1) (if present);
    - save information in static data base;
30. the MSS-Interworking layer extracts the Broadband bearer capability Information Element:
    - bearer class (BCOB-A, BCOB-C, BCOB-X(most likely to be used));
    - traffic type (CBR, VBR);
    - timing requirements (no indication, end-to-end timing required, end-to-end timing not required;
    - susceptibility to clipping (yes,no);
    - user plane connection (point-to-point, point-to-multipoint);
    - save information in static data base;
31. the MSS-Interworking layer extracts the Called party Number Information Element and deduces the ATM end system address;
32. the MSS-Interworking layer extracts the QoS Information element. The QoS information element can contain information concerning the:
    - cell error ratio;
    - severly-errored cell block ratio;
    - cell misinsertion ratio;
    - cell transfer delay;
    - mean cell transfer delay;
    - cell delay variation;
    - save information in static data base;
33. the MSS-Interworking layer checks if message is errorless (Yes assumed);
34. the MSS-Interworking layer decides if the end-user is:
    - an MEDIAN system user;
    - not an MEDIAN system user;
35. the MSS-Interworking layer checks if requested QoS can be provided, using the extracted QoS information. There is additional information needed about the behaviour of the overall system for example BER, bursty characteristic of errors, propagation delay, transport delay of ATM cells in ATM layer, buffer capacity in Median Server Station etc. It should be possible to deduce the necessary information from the information maintained in the static data base;
36. the MSS-Interworking layer checks if able to support combination of traffic parameters in the directions (MSS⇒ MPS and MPS⇒ MSS):
    - peak cell rate when CLP=0;
    - peak cell rate when CLP=0+1;
    - tagging (if requested);
    - sustainable cell rate CLP=0;
    - sustainable cell rate CLP=0+1;
    - maximum burst size CLP=0;
    - maximum burst size CLP=0+1.

    The MSS-Interworking layer maintains the information of the·static list, therefore having the data to derive the free slots in the uplink and the downlink;
37. QoS check=okay AND traffic parameter check=okay AND non MEDIAN user(because this is scenario II_Setup_b);
38. MSS-Interworking layer maps ATM end user address on an internal address;
39. MSS-Interworking layer: deduce time on which cells, up- and downlink, expire and retain these values in the static list (or just do selective queuing);
40. MSS-Interworking layer chooses and saves an unused RVCI for this call/connection;
41. MSS-Interworking layer starts timer T303;
42. IF no reaction to the SETUP message before expiry of timer T303 THEN initiate call/connection release;
43. the signalling (SETUP) message is delivered it to the MSS-SAAL;
44. the MSS-SAAL delivers the segmented signalling (SETUP) message to the MSS-ATM layer;
45. the MSS-ATM layer delivers the bitstream to the MSS-PHY layer;
46. the 'SETUP message' is send to the ATM-switch over the optical channel;

47. the ATM-switch decides to send a CALL PROCEEDING message to the calling MPS;
48. CALL PROCEEDING message is transported to the calling MPS, including the VPI/VCI value indicated in the Connection Identifier information element;

49. MSS-PHY layer delivers received bit stream to ATM layer;
50. MSS-ATM layer delivers the signalling ATM cells without the header to the MSS-SAAL;
51. MSS-SAAL reassembles the signalling ATM cells and delivers the signalling message (CALL PROCEEDING message) to the MSS-Interworking layer;
52. the MSS-Interworking layer receives the incoming signalling (CALL PROCEEDING) message and buffers the message;
53. the MSS-Interworking layer extracts the Protocol discriminator Information Element:
    - must be 000010001 (Q.2931)
54. the MSS-Interworking layer extracts the Call reference Information Element:
    - flag must be 1 (message sent to side that originates the call reference (calling party));
55. the deduced call reference value is also saved in static table, and uniquely identifies the call the signalling message refers to;
56. the MSS-Interworking layer extracts the Message type Information Element:
    - message type must be 0000 0010 (CALL PROCEEDING message);
    - change state of connection from Call Initiated(U1) into Outgoing Call Proceeding(U3) state in static table;
    - stops timer T303;
57. IF no reaction to the CALL PROCEEDING message before expiry of timer T310 THEN initiate call/connection release;
58. the MSS-Interworking layer extracts the Message length Information Element:
59. the MSS-Interworking layer extracts the length of message (excluding protocol discriminator, call reference, message type, message length indication itself);
60. because this CALL PROCEEDING message is the first message in response to the SETUP message, the Connection Identifier Information Element is mandatory;
61. the MSS-Interworking layer extracts the Connection Identifier Information Element:
    - extract the VPI value;
    - extract the VCI value;
62. save retrieved VPI/VCI value in the static table (mapping on the RVCI is established, because RVCI has been already selected);
63. the MSS-Interworking layer checks if message is errorless (Yes assumed);
64. no modification of the CALL PROCEEDING message, the Median Server Station seems to be transparent to the CALL PROCEEDING message in this scenario;
65. the MSS-Interworking layer starts timer T310;
66. IF no reaction to the CALL PROCEEDING message before expiry of timer T310 THEN initiate call/connection release;
67. CALL PROCEEDING message is delivered to the MSS-SAAL layer (it is possible to directly sent the buffered CALL PROCEEDING message (in received buffer) to the SAAL because no modification has taken place);
68. after the MSS-SAAL layer has done the necessary actions and segmented the CALL PROCEEDING message into ATM-information containers. The flow of ATM information containers is delivered to the MSS-ATM layer;
69. the MSS-ATM layer adds the ATM-header to the ATM-information container resulting in the same ATM cell corresponding to the arrived ATM-cell at the beginning (meaning that the CALL PROCEEDING message is transparently transported through the MSS-Interworking layer);
70. the ATM cells and all the necessary information being of importance to the MSS-MAC are send to the MSS-MAC.;
71. MSS-MAC delivers the bitstream to the MSS-PHY layer (MAC and PHY take care of the 'reliable' transportation of the 'CALL PROCEEDING message' to the PRP);

72. 'CALL PROCEEDING message' is transported to the MPS using the radiochannel;

73. the PRP-PHY layer delivers the constructed bitstream to the PRP-MAC layer;
74. the PRP-MAC delivers the broadcast cell to the PRP-Interworking layer;
75. the PRP-MAC delivers (if any) signalling cell(s) (VPI/VCI=0/5) to the Interworking layer;

76. the PRP-Interworking layer buffers the broadcast cell;
77. the PRP-Interworking layer buffers the signalling cell(s);
78. the PRP-Interworking layer deduces signalling broadcast cell from the broadcast cell;
79. the PRP-Interworking layer checks if signalling broadcast cell is nil;
80. if nil then delete broadcast cell and signalling broadcast cell, else continue (continue is assumed);
81. the PRP-Interworking layer deduces the RVCI and the internal address (or ATM end system address or E.164 address) from the signalling broadcast cell;
82. the PRP-Interworking layer checks if the internal address is the address of the PRP (MPS)
    - if unequal then do not proceed and delete the buffered ATM signalling cells (if any) and the broadcast cell;
    - if yes then proceed (yes is assumed);
83. the PRP-Interworking layer saves the RVCI in the static table;
84. the PRP-Interworking layer sets the in-use field in the static table to yes;
85. the PRP-Interworking layer starts timer concerning reaction of the UTE;
86. if timer expiry before answer of UTE then set the in-use field to no;
87. the PRP-Interworking layer feeds the buffered ATM (signalling) cells to the ATM-layer (where, among other things, the ATM cells are serialised into a bitstream);
88. PRP-ATM layer delivers bitstream to the PRP-PHY layer;

89. CALL PROCEEDING message is transported to the calling UTE, using the optical channel;
90. the 'CALL PROCEEDING message' arrives at the destination, calling MPS (UTE);

91. calling UTE knows that all the information needed by the ATM switch has arrived;
92. the ATM-switch decides to reject the call establishment and sends a RELEASE COMPLETE message with the appropriate cause field to the calling MPS;

93. MSS-PHY layer receives the 'RELEASE COMPLETE message' from the ATM switch;
94. MSS-PHY layer delivers the 'RELEASE COMPLETE message' to the MSS-ATM layer;
95. MSS-ATM layer delivers the signalling ATM cells without the header to the MSS-SAAL;
96. MSS-SAAL reassembles the signalling ATM cells and delivers the RELEASE COMPLETE to the MSS-Interworking layer;
97. the MSS-Interworking layer receives the incoming signalling (RELEASE COMPLETE) message and buffers the message;
98. the MSS-Interworking layer extracts the Protocol discriminator Information Element:
    - must be 00001001 (Q.2931)
99. the MSS-Interworking layer extracts the Call reference Information Element:
    - flag must be 1 (message sent to side that originates the call reference (calling party));
100. the deduced call reference value has been saved in the static table, and uniquely identifies the call the signalling message refers to;
101. the MSS-Interworking layer extracts the Message type Information Element:
    - message type must be 0101 1010 (RELEASE COMPLETE message);
    - change state of connection from Call Proceeding(U3) into Null(U0) state in static table;
    - stop timer T310;
102. the MSS-Interworking layer extracts the Message length Information Element:
103. extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself);
104. the MSS-Interworking layer checks if message is errorless (Yes assumed);
105. deduce information needed to construct the static data base (optional);
106. the MSS-Interworking layer locally releases all the reservations made concerning the connection (connection is identified by the call reference Information element);
107. MSS-Interworking layer does not change the RELEASE COMPLETE message (only reading), and the RELEASE COMPLETE message is delivered to the MSS-SAAL;
108. after the MSS-SAAL layer has done the necessary actions and segmented the RELEASE COMPLETE message into ATM-information containers. The flow of ATM information containers is delivered to the MSS-ATM layer;
109. the MSS-ATM layer adds the ATM-header to the ATM-information container resulting in the same ATM cell corresponding to the arrived ATM-cell at the beginning (meaning that the RELEASE COMPLETE message is transparently transported through the MSS-Interworking layer);

110. the ATM cells and all the necessary information being of importance to the MSS-MAC are send to the MSS-MAC;
111. MSS-MAC delivers the bitstream to the MSS-PHY layer;

112. 'RELEASE COMPLETE message' is transported to the PRP using the radiochannel;

113. the PRP-PHY layer delivers the constructed bitstream to the PRP-MAC layer;
114. the PRP-MAC delivers the broadcast cell to the PRP-Interworking layer;
115. the PRP-MAC delivers (if any) signalling cell(s) (VPI/VCI=0/5) to the Interworking layer;
116. the PRP-Interworking layer buffers the broadcast cell;
117. the PRP-Interworking layer buffers the signalling cell(s);
118. the PRP-Interworking layer deduces signalling broadcast cell from the broadcast cell;
119. the PRP-Interworking layer checks if signalling broadcast cell is nil;
120. if nil then delete broadcast cell and signalling broadcast cell, else continue (continue is assumed);
121. the PRP-Interworking layer deduces the RVCI and the internal address (or ATM end system address or E.164 address) from the signalling broadcast cell;
122. the PRP-Interworking layer checks if the internal address is the address of the PRP (MPS)
    - if unequal then do not proceed and delete the buffered ATM signalling cells (if any) and the broadcast cell;
    - if yes then proceed (yes is assumed);
123. the PRP-Interworking layer saves the RVCI in the static table;
124. the PRP-Interworking layer sets the in-use field in the static table to yes;
125. the PRP-Interworking layer starts timer concerning reaction of the UTE;
126. if timer expiry before answer of UTE then set the in-use field to no;
127. the PRP-Interworking layer feeds the buffered ATM (signalling) cells to the ATM-layer (where, among other things, the ATM cells are serialised into a bitstream);
128. PRP-ATM layer delivers bitstream to the PRP-PHY layer;

129. 'RELEASE COMPLETE message' is transported to the calling MPS (UTE);
130. call establishment initiated by the calling MPS is rejected;

II_Setup_c

1. calling UTE wants to establish a connection with a ATM end-user;
2. calling UTE initiates call establishment by sending SETUP message to the ATM switch (through the calling PRP and the MSS);

3. UTE transmits a 'SETUP message' to the PRP using the opticalchannel;
4. calling MPS shall not include the Connection Identifier information element;
5.
6. the PRP-PHY layer delivers the received bitstream to the PRP-ATM layer;
7. the PRP-ATM layer deserialises the bitstream into ATM cells;
8. the PRP-ATM layer delivers signalling ATM cell (if any) to the PRP-Interworking layer;
9. the PRP-Interworking layer buffers (in a receive buffer) the incoming signalling ATM cells (buffering is needed because the PRP-MAC ,maybe , first has to content for a free (signalling) slot);
10. the PRP-Interworking layer starts a timer to measure the time between two successive signalling cells;
11. on expiry of the timer, set the static table inuse value to false;
12. the PRP-Interworking layer checks if the static table inuse value is true or false;
13. if inuse is false then
    - ask the MAC layer to content for an uplink (signalling) timeslot and start timer for contention time;
    - on expiry of timer delete buffered signalling ATM cell
    - the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the PRP-Interworking layer stops the timer;
    - the PRP-Interworking layer changes the static table field inuse to true;
14. if inuse is true then the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
15. the PRP-Interworking layer stops the timer;
16. the PRP-Interworking layer checks if there is another signalling ATM cell in the de-serialiser (or better, waits until the PRP-ATM layer delivers a signalling cell before timer expiry);
17. if no signalling ATM cell available in de-serialiser then change static table field inuse to false (end);
18. if ATM cell available in de-serialiser then
    - the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the PRP-Interworking layer continues this procedure until there is no signalling ATM cell in the de-serialiser;
19. the PRP-MAC layer delivers the bitstream to the PRP-PHY layer;

20. the 'SETUP message' is transported to the MSS over the radio channel;.

21. at the MSS, the MSS-PHY layer delivers the constructed bitstream to the MSS-MAC layer;
22. the MSS-MAC layer delivers the 'signalling (SETUP) message to the MSS-ATM layer;
23. the MSS-ATM layer removes the ATM cell header (among other things) and delivers the ATM information cell to the MSS-SAAL;
24. the MSS-SAAL reassembles (among other things) the signalling message. The MSS-SAAL delivers the signalling message to the MSS-Interworking layer;
25. the MSS-Interworking layer receives the incoming signalling (SETUP) message and buffers the message;
26. the MSS-Interworking layer extracts the Protocol discriminator Information Element:
    - must be 0000 1001 (Q.2931)
27. the MSS-Interworking layer extracts the Call reference Information Element:
    - flag must be 0 (message sent from the side that originates the call reference);
    - call reference value (23 bit) saved in static table;
28. the MSS-Interworking layer extracts the Message type Information Element:
    - message type must be 0000 0101 (SETUP message);
    - change state of connection from Null(U0) into Call Initiated(U1) state in static table;
    - save internal in 'originated' field in the static table;
29. the MSS-Interworking layer extracts the Message length Information Element:
    - extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself);

30. the MSS-Interworking layer extracts the ATM traffic descriptor Information Element:
    - deduce information about the forward/backward peak cell rate (CLP=0 and CLP=0+1) (if present);
    - deduce information about the forward/backward sustainable cell rate (CLP=0 and CLP=0+1) (if present);
    - deduce information about the forward/backward maximum burst size (CLP=0 and CLP=0+1) (if present);
    - save information in the static data base;
31. the MSS-Interworking layer extracts the Broadband bearer capability Information Element:
    - bearer class (BCOB-A, BCOB-C, BCOB-X(most likely to be used));
    - traffic type (CBR, VBR);
    - timing requirements (no indication, end-to-end timing required, end-to-end timing not required;
    - susceptibility to clipping (yes,no);
    - user plane connection (point-to-point, point-to-multipoint);
    - save information in the static data base;
32. the MSS-Interworking layer extracts the Called party Number Information Element and deduces the ATM end system address;
33. the MSS-Interworking layer extracts the QoS Information element. The QoS information element can contain information concerning the:
    - cell error ratio;
    - severly-errored cell block ratio;
    - cell misinsertion ratio;
    - cell transfer delay;
    - mean cell transfer delay;
    - cell delay variation;
    - save information in the static data base;
34. the MSS-Interworking layer decides if the end-user is:
    - an MEDIAN system user;
    - not an MEDIAN system user;
35. the MSS-Interworking layer checks if message is errorless (Yes assumed);
36. the MSS-Interworking layer checks if requested QoS can be provided, using the extracted QoS information. There is additional information needed about the behaviour of the overall system for example BER, bursty characteristic of errors, propagation delay, transport delay of ATM cells in ATM layer, buffer capacity in Median Server Station etc. It should be possible to deduce the necessary information from the information maintained in the static data base;
37. the MSS-Interworking layer checks if able to support combination of traffic parameters in the directions (MSS⇒ MPS and MPS⇒ MSS):
    - peak cell rate when CLP=0;
    - peak cell rate when CLP=0+1;
    - tagging (if requested);
    - sustainable cell rate CLP=0;
    - sustainable cell rate CLP=0+1;
    - maximum burst size CLP=0;
    - maximum burst size CLP=0+1.

    The MSS-Interworking layer maintains the information of the static list, therefore having the data to derive the free slots in the uplink and the downlink;
38. QoS check=okay AND traffic parameter check=okay AND non MEDIAN user(because this is scenario II_Setup_b);
39. MSS-Interworking layer: deduce time on which cells, up- and downlink, expire and retain this value in the static list (or just do selective queuing);
40. MSS-Interworking layer chooses and saves an unused RVCI for this call/connection;
41. MSS-Interworking layer starts timer T303;
42. IF no reaction to the SETUP message before expiry of timer T303 THEN initiate call/connection release;
43. the signalling (SETUP) message is delivered it to the MSS-SAAL;
44. the MSS-SAAL delivers the segmented signalling (SETUP) message to the MSS-ATM layer;
45. the MSS-ATM layer delivers the bitstream to the MSS-PHY layer;
46. the 'SETUP message' is send to the ATM switch over the optical channel;
47. the ATM switch decides to send a CALL PROCEEDING message to the calling UTE;

48. CALL PROCEEDING message is transported to the calling UTE (through the MSS and the PRP), including the VPI/VCI value indicated in the Connection Identifier information element;
49. MSS-PHY layer delivers received bit stream to ATM layer;
50. MSS-ATM layer delivers the signalling ATM cells without the header to the MSS-SAAL;
51. MSS-SAAL reassembles the signalling ATM cells and delivers the signalling message (CALL PROCEEDING message) to the MSS-Interworking layer;
52. the MSS-Interworking layer receives the incoming signalling (CALL PROCEEDING) message and buffers the message;
53. the MSS-Interworking layer extracts the Protocol discriminator Information Element:
    - must be 000010001 (Q.2931)
54. the MSS-Interworking layer extracts the Call reference Information Element:
    - flag must be 1 (message sent to side that originates the call reference (calling party));
55. the deduced call reference value is also saved in static table, and uniquely identifies the call the signalling message refers to;
56. the MSS-Interworking layer extracts the Message type Information Element:
    - message type must be 0000 0010 (CALL PROCEEDING message);
    - change state of connection from Call Initiated(U1) into Outgoing Call Proceeding(U3) state in static table;
    - stops timer T303;
57. the MSS-Interworking layer extracts the Message length Information Element:
58. extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself);
59. because this CALL PROCEEDING message is the first message in response to the SETUP message, the Connection Identifier Information Element is mandatory;
60. the MSS-Interworking layer extracts the Connection Identifier Information Element:
    - extract the VPI value;
    - extract the VCI value;
61. save retrieved VPI/VCI value in the static table (mapping on the RVCI is established, because RVCI has been already selected);
62. the MSS-Interworking layer checks if message is errorless (Yes assumed);
63. no modification of the CALL PROCEEDING message, the Median Server Station seems to be transparent to the CALL PROCEEDING message in this scenario;
64. the MSS-Interworking layer starts timer T310;
65. IF no reaction to the CALL PROCEEDING message before expiry of timer T310 THEN initiate call/connection release;
66. CALL PROCEEDING message is delivered to the MSS-SAAL layer (it is possible to directly sent the buffered CALL PROCEEDING message (in received buffer) to the SAAL because no modification has taken place);
67. after the MSS-SAAL layer has done the necessary actions and segmented the CALL PROCEEDING message into ATM-information containers. The flow of ATM information containers is delivered to the MSS-ATM layer;
68. the MSS-ATM layer adds the ATM-header to the ATM-information container resulting in the same ATM cell corresponding to the arrived ATM-cell at the beginning (meaning that the CALL PROCEEDING message is transparently transported through the MSS-Interworking layer);
69. the ATM cells and all the necessary information being of importance to the MSS-MAC are send to the MSS-MAC.;
70. MSS-MAC delivers the bitstream to the MSS-PHY layer (MAC and PHY take care of the 'reliable' transportation of the 'CALL PROCEEDING message' to the PRP);

71. 'CALL PROCEEDING message' is transported to the MPS using the radiochannel;

72. the PRP-PHY layer delivers the constructed bitstream to the PRP-MAC layer;
73. the PRP-MAC delivers the broadcast cell to the PRP-Interworking layer;
74. the PRP-MAC delivers (if any) signalling cell(s) (VPI/VCI=0/5) to the Interworking layer;
75. the PRP-Interworking layer buffers the broadcast cell;
76. the PRP-Interworking layer buffers the signalling cell(s);
77. the PRP-Interworking layer deduces signalling broadcast cell from the broadcast cell;

78. the PRP-Interworking layer checks if signalling broadcast cell is nil;
79. if nil then delete broadcast cell and signalling broadcast cell, else continue (continue is assumed);
80. the PRP-Interworking layer deduces the RVCI and the internal address (or ATM end system address or E.164 address) from the signalling broadcast cell;
81. the PRP-Interworking layer checks if the internal address is the address of the PRP (MPS)
    - if unequal then do not proceed and delete the buffered ATM signalling cells (if any) and the broadcast cell;
    - if yes then proceed (yes is assumed);
82. the PRP-Interworking layer saves the RVCI in the static table;
83. the PRP-Interworking layer sets the in-use field in the static table to yes;
84. the PRP-Interworking layer starts timer concerning reaction of the UTE;
85. if timer expiry before answer of UTE then set the in-use field to no;
86. the PRP-Interworking layer feeds the buffered ATM (signalling) cells to the ATM-layer (where, among other things, the ATM cells are serialised into a bitstream);
87. PRP-ATM layer delivers bitstream to the PRP-PHY layer;

88. CALL PROCEEDING message is transported to the calling UTE, using the optical channel;
89. the 'CALL PROCEEDING message' arrives at the destination, calling UTE;

90. calling MPS knows that all the information needed by the ATM switch has arrived;
91. the called ATM end-user receives the SETUP message and decides to reject the call establishment;

92. the ATM switch receives the RELEASE COMPLETE message from the ATM end-user and sends a RELEASE COMPLETE message with the appropriate cause field to the calling UTE (through the MSS and the PRP) over the optical channel;

93. MSS-PHY layer receives the 'RELEASE COMPLETE message' from the ATM switch;
94. MSS-PHY layer delivers the 'RELEASE COMPLETE message' to the MSS-ATM layer;
95. MSS-ATM layer delivers the signalling ATM cells without the header to the MSS-SAAL;
96. MSS-SAAL reassembles the signalling ATM cells and delivers the RELEASE COMPLETE to the MSS-Interworking layer;
97. the MSS-Interworking layer receives the incoming signalling (RELEASE COMPLETE) message and buffers the message;
98. the MSS-Interworking layer extracts the Protocol discriminator Information Element:
    - must be 00001001 (Q.2931)
99. the MSS-Interworking layer extracts the Call reference Information Element:
    - flag must be 1 (message sent to side that originates the call reference (calling party));
100. the deduced call reference value has been saved in the static table, and uniquely identifies the call the signalling message refers to;
101. the MSS-Interworking layer extracts the Message type Information Element:
    - message type must be 0101 1010 (RELEASE COMPLETE message);
    - change state of connection from Call Proceeding(U3) into Null(U0) state in static table;
    - stop timer T310;
102. the MSS-Interworking layer extracts the Message length Information Element:
103. extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself);
104. the MSS-Interworking layer checks if message is errorless (Yes assumed);
105. deduce information needed to construct the static data base (optional);
106. the MSS-Interworking layer locally releases all the reservations made concerning the connection (connection is identified by the call reference Information element);
107. MSS-Interworking layer does not change the RELEASE COMPLETE message (only reading), and the RELEASE COMPLETE message is delivered to the MSS-SAAL;
108. after the MSS-SAAL layer has done the necessary actions and segmented the RELEASE COMPLETE message into ATM-information containers. The flow of ATM information containers is delivered to the MSS-ATM layer;
109. the MSS-ATM layer adds the ATM-header to the ATM-information container resulting in the same ATM cell corresponding to the arrived ATM-cell at the beginning (meaning that the RELEASE COMPLETE message is transparently transported through the MSS-Interworking layer);

110.the ATM cells and all the necessary information being of importance to the MSS-MAC are send to the MSS-MAC;
111.MSS-MAC delivers the bitstream to the MSS-PHY layer;

112.'RELEASE COMPLETE message' is transported to the PRP using the radiochannel;

113.the PRP-PHY layer delivers the constructed bitstream to the PRP-MAC layer;
114.the PRP-MAC delivers the broadcast cell to the PRP-Interworking layer;
115.the PRP-MAC delivers (if any) signalling cell(s) (VPI/VCI=0/5) to the Interworking layer;
116.the PRP-Interworking layer buffers the broadcast cell;
117.the PRP-Interworking layer buffers the signalling cell(s);
118.the PRP-Interworking layer deduces signalling broadcast cell from the broadcast cell;
119.the PRP-Interworking layer checks if signalling broadcast cell is nil;
120.if nil then delete broadcast cell and signalling broadcast cell, else continue (continue is assumed);
121.the PRP-Interworking layer deduces the RVCI and the internal address (or ATM end system address or E.164 address) from the signalling broadcast cell;
122.the PRP-Interworking layer checks if the internal address is the address of the PRP (MPS)
- if unequal then do not proceed and delete the buffered ATM signalling cells (if any) and the broadcast cell;
- if yes then proceed (yes is assumed);

123.the PRP-Interworking layer saves the RVCI in the static table;
124.the PRP-Interworking layer sets the in-use field in the static table to yes;
125.the PRP-Interworking layer starts timer concerning reaction of the UTE;
126.if timer expiry before answer of UTE then set the in-use field to no;
127.the PRP-Interworking layer feeds the buffered ATM (signalling) cells to the ATM-layer (where, among other things, the ATM cells are serialised into a bitstream);
128.PRP-ATM layer delivers bitstream to the PRP-PHY layer;

129.'RELEASE COMPLETE message' is transported to the calling UTE;
130.call establishment initiated by the calling UTE is rejected;

II_Setup_d

1. calling MPS wants to establish a connection with a ATM end-user;
2. calling MPS initiates call establishment by sending SETUP message to the ATM switch (through the calling PRP and the MSS);

3. UTE transmits a 'SETUP message' to the PRP using the opticalchannel;
4. calling MPS shall not include the Connection Identifier information element;
5.
6. the PRP-PHY layer delivers the received bitstream to the PRP-ATM layer;
7. the PRP-ATM layer deserialises the bitstream into ATM cells;
8. the PRP-ATM layer delivers signalling ATM cell (if any) to the PRP-Interworking layer;
9. the PRP-Interworking layer buffers (in a receive buffer) the incoming signalling ATM cells (buffering is needed because the PRP-MAC ,maybe , first has to content for a free (signalling) slot);
10. the PRP-Interworking layer starts a timer to measure the time between two successive signalling cells;
11. on expiry of the timer, set the static table inuse value to false;
12. the PRP-Interworking layer checks if the static table inuse value is true or false;
13. if inuse is false then
    - ask the MAC layer to content for an uplink (signalling) timeslot and start timer for contention time;
    - on expiry of timer delete buffered signalling ATM cell
    - the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the PRP-Interworking layer stops the timer;
    - the PRP-Interworking layer changes the static table field inuse to true;
14. if inuse is true then the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
15. the PRP-Interworking layer stops the timer;
16. the PRP-Interworking layer checks if there is another signalling ATM cell in the de-serialiser (or better, waits until the PRP-ATM layer delivers a signalling cell before timer expiry);
17. if no signalling ATM cell available in de-serialiser then change static table field inuse to false (end);
18. if ATM cell available in de-serialiser then
    - the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the PRP-Interworking layer continues this procedure until there is no signalling ATM cell in the de-serialiser;
19. the PRP-MAC layer delivers the bitstream to the PRP-PHY layer;

20. at the MSS, the MSS-PHY layer delivers the constructed bitstream to the MSS-MAC layer;
21. the MSS-MAC layer delivers the 'signalling (SETUP) message to the MSS-ATM layer;
22. the MSS-ATM layer removes the ATM cell header (among other things) and delivers the ATM information cell to the MSS-SAAL;
23. the MSS-SAAL reassembles (among other things) the signalling message. The MSS-SAAL delivers the signalling message to the MSS-Interworking layer;
24. the MSS-Interworking layer receives the incoming signalling (SETUP) message and buffers the message;
25. the MSS-Interworking layer extracts the Protocol discriminator Information Element:
    - must be 0000 1001 (Q.2931)
26. the MSS-Interworking layer extracts the Call reference Information Element:
    - flag must be 0 (message sent from the side that originates the call reference);
    - call reference value (23 bit) saved in static table;
27. the MSS-Interworking layer extracts the Message type Information Element:
    - message type must be 0000 0101 (SETUP message);
    - change state of connection from Null(U0) into Call Initiated(U1) state in static table;
    - save internal in 'originated' field in the static table;
28. the MSS-Interworking layer extracts the Message length Information Element:
    - extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself);

29. the MSS-Interworking layer extracts the ATM traffic descriptor Information Element:
    - deduce information about the forward/backward peak cell rate (CLP=0 and CLP=0+1) (if present);
    - deduce information about the forward/backward sustainable cell rate (CLP=0 and CLP=0+1) (if present);
    - deduce information about the forward/backward maximum burst size (CLP=0 and CLP=0+1) (if present);
    - save information in the static data base;
30. the MSS-Interworking layer extracts the Broadband bearer capability Information Element:
    - bearer class (BCOB-A, BCOB-C, BCOB-X(most likely to be used));
    - traffic type (CBR, VBR);
    - timing requirements (no indication, end-to-end timing required, end-to-end timing not required;
    - susceptibility to clipping (yes,no);
    - user plane connection (point-to-point, point-to-multipoint);
    - save information in the static data base;
31. the MSS-Interworking layer extracts the Called party Number Information Element and deduces the ATM end system address;
32. the MSS-Interworking layer extracts the QoS Information element. The QoS information element can contain information concerning the:
    - cell error ratio;
    - severly-errored cell block ratio;
    - cell misinsertion ratio;
    - cell transfer delay;
    - mean cell transfer delay;
    - cell delay variation;
    - save information in the static data base;
33. the MSS-Interworking layer decides if the end-user is:
    - an MEDIAN system user;
    - not an MEDIAN system user;
34. the MSS-Interworking layer checks if message is errorless (Yes assumed);
35. the MSS-Interworking layer checks if requested QoS can be provided, using the extracted QoS information. There is additional information needed about the behaviour of the overall system for example BER, bursty characteristic of errors, propagation delay, transport delay of ATM cells in ATM layer, buffer capacity in Median Server Station etc. It should be possible to deduce the necessary information from the information maintained in the static data base;
36. the MSS-Interworking layer checks if able to support combination of traffic parameters in the directions (MSS⇒ MPS and MPS⇒ MSS):
    - peak cell rate when CLP=0;
    - peak cell rate when CLP=0+1;
    - tagging (if requested);
    - sustainable cell rate CLP=0;
    - sustainable cell rate CLP=0+1;
    - maximum burst size CLP=0;
    - maximum burst size CLP=0+1.

    The MSS-Interworking layer maintains the information of the static list, therefore having the data to derive the free slots in the uplink and the downlink;
37. QoS check=okay AND traffic parameter check=okay AND non MEDIAN user (because this is scenario II_Setup_d);
38. MSS-Interworking layer: deduce time on which cells, up- and downlink, expire and retain this value in the static list (or just do selective queuing);
39. MSS-Interworking layer chooses and saves an unused RVCI for the calling party;
40. MSS-Interworking layer starts timer T303;
41. IF no reaction to the SETUP message before expiry of timer T303 THEN initiate call/connection release;
42. the signalling (SETUP) message is delivered it to the MSS-SAAL;
43. the MSS-SAAL delivers the segmented signalling (SETUP) message to the MSS-ATM layer;
44. the MSS-ATM layer delivers the bitstream to the MSS-PHY layer;

45. the 'SETUP message' is send to the ATM switch over the optical channel;
46. the ATM switch decides to send a CALL PROCEEDING message to the calling UTE;

47. CALL PROCEEDING message is transported to the calling UTE (through the MSS and the PRP), including the VPI/VCI value indicated in the Connection Identifier information element;
48. MSS-PHY layer delivers received bit stream to ATM layer;
49. MSS-ATM layer delivers the signalling ATM cells without the header to the MSS-SAAL;
50. MSS-SAAL reassembles the signalling ATM cells and delivers the signalling message (CALL PROCEEDING message) to the MSS-Interworking layer;
51. the MSS-Interworking layer receives the incoming signalling (CALL PROCEEDING) message and buffers the message;
52. the MSS-Interworking layer extracts the Protocol discriminator Information Element:
    - must be 000010001 (Q.2931)
53. the MSS-Interworking layer extracts the Call reference Information Element:
    - flag must be 1 (message sent to side that originates the call reference (calling party));
54. the deduced call reference value is also saved in static table, and uniquely identifies the call the signalling message refers to;
55. the MSS-Interworking layer extracts the Message type Information Element:
    - message type must be 0000 0010 (CALL PROCEEDING message);
    - change state of connection from Call Initiated(U1) into Outgoing Call Proceeding(U3) state in static table;
    - stops timer T303;
56. the MSS-Interworking layer extracts the Message length Information Element:
57. extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself);
58. because this CALL PROCEEDING message is the first message in response to the SETUP message, the Connection Identifier Information Element is mandatory;
59. the MSS-Interworking layer extracts the Connection Identifier Information Element:
    - extract the VPI value;
    - extract the VCI value;
60. save retrieved VPI/VCI value in the static table (mapping on the RVCI is established, because RVCI has been already selected);
61. the MSS-Interworking layer checks if message is errorless (Yes assumed);
62. no modification of the CALL PROCEEDING message, the Median Server Station seems to be transparent to the CALL PROCEEDING message in this scenario;
63. the MSS-Interworking layer starts timer T310;
64. IF no reaction to the CALL PROCEEDING message before expiry of timer T310 THEN initiate call/connection release;
65. CALL PROCEEDING message is delivered to the MSS-SAAL layer (it is possible to directly sent the buffered CALL PROCEEDING message (in received buffer) to the SAAL because no modification has taken place);
66. after the MSS-SAAL layer has done the necessary actions and segmented the CALL PROCEEDING message into ATM-information containers. The flow of ATM information containers is delivered to the MSS-ATM layer;
67. the MSS-ATM layer adds the ATM-header to the ATM-information container resulting in the same ATM cell corresponding to the arrived ATM-cell at the beginning (meaning that the CALL PROCEEDING message is transparently transported through the MSS-Interworking layer);
68. the ATM cells and all the necessary information being of importance to the MSS-MAC are send to the MSS-MAC.;
69. MSS-MAC delivers the bitstream to the MSS-PHY layer (MAC and PHY take care of the 'reliable' transportation of the 'CALL PROCEEDING message' to the PRP);

70. 'CALL PROCEEDING message' is transported to the MPS using the radiochannel;

71. the PRP-PHY layer delivers the constructed bitstream to the PRP-MAC layer;
72. the PRP-MAC delivers the broadcast cell to the PRP-Interworking layer;
73. the PRP-MAC delivers (if any) signalling cell(s) (VPI/VCI=0/5) to the Interworking layer;
74. the PRP-Interworking layer buffers the broadcast cell;
75. the PRP-Interworking layer buffers the signalling cell(s);
76. the PRP-Interworking layer deduces signalling broadcast cell from the broadcast cell;

77. the PRP-Interworking layer checks if signalling broadcast cell is nil;
78. if nil then delete broadcast cell and signalling broadcast cell, else continue (continue is assumed);
79. the PRP-Interworking layer deduces the RVCI and the internal address (or ATM end system address or E.164 address) from the signalling broadcast cell;
80. the PRP-Interworking layer checks if the internal address is the address of the PRP (MPS)
    - if unequal then do not proceed and delete the buffered ATM signalling cells (if any) and the broadcast cell;
    - if yes then proceed (yes is assumed);
81. the PRP-Interworking layer saves the RVCI in the static table;
82. the PRP-Interworking layer sets the in-use field in the static table to yes;
83. the PRP-Interworking layer starts timer concerning reaction of the UTE;
84. if timer expiry before answer of UTE then set the in-use field to no;
85. the PRP-Interworking layer feeds the buffered ATM (signalling) cells to the ATM-layer (where, among other things, the ATM cells are serialised into a bitstream);
86. PRP-ATM layer delivers bitstream to the PRP-PHY layer;

87. CALL PROCEEDING message is transported to the calling UTE, using the optical channel;
88. the 'CALL PROCEEDING message' arrives at the destination, calling UTE;

89. CALL PROCEEDING message is transported to the calling UTE, using the optical channel;
90. the 'CALL PROCEEDING message' arrives at the destination, calling UTE;

91. calling UTE knows that all the information needed by the ATM switch has arrived;
92. after precipitance of the SETUP message by the called ATM end-user, the ATM end-user decides to accept the call;
93. the called ATM end-user sends a CONNECT message to the ATM switch
94. the ATM switch sends, on receiptance of the CONNECT message, sends a CONNECT ACKNOWLEDGE message to the called ATM end-user and a CONNECT message to the calling UTE (through the MSS and the PRP);

95. MSS-PHY layer receives the 'signalling (CONNECT) message' from the ATM switch;
96. MSS-PHY layer delivers the 'signalling (CONNECT) message' to the MSS-ATM layer;
97. MSS-ATM layer delivers the signalling ATM cells without the header to the MSS-SAAL;
98. MSS-SAAL reassembles the signalling ATM cells and delivers the signalling message (CONNECT message) to the MSS-Interworking layer (synchronise by using primitives request and indication);
99. the MSS-Interworking layer receives the incoming signalling (CONNECT) message and buffers the message;
100. the MSS-Interworking layer extracts the Protocol discriminator Information Element:
    - must be 000010001 (Q.2931)
101. the MSS-Interworking layer extracts the Call reference Information Element:
    - flag must be 1 (message sent to side that originates the call reference (calling party));
102. the deduced call reference value is also saved in static table, and uniquely identifies the call the signalling message refers to;
103. the MSS-Interworking layer extracts the Message type Information Element:
    - message type must be 0000 0111 (CONNECT message);
    - change state of connection from Outgoing Call Proceeding(U3) into Active(U10) state in static table;
    - stop timer T310;
104. the MSS-Interworking layer extracts the Message length Information Element:
105. extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself);
106. the MSS-Interworking layer checks if message is errorless (Yes assumed);
107. no modification of the CONNECT message, the Median Server Station seems to be transparent to the CONNECT message in this scenario;
108. the MSS-Interworking layer starts timer Txxx (timer for con.ack. from UTE);
109. IF no reaction to the CONNECT message before expiry of timer Txxx THEN initiate call/connection release;
110. CONNECT message is delivered to the MSS-SAAL layer (it is possible to directly sent the buffered CONNECT message to the SAAL because no modification has taken place);

111.after the MSS-SAAL layer has done the necessary actions and segmented the CONNECT message into ATM-information containers. The flow of ATM information containers is delivered to the MSS-ATM layer;
112.the MSS-ATM layer adds the ATM-header to the ATM-information container resulting in the same ATM cell corresponding to the arrived ATM-cell at the beginning (meaning that the CONNECT message is transparently transported through the MSS-Interworking layer);
113.the ATM cells and all the necessary information being of importance to the MSS-MAC are send to the MSS-MAC. There is a way of synchronisation needed between the MSS-MAC and the MSS-Interworking layer which can be accomplished by the use of the primitives: request and indication with there own message indication;
114.MSS-MAC delivers the bitstream to the MSS-PHY layer;

115.'CONNECT message' is transported to the PRP using the radiochannel;

116.the PRP-PHY layer delivers the constructed bitstream to the PRP-MAC layer;
117.the PRP-MAC delivers the broadcast cell to the PRP-Interworking layer;
118.the PRP-MAC delivers (if any) signalling cell(s) (VPI/VCI=0/5) to the Interworking layer;
119.the PRP-Interworking layer buffers the broadcast cell;
120.the PRP-Interworking layer buffers the signalling cell(s);
121.the PRP-Interworking layer deduces signalling broadcast cell from the broadcast cell;
122.the PRP-Interworking layer checks if signalling broadcast cell is nil;
123.if nil then delete broadcast cell and signalling broadcast cell, else continue (continue is assumed);
124.the PRP-Interworking layer deduces the RVCI and the internal address (or ATM end system address or E.164 address) from the signalling broadcast cell;
125.the PRP-Interworking layer checks if the internal address is the address of the PRP (MPS)
- if unequal then do not proceed and delete the buffered ATM signalling cells (if any) and the broadcast cell;
- if yes then proceed (yes is assumed);

126.the PRP-Interworking layer saves the RVCI in the static table;
127.the PRP-Interworking layer sets the in-use field in the static table to yes;
128.the PRP-Interworking layer starts timer concerning reaction of the UTE;
129.if timer expiry before answer of UTE then set the in-use field to no;
130.the PRP-Interworking layer feeds the buffered ATM (signalling) cells to the ATM-layer (where, among other things, the ATM cells are serialised into a bitstream);
131.PRP-ATM layer delivers bitstream to the PRP-PHY layer;

132.'CONNECT message' is transported to the UTE-PHY layer, using the optical channel;
133.the 'CONNECT message' arrives at the destination, UTE

134.the UTE now knows that the call establishment has been awarded and decides to send a CONNECT ACKNOWLEDGE to the ATM switch (through the PRP and the MEDIAN server station;

135.the PRP-PHY layer delivers the received bitstream to the PRP-ATM layer;
136.the PRP-ATM layer deserialises the bitstream into ATM cells;
137.the PRP-ATM layer delivers signalling ATM cell (if any) to the PRP-Interworking layer;
138.the PRP-Interworking layer buffers (in a receive buffer) the incoming signalling ATM cells (buffering is needed because the PRP-MAC ,maybe , first has to content for a free (signalling) slot);
139.the PRP-Interworking layer starts a timer to measure the time between two successive signalling cells;
140.on expiry of the timer, set the static table inuse value to false;
141.the PRP-Interworking layer checks if the static table inuse value is true or false;
142.if inuse is false then
- ask the MAC layer to content for an uplink (signalling) timeslot and start timer for contention time;
- on expiry of timer delete buffered signalling ATM cell
- the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
- the PRP-Interworking layer stops the timer;
- the PRP-Interworking layer changes the static table field inuse to true;

143.if inuse is true then the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
144.the PRP-Interworking layer stops the timer;

145.the PRP-Interworking layer checks if there is another signalling ATM cell in the de-serialiser (or better, waits until the PRP-ATM layer delivers a signalling cell before timer expiry);
146.if no signalling ATM cell available in de-serialiser then change static table field inuse to false (end);
147.if ATM cell available in de-serialiser then
- the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
- the PRP-Interworking layer continues this procedure until there is no signalling ATM cell in the de-serialiser;

148.the PRP-MAC layer delivers the bitstream to the PRP-PHY layer;

149.the 'CONNECT ACKNOWLEDGE message' is transported to the MSS over the radio channel;.

150.at the MSS the MSS-PHY layer delivers the constructed bitstream to the MSS-MAC layer;
151.the MSS-MAC layer delivers the 'CONNECT ACKNOWLEDGE message' (segmented in ATM cells) to the MSS-ATM layer;
152.the MSS-ATM layer removes the ATM cell header and delivers the ATM information cell to the MSS-SAAL;
153.the MSS-SAAL reassembles (among other things) the 'CONNECT ACKNOWLEDGE message'. The MSS-SAAL delivers the CONNECT ACKNOWLEDGE message to the MSS-Interworking layer;
154.the MSS-Interworking layer receives the incoming signalling (CONNECT ACKNOWLEDGE) message and buffers the message;
155.the MSS-Interworking layer extracts the Protocol discriminator Information Element:
- must be 000010001 (Q.2931)

156.the MSS-Interworking layer extracts the Call reference Information Element:
- flag must be 0 (message sent from side that originates the call reference (calling party));

157.the MSS-Interworking layer extracts the Message type Information Element:
- message type must be 0000 1111 (CONNECT ACKNOWLEDGE message);
- stop timer Txxx

158.the MSS-Interworking layer extracts the Message length Information Element:
159.extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself);
160.the MSS-Interworking layer checks if message is errorless (Yes assumed);
161.the MSS-Interworking layer delivers the received (buffered) message to the MSS-SAAL;
162.the CONNECT ACKNOWLEDGE message is delivered it to the MSS-SAAL;
163.the MSS-SAAL delivers the segmented 'CONNECT ACKNOWLEDGE' message to the MSS-ATM layer (synchronised);
164.the MSS-ATM layer delivers the bitstream to the MSS-PHY layer;

165.the 'CONNECT ACKNOWLEDGE message' is send to the ATM switch over the optical channel;

166.the ATM-switch receives the CONNECT ACKNOWLEDGE message and call establishment has been successful;

III_Setup_a

1. MPS in MEDIAN system wants to setup a connection with another MPS;
2. calling UTE transmits an 'SETUP message' to the PRP using the opticalchannel;

3. the CALLING-PRP-PHY layer delivers the received bitstream to the CALLING-PRP-ATM layer;
4. the CALLING-PRP-ATM layer deserialises the bitstream into ATM cells;
5. the CALLING-PRP-ATM layer delivers signalling ATM cell (if any) to the CALLING-PRP-Interworking layer;
6. the CALLING-PRP-Interworking layer buffers (in a receive buffer) the incoming signalling ATM cells (buffering is needed because the CALLING-PRP-MAC ,maybe , first has to content for a free (signalling) slot);
7. the CALLING-PRP-Interworking layer starts a timer to measure the time between two successive signalling cells;
8. on expiry of the timer, set the static table inuse value to false;
9. the CALLING-PRP-Interworking layer checks if the static table inuse value is true or false;
10. if inuse is false then
    - ask the MAC layer to content for an uplink (signalling) timeslot and start timer for contention time;
    - on expiry of timer delete buffered signalling ATM cell
    - the CALLING-PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the CALLING-PRP-Interworking layer stops the timer;
    - the CALLING-PRP-Interworking layer changes the static table field inuse to true;
11. if inuse is true then the CALLING-PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
12. the CALLING-PRP-Interworking layer stops the timer;
13. the CALLING-PRP-Interworking layer checks if there is another signalling ATM cell in the de-serialiser (or better, waits until the CALLING-PRP-ATM layer delivers a signalling cell before timer expiry);
14. if no signalling ATM cell available in de-serialiser then change static table field inuse to false (end);
15. if ATM cell available in de-serialiser then
    - the CALLING-PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the CALLING-PRP-Interworking layer continues this procedure until there is no signalling ATM cell in the de-serialiser;
16. the CALLING-PRP-MAC layer delivers the bitstream to the CALLING-PRP-PHY layer;

17. the 'SETUP message' is transported to the MSS over the radio channel;

18. at the MSS the MSS-PHY layer delivers the constructed bitstream to the MSS-MAC layer;
19. the MSS-MAC layer delivers the SETUP message (segmented in ATM cells) to the MSS-ATM layer;
20. the MSS-ATM layer removes the ATM cell header and delivers the ATM information cell to the MSS-SAAL;
21. the MSS-SAAL reassembles (among other things) the SETUP message. The MSS-SAAL delivers the SETUP message to the MSS-Interworking layer;
22. the MSS-Interworking layer receives the incoming signalling (SETUP) message and buffers the message;
23. the MSS-Interworking layer extracts the Protocol discriminator Information Element:
    - must be 0000 1001 (Q.2931)
24. the MSS-Interworking layer extracts the Call reference Information Element:
    - flag must be 0(message sent from the side that originates the call reference);
    - call reference value (23 bit) saved in static table;
25. the MSS-Interworking layer extracts the Message type Information Element:
    - message type must be 0000 0101 (SETUP message);
    - change state of connection from Null(U0) into Call initiated(U1) state in static table;
    - save internal in call originated field in the static table;
26. the MSS-Interworking layer extracts the Message length Information Element:
    - extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself);

27. the MSS-Interworking layer extracts the ATM traffic descriptor Information Element:
    - deduce information about the forward/backward peak cell rate (CLP=0 and CLP=0+1) (if present);
    - deduce information about the forward/backward sustainable cell rate (CLP=0 and CLP=0+1) (if present);
    - deduce information about the forward/backward maximum burst size (CLP=0 and CLP=0+1) (if present);
    - save information in the static data base;
28. the MSS-Interworking layer extracts the Broadband bearer capability Information Element:
    - bearer class (BCOB-A, BCOB-C, BCOB-X(most likely to be used));
    - traffic type (CBR, VBR);
    - timing requirements (no indication, end-to-end timing required, end-to-end timing not required;
    - susceptibility to clipping (yes,no);
    - user plane connection (point-to-point, point-to-multipoint);
    - save information in the static data base;
29. the MSS-Interworking layer extracts the Called party Number Information Element and deduces the ATM end system address;
30. the MSS-Interworking layer extracts the QoS Information element, which can contain information concerning the:
    - cell error ratio;
    - severly-errored cell block ratio;
    - cell misinsertion ratio;
    - cell transfer delay;
    - mean cell transfer delay;
    - cell delay variation.
    - save information in the static data base;
31. the MSS-Interworking layer deduces if the end-user is:
    - an MEDIAN system end-user (is valid in this scenario);
    - not an MEDIAN system end-user;
32. the MSS-Interworking layer checks if message is errorless (Yes assumed);
33. the MSS-Interworking layer checks if requested QoS can be provided, using the extracted QoS information. There is additional information needed about the behaviour of the overall system for example BER, bursty characteristic of errors, propagation delay, transport delay of ATM cells in ATM layer, buffer capacity in Median Server Station etc.;
34. the MSS-Interworking layer checks if able to support combination of traffic parameters in both the directions (MSS⇒ MPS and MPS⇒ MSS). The MSS-Interworking layer maintains the information of the static list, therefore having the data to derive the free slots in the uplink and the downlink;
35. the MSS-Interworking layer checks if a RVCI is available for the connection form the MSS to the calling UTE and from MSS to called UTE;
36. at least one of the previous checks is not okay (because this is scenario III_Setup_a);
37. change state form Call initiated(U1) into Null(U0) in static table;
38. if QoS check not okay then cause#49 = QoS unavailable;
39. if traffic parameter check not okay then cause#51 = user cell rate unavailable;
40. if traffic parameter contains a non supported set of traffic parameters then cause#73 = unsupported combination of traffic parameters;
41. if less then 2 RVCI's available then cause#45 no VPCI/VCI available;
42. MSS_interworking layer terminates call establishment by not sending the SETUP message to the called UTE and by initiating sending of RELEASE COMPLETE message to the calling UTE;
43. before the MSS-Interworking is able to send the RELEASE COMPLETE message, extract additional information to generate the cause field. The RELEASE COMPLETE message consists of:
    - protocol discriminator ⇒ 0000 1001 (same as in the SETUP message);
    - call reference ⇒ same as in the SETUP message;
    - message type ⇒ 0001 1010 (RELEASE COMPLETE message);
    - message length ⇒ number of octets of the message contents, excluding the octets used for 'protocol discriminator', ',message type', and message length indication itself
    - cause ⇒ depending on the reason which caused the rejection of the call;

44. MSS-Interworking layer generates cause field subfields:
    - length of cause information contents;
    - location (describing the location of call rejection);
    - cause value (describing the reason of rejection);
    - diagnostic(s)(if any);
45. MSS-Interworking layer calculates the message length;
46. MSS-Interworking layer deduces statistical data (what ever this maybe) for static database
47. MSS-Interworking layer locally releases the reservations made for this connection establishment;
48. MSS-Interworking layer delivers the RELEASE COMPLETE message to the SAAL layer;
49. after the MSS-SAAL layer has done the necessary actions and segmented the setup message into ATM-information containers, it delivers the ATM-information containers to the MSS-ATM layer;
50. the MSS-ATM layer adds the ATM-header to the ATM-information container;
51. the ATM cells and all the necessary information being of importance to the MSS-MAC are send to the MSS-MAC;
52. MSS-MAC delivers the bitstream to the MSS-PHY layer (MAC and PHY take care of the 'reliable' transportation of the RELEASE COMPLETE message to the PRP);

53. RELEASE COMPLETE message is transported to the PRP using the radiochannel;

54. the CALLING-PRP-PHY layer delivers the constructed bitstream to the CALLING-PRP-MAC layer;
55. the CALLING-PRP-MAC delivers the broadcast cell to the CALLING-PRP-Interworking layer;
56. the CALLING-PRP-MAC delivers (if any) signalling cell(s) (VPI/VCI=0/5) to the Interworking layer;
57. the CALLING-PRP-Interworking layer buffers the broadcast cell;
58. the CALLING-PRP-Interworking layer buffers the signalling cell(s);
59. the CALLING-PRP-Interworking layer deduces signalling broadcast cell from the broadcast cell;
60. the CALLING-PRP-Interworking layer checks if signalling broadcast cell is nil;
61. if nil then delete broadcast cell and signalling broadcast cell, else continue (continue is assumed);
62. the CALLING-PRP-Interworking layer deduces the RVCI and the internal address (or ATM end system address or E.164 address) from the signalling broadcast cell;
63. the CALLING-PRP-Interworking layer checks if the internal address is the address of the CALLING-PRP (MPS)
    - if unequal then do not proceed and delete the buffered ATM signalling cells (if any) and the broadcast cell;
    - if yes then proceed (yes is assumed);
64. the CALLING-PRP-Interworking layer saves the RVCI in the static table;
65. the CALLING-PRP-Interworking layer sets the in-use field in the static table to yes;
66. the CALLING-PRP-Interworking layer starts timer concerning reaction of the UTE;
67. if timer expiry before answer of UTE then set the in-use field to no;
68. the CALLING-PRP-Interworking layer feeds the buffered ATM (signalling) cells to the ATM-layer (where, among other things, the ATM cells are serialised into a bitstream);
69. CALLING-PRP-ATM layer delivers bitstream to the CALLING-PRP-PHY layer;

70. 'RELEASE COMPLETE message' is transported to the calling UTE using the optical channel;

71. UTE knows that call establishment has been rejected;

III_Setup_b

1. MPS in MEDIAN system wants to setup a connection with another MPS;
2. calling UTE transmits an 'SETUP message' to the PRP using the opticalchannel;

3. the CALLING-PRP-PHY layer delivers the received bitstream to the CALLING-PRP-ATM layer;
4. the CALLING-PRP-ATM layer deserialises the bitstream into ATM cells;
5. the CALLING-PRP-ATM layer delivers signalling ATM cell (if any) to the CALLING-PRP-Interworking layer;
6. the CALLING-PRP-Interworking layer buffers (in a receive buffer) the incoming signalling ATM cells (buffering is needed because the CALLING-PRP-MAC ,maybe , first has to content for a free (signalling) slot);
7. the CALLING-PRP-Interworking layer starts a timer to measure the time between two successive signalling cells;
8. on expiry of the timer, set the static table inuse value to false;
9. the CALLING-PRP-Interworking layer checks if the static table inuse value is true or false;
10. if inuse is false then
    - ask the MAC layer to content for an uplink (signalling) timeslot and start timer for contention time;
    - on expiry of timer delete buffered signalling ATM cell
    - the CALLING-PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the CALLING-PRP-Interworking layer stops the timer;
    - the CALLING-PRP-Interworking layer changes the static table field inuse to true;
11. if inuse is true then the CALLING-PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
12. the CALLING-PRP-Interworking layer stops the timer;
13. the CALLING-PRP-Interworking layer checks if there is another signalling ATM cell in the de-serialiser (or better, waits until the CALLING-PRP-ATM layer delivers a signalling cell before timer expiry);
14. if no signalling ATM cell available in de-serialiser then change static table field inuse to false (end);
15. if ATM cell available in de-serialiser then
    - the CALLING-PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the CALLING-PRP-Interworking layer continues this procedure until there is no signalling ATM cell in the de-serialiser;
16. the CALLING-PRP-MAC layer delivers the bitstream to the CALLING-PRP-PHY layer;

17. the 'SETUP message' is transported to the MSS over the radio channel;

18. at the MSS the MSS-PHY layer delivers the constructed bitstream to the MSS-MAC layer;
19. the MSS-MAC layer delivers the SETUP message (segmented in ATM cells) to the MSS-ATM layer;
20. the MSS-ATM layer removes the ATM cell header and delivers the ATM information cell to the MSS-SAAL;
21. the MSS-SAAL reassembles (among other things) the SETUP message. The MSS-SAAL delivers the SETUP message to the MSS-Interworking layer;
22. the MSS-Interworking layer receives the incoming signalling (SETUP) message and buffers the message;
23. the MSS-Interworking layer extracts the Protocol discriminator Information Element:
    - must be 0000 1001 (Q.2931)
24. the MSS-Interworking layer extracts the Call reference Information Element:
    - flag must be 0(message sent from the side that originates the call reference);
    - call reference value (23 bit) saved in static table;
25. the MSS-Interworking layer extracts the Message type Information Element:
    - message type must be 0000 0101 (SETUP message);
    - change state of connection from Null(U0) into Call initiated(U1) state in static table;
    - save internal in call originated field in the static table;
26. the MSS-Interworking layer extracts the Message length Information Element:
    - extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself);

27. the MSS-Interworking layer extracts the ATM traffic descriptor Information Element:
    - deduce information about the forward/backward peak cell rate (CLP=0 and CLP=0+1) (if present);
    - deduce information about the forward/backward sustainable cell rate (CLP=0 and CLP=0+1) (if present);
    - deduce information about the forward/backward maximum burst size (CLP=0 and CLP=0+1) (if present);
    - save information in the static data base;
28. the MSS-Interworking layer extracts the Broadband bearer capability Information Element:
    - bearer class (BCOB-A, BCOB-C, BCOB-X(most likely to be used));
    - traffic type (CBR, VBR);
    - timing requirements (no indication, end-to-end timing required, end-to-end timing not required;
    - susceptibility to clipping (yes,no);
    - user plane connection (point-to-point, point-to-multipoint);
    - save information in the static data base;
29. the MSS-Interworking layer extracts the Called party Number Information Element and deduces the ATM end system address;
30. the MSS-Interworking layer extracts the QoS Information element, which can contain information concerning the:
    - cell error ratio;
    - severly-errored cell block ratio;
    - cell misinsertion ratio;
    - cell transfer delay;
    - mean cell transfer delay;
    - cell delay variation.
    - save information in the static data base;
31. the MSS-Interworking layer deduces if the end-user is:
    - an MEDIAN system end-user (is valid in this scenario);
    - not an MEDIAN system end-user;
32. the MSS-Interworking layer checks if message is errorless (Yes assumed);
33. the MSS-Interworking layer checks if requested QoS can be provided, using the extracted QoS information. There is additional information needed about the behaviour of the overall system for example BER, bursty characteristic of errors, propagation delay, transport delay of ATM cells in ATM layer, buffer capacity in Median Server Station etc. It should be possible to deduce the necessary information from the information maintained in the static data base;
34. the MSS-Interworking layer checks if able to support combination of traffic parameters in both the directions (MSS$\Rightarrow$ MPS and MPS$\Rightarrow$ MSS). The MSS-Interworking layer maintains the information of the static list, therefore having the data to derive the free slots in the uplink and the downlink;
35. the MSS-Interworking layer checks if a RVCI is available for the connection form the MSS to the calling UTE and from MSS to called UTE;
36. the previous checks are okay (because this is scenario III_Setup_b);
37. MSS-Interworking layer chooses a unused VPCI/VCI combination. The VPI/VCI information is saved in the static list;
38. MSS-Interworking layer chooses a unused RVCI for the called MEDIAN party (saved in static list);
39. MSS-Interworking layer chooses a unused RVCI for the calling MEDIAN party (saved in static list);
40. MSS-Interworking layer: deduce time on which cells belonging to the selected RVCI (eq. VPCI/VCI) expire and retain this value in the static list (in row belonging to the calling and the called MEDIAN party). The needed values, forward and backward cell transfer delay and forward and backward mean cell transfer delay; can be found in the QoS parameter information element located in the SETUP message;
41. copy call reference value form called party row to calling party row;
42. copy VPCI/VCI value form called party row to calling party row;
43. state from called RVCI is changed from Null(U0) into Call Present (U6);

44. modification of the setup message:
    - generate Connection Identifier Information Element:
    - the MSS-Interworking layer inserts the chosen VPI/VCI (VPI/RVCI) (on the same way the ATM switch would) in the default structure of the Connection Identifier Information Element;
    - this message is saved, because it also has to be added to the CALL PROCEEDING message which will be sent by the called UTE to the MSS (and which has to be sent by the MSS to the calling UTE);
45. MSS-Interworking layer starts timer T303;
46. IF no reaction to the SETUP message before expiry of timer T303 THEN initiate call/connection release;
47. signalling (SETUP) message is delivered to the MSS-SAAL layer;
48. after the MSS-SAAL layer has done the necessary actions and segmented the signalling message into ATM-information containers, it delivers the ATM-information containers to the MSS-ATM layer;
49. the MSS-ATM layer adds the ATM-header to the ATM-information container ;
50. the ATM cells and all the necessary information being of importance to the MSS-MAC are send to the MSS-MAC;
51. MSS-MAC delivers the bitstream to the MSS-PHY layer (MAC and PHY take care of the 'reliable' transportation of the SETUP message to the PRP);

52. 'SETUP message' is transported to the called-PRP using the radiochannel;

53. the CALLED-PRP-PHY layer delivers the constructed bitstream to the CALLED-PRP-MAC layer;
54. the CALLED-PRP-MAC delivers the broadcast cell to the CALLED-PRP-Interworking layer;
55. the CALLED-PRP-MAC delivers (if any) signalling cell(s) (VPI/VCI=0/5) to the Interworking layer;
56. the CALLED-PRP-Interworking layer buffers the broadcast cell;
57. the CALLED-PRP-Interworking layer buffers the signalling cell(s);
58. the CALLED-PRP-Interworking layer deduces signalling broadcast cell from the broadcast cell;
59. the CALLED-PRP-Interworking layer checks if signalling broadcast cell is nil;
60. if nil then delete broadcast cell and signalling broadcast cell, else continue (continue is assumed);
61. the CALLED-PRP-Interworking layer deduces the RVCI and the internal address (or ATM end system address or E.164 address) from the signalling broadcast cell;
62. the CALLED-PRP-Interworking layer checks if the internal address is the address of the CALLED-PRP (MPS)
    - if unequal then do not proceed and delete the buffered ATM signalling cells (if any) and the broadcast cell;
    - if yes then proceed (yes is assumed);
63. the CALLED-PRP-Interworking layer saves the RVCI in the static table;
64. the CALLED-PRP-Interworking layer sets the in-use field in the static table to yes;
65. the CALLED-PRP-Interworking layer starts timer concerning reaction of the UTE;
66. if timer expiry before answer of UTE then set the in-use field to no;
67. the CALLED-PRP-Interworking layer feeds the buffered ATM (signalling) cells to the ATM-layer (where, among other things, the ATM cells are serialised into a bitstream);
    CALLED-PRP-ATM layer delivers bitstream to the CALLED-PRP-PHY layer;

68. 'SETUP message' is transported to the called-UTE using the optical channel;

69. the called-UTE decides to send a CALL PROCEEDING message to the calling UTE (through the MSS);
70. 'CALL PROCEEDING message' is transported to the called-PRP using the optical channel;

71. the CALLED-PRP-PHY layer delivers the received bitstream to the CALLED-PRP-ATM layer;
72. the CALLED-PRP-ATM layer deserialises the bitstream into ATM cells;
73. the CALLED-PRP-ATM layer delivers signalling ATM cell (if any) to the CALLED-PRP-Interworking layer;
74. the CALLED-PRP-Interworking layer buffers (in a receive buffer) the incoming signalling ATM cells (buffering is needed because the CALLED-PRP-MAC ,maybe , first has to content for a free (signalling) slot);
75. the CALLED-PRP-Interworking layer starts a timer to measure the time between two successive signalling cells;
76. on expiry of the timer, set the static table inuse value to false;
77. the CALLED-PRP-Interworking layer checks if the static table inuse value is true or false;

78. if inuse is false then
    - ask the MAC layer to content for an uplink (signalling) timeslot and start timer for contention time;
    - on expiry of timer delete buffered signalling ATM cell
    - the CALLED-PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the CALLED-PRP-Interworking layer stops the timer;
    - the CALLED-PRP-Interworking layer changes the static table field inuse to true;
79. if inuse is true then the CALLED-PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
80. the CALLED-PRP-Interworking layer stops the timer;
81. the CALLED-PRP-Interworking layer checks if there is another signalling ATM cell in the de-serialiser (or better, waits until the CALLED-PRP-ATM layer delivers a signalling cell before timer expiry);
82. if no signalling ATM cell available in de-serialiser then change static table field inuse to false (end);
83. if ATM cell available in de-serialiser then
    - the CALLED-PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the CALLED-PRP-Interworking layer continues this procedure until there is no signalling ATM cell in the de-serialiser;
84. the CALLED-PRP-MAC layer delivers the bitstream to the CALLED-PRP-PHY layer;
85. the 'signalling (CALL PROCEEDING) message' is transported to the MSS over the radio channel;.

86. at the MSS the MSS-PHY layer delivers the constructed bitstream to the MSS-MAC layer;
87. the MSS-MAC layer delivers the 'signalling (CALL PROCEEDING) message' to the MSS-ATM layer;
88. the MSS-ATM layer removes the ATM cell header and delivers the ATM information cell to the MSS-SAAL;
89. the MSS-SAAL reassembles (among other things) the received signalling ATM information cell containers. The MSS-SAAL delivers the CALL PROCEEDING message to the MSS-Interworking layer;
90. the MSS-Interworking layer receives the incoming signalling (CALL PROCEEDING) message and buffers the message;
91. the MSS-Interworking layer extracts the Protocol discriminator Information Element:
    - must be 000010001 (Q.2931)
92. the MSS-Interworking layer extracts the Call reference Information Element:
    - flag must be 1 (message sent to side that originates the call reference (calling party));
93. the deduced call reference value is also saved in static table, and uniquely identifies the call the signalling message refers to;
94. the MSS-Interworking layer extracts the Message type Information Element:
    - message type must be 0000 0010 (CALL PROCEEDING message);
    - change state from calling party from Call Initiated(U1) into Incoming Call Proceeding(U9) state in static table;
    - change state from called party from Call Present(U6) into Outgoing Call Proceeding(U3) state in static table;
    - stop timer T303;
95. the MSS- Interworking layer adds the saved Connection identifier Information Element to the CALL PROCEEDING message;
96. the MSS-Interworking layer extracts the Message length Information Element;
97. the MSS-Interworking layer checks if message is errorless (Yes assumed);
98. the MSS- Interworking layer calculates the new length of the signalling message (old length + length (connection identifier Information element) and changes the Message length Information Element according to the calculated length;
99. the MSS- Interworking layer starts timer T310
100. IF no reaction to the CALL PROCEEDING message before expiry of timer T310 THEN initiate call/connection release;
101. MSS-Interworking layer delivers the changed signalling (CALL PROCEEDING) message to the MSS-SAAL (to the calling MEDIAN party);
102. after the MSS-SAAL layer has done the necessary actions and segmented the setup message into ATM-information containers, it delivers the ATM-information containers to the MSS-ATM layer;
103. the MSS-ATM layer adds the ATM-header to the ATM-information container;
104. the ATM cells and all the necessary information being of importance to the MSS-MAC are send to the MSS-MAC. There is a way of synchronisation needed between the MSS-MAC and the MSS-Interworking

layer which can be accomplished by the use of the primitives: request and indication with there own message indication;

105.MSS-MAC delivers the bitstream to the MSS-PHY layer (MAC and PHY take care of the 'reliable' transportation of the CALL PROCEEDING message to the PRP);

106.CALL PROCEEDING message is transported to the calling-PRP using the radiochannel;

107.the CALLING-PRP-PHY layer delivers the constructed bitstream to the CALLING-PRP-MAC layer;
108.the CALLING-PRP-MAC delivers the broadcast cell to the CALLING-PRP-Interworking layer;
109.the CALLING-PRP-MAC delivers (if any) signalling cell(s) (VPI/VCI=0/5) to the Interworking layer;
110.the CALLING-PRP-Interworking layer buffers the broadcast cell;
111.the CALLING-PRP-Interworking layer buffers the signalling cell(s);
112.the CALLING-PRP-Interworking layer deduces signalling broadcast cell from the broadcast cell;
113.the CALLING-PRP-Interworking layer checks if signalling broadcast cell is nil;
114.if nil then delete broadcast cell and signalling broadcast cell, else continue (continue is assumed);
115.the CALLING-PRP-Interworking layer deduces the RVCI and the internal address (or ATM end system address or E.164 address) from the signalling broadcast cell;
116.the CALLING-PRP-Interworking layer checks if the internal address is the address of the CALLING-PRP (MPS)
- if unequal then do not proceed and delete the buffered ATM signalling cells (if any) and the broadcast cell;
- if yes then proceed (yes is assumed);

117.the CALLING-PRP-Interworking layer saves the RVCI in the static table;
118.the CALLING-PRP-Interworking layer sets the in-use field in the static table to yes;
119.the CALLING-PRP-Interworking layer starts timer concerning reaction of the UTE;
120.if timer expiry before answer of UTE then set the in-use field to no;
121.the CALLING-PRP-Interworking layer feeds the buffered ATM (signalling) cells to the ATM-layer (where, among other things, the ATM cells are serialised into a bitstream);
122.CALLING-PRP-ATM layer delivers bitstream to the CALLING-PRP-PHY layer;

123.'CALL PROCEEDING message' is transported to the calling-UTE-PHY layer, using the optical channel;
124.the called UTE decides to reject the requested call establishment;
125.the called UTE sends a RELEASE COMPLETE message to the calling-UTE;

126.'RELEASE COMPLETE message' is transported to the called-PRP using the optical channel;

127.the CALLED-PRP-PHY layer delivers the received bitstream to the CALLED-PRP-ATM layer;
128.the CALLED-PRP-ATM layer deserialises the bitstream into ATM cells;
129.the CALLED-PRP-ATM layer delivers signalling ATM cell (if any) to the CALLED-PRP-Interworking layer;
130.the CALLED-PRP-Interworking layer buffers (in a receive buffer) the incoming signalling ATM cells (buffering is needed because the CALLED-PRP-MAC ,maybe , first has to content for a free (signalling) slot);
131.the CALLED-PRP-Interworking layer starts a timer to measure the time between two successive signalling cells;
132.on expiry of the timer, set the static table inuse value to false;
133.the CALLED-PRP-Interworking layer checks if the static table inuse value is true or false;
134.if inuse is false then
- ask the MAC layer to content for an uplink (signalling) timeslot and start timer for contention time;
- on expiry of timer delete buffered signalling ATM cell
- the CALLED-PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
- the CALLED-PRP-Interworking layer stops the timer;
- the CALLED-PRP-Interworking layer changes the static table field inuse to true;

135.if inuse is true then the CALLED-PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
136.the CALLED-PRP-Interworking layer stops the timer;
137.the CALLED-PRP-Interworking layer checks if there is another signalling ATM cell in the de-serialiser (or better, waits until the CALLED-PRP-ATM layer delivers a signalling cell before timer expiry);

138.if no signalling ATM cell available in de-serialiser then change static table field inuse to false (end);
139.if ATM cell available in de-serialiser then
- the CALLED-PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
- the CALLED-PRP-Interworking layer continues this procedure until there is no signalling ATM cell in the de-serialiser;

140.the CALLED-PRP-MAC layer delivers the bitstream to the CALLED-PRP-PHY layer;

141.the 'RELEASE COMPLETE message' is transported to the MSS over the radio channel;.

142.at the MSS the MSS-PHY layer delivers the constructed bitstream to the MSS-MAC layer;
143.the MSS-MAC layer delivers the 'signalling (RELEASE COMPLETE) message' (segmented in ATM cells plus additional overhead) to the MSS-ATM layer;
144.the MSS-ATM layer removes the ATM cell header and delivers the ATM information cell to the MSS-SAAL;
145.the MSS-SAAL reassembles (among other things) the received signalling ATM information cell containers. The MSS-SAAL delivers the RELEASE COMPLETE message to the MSS-Interworking layer;
146.the MSS-Interworking layer receives the incoming signalling (RELEASE COMPLETE) message and buffers the message;
147.the MSS-Interworking layer extracts the Protocol discriminator Information Element:
- must be 00001000l (Q.2931)

148.the MSS-Interworking layer extracts the Call reference Information Element:
- flag must be 1 (message sent to side that originates the call reference (calling party));

149.the deduced call reference value has been saved in the static table, and uniquely identifies the call the signalling message refers to;
150.the MSS-Interworking layer extracts the Message type Information Element:
- message type must be 0101 1010 (RELEASE COMPLETE message);
- change state from calling party from Incoming Call Proceeding(U9) into Null(U0) state in static table;
- change state from called party from Outgoing Call Proceeding(U3) into Null(U0) state in static table;
- stop timer T310;

151.the MSS-Interworking layer extracts the Message length Information Element:
152.extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself);
153.the MSS-Interworking layer extracts the Cause Information Element;
154.the MSS-Interworking layer checks if message is errorless (Yes assumed);
155.deduce information needed to construct the static data base (optional);
156.the MSS-Interworking layer locally releases all the reservations made concerning the connection (connection is identified by the call reference Information element)(both the RVCI's);
157.MSS-Interworking layer does not change the RELEASE COMPLETE message (only reading), and the RELEASE COMPLETE message is delivered to the MSS-SAAL;
158.after the MSS-SAAL layer has done the necessary actions and segmented the setup message into ATM-information containers, it delivers the ATM-information containers to the MSS-ATM layer;
159.the MSS-ATM layer adds the ATM-header to the ATM-information container;
160.the ATM cells and all the necessary information being of importance to the MSS-MAC are send to the MSS-MAC. There is a way of synchronisation needed between the MSS-MAC and the MSS-Interworking layer which can be accomplished by the use of the primitives: request and indication with there own message indication;
161.MSS-MAC delivers the bitstream to the MSS-PHY layer (MAC and PHY take care of the 'reliable' transportation of the RELEASE COMPLETE message to the PRP);

162.RELEASE COMPLETE message is transported to the calling-PRP using the radiochannel;

163.the CALLING-PRP-PHY layer delivers the constructed bitstream to the CALLING-PRP-MAC layer;
164.the CALLING-PRP-MAC delivers the broadcast cell to the CALLING-PRP-Interworking layer;
165.the CALLING-PRP-MAC delivers (if any) signalling cell(s) (VPI/VCI=0/5) to the Interworking layer;
166.the CALLING-PRP-Interworking layer buffers the broadcast cell;
167.the CALLING-PRP-Interworking layer buffers the signalling cell(s);
168.the CALLING-PRP-Interworking layer deduces signalling broadcast cell from the broadcast cell;
169.the CALLING-PRP-Interworking layer checks if signalling broadcast cell is nil;

170. if nil then delete broadcast cell and signalling broadcast cell, else continue (continue is assumed);
171. the CALLING-PRP-Interworking layer deduces the RVCI and the internal address (or ATM end system address or E.164 address) from the signalling broadcast cell;
172. the CALLING-PRP-Interworking layer checks if the internal address is the address of the CALLING-PRP (MPS)
    - if unequal then do not proceed and delete the buffered ATM signalling cells (if any) and the broadcast cell;
    - if yes then proceed (yes is assumed);
173. the CALLING-PRP-Interworking layer saves the RVCI in the static table;
174. the CALLING-PRP-Interworking layer sets the in-use field in the static table to yes;
175. the CALLING-PRP-Interworking layer starts timer concerning reaction of the UTE;
176. if timer expiry before answer of UTE then set the in-use field to no;
177. the CALLING-PRP-Interworking layer feeds the buffered ATM (signalling) cells to the ATM-layer (where, among other things, the ATM cells are serialised into a bitstream);
178. CALLING-PRP-ATM layer delivers bitstream to the CALLING-PRP-PHY layer;

179. 'RELEASE COMPLETE message' is transported to the calling-UTE, using the optical channel;

III_Setup_c

1. MPS wants to setup a connection with another MPS;
2. calling UTE transmits an 'SETUP message' to the calling-PRP using the optical channel;

3. the CALLING-PRP-PHY layer delivers the received bitstream to the CALLING-PRP-ATM layer;
4. the CALLING-PRP-ATM layer deserialises the bitstream into ATM cells;
5. the CALLING-PRP-ATM layer delivers signalling ATM cell (if any) to the CALLING-PRP-Interworking layer;
6. the CALLING-PRP-Interworking layer buffers (in a receive buffer) the incoming signalling ATM cells (buffering is needed because the CALLING-PRP-MAC ,maybe , first has to content for a free (signalling) slot);
7. the CALLING-PRP-Interworking layer starts a timer to measure the time between two successive signalling cells;
8. on expiry of the timer, set the static table inuse value to false;
9. the CALLING-PRP-Interworking layer checks if the static table inuse value is true or false;
10. if inuse is false then
    - ask the MAC layer to content for an uplink (signalling) timeslot and start timer for contention time;
    - on expiry of timer delete buffered signalling ATM cell
    - the CALLING-PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the CALLING-PRP-Interworking layer stops the timer;
    - the CALLING-PRP-Interworking layer changes the static table field inuse to true;
11. if inuse is true then the CALLING-PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
12. the CALLING-PRP-Interworking layer stops the timer;
13. the CALLING-PRP-Interworking layer checks if there is another signalling ATM cell in the de-serialiser (or better, waits until the CALLING-PRP-ATM layer delivers a signalling cell before timer expiry);
14. if no signalling ATM cell available in de-serialiser then change static table field inuse to false (end);
15. if ATM cell available in de-serialiser then
    - the CALLING-PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the CALLING-PRP-Interworking layer continues this procedure until there is no signalling ATM cell in the de-serialiser;
16. the CALLING-PRP-MAC layer delivers the bitstream to the CALLING-PRP-PHY layer;

17. the 'SETUP message' is transported to the MSS over the radio channel;.

18. at the MSS the MSS-PHY layer delivers the constructed bitstream to the MSS-MAC layer;
19. the MSS-MAC layer delivers the SETUP message (segmented in ATM cells) to the MSS-ATM layer;
20. the MSS-ATM layer removes the ATM cell header and delivers the ATM information cell to the MSS-SAAL;
21. the MSS-SAAL reassembles (among other things) the SETUP message. The MSS-SAAL delivers the SETUP message to the MSS-Interworking layer;
22. the MSS-Interworking layer receives the incoming signalling (SETUP) message and buffers the message;
23. the MSS-Interworking layer extracts the Protocol discriminator Information Element:
    - must be 0000 1001 (Q.2931)
24. the MSS-Interworking layer extracts the Call reference Information Element:
    - flag must be 0(message sent from the side that originates the call reference);
    - call reference value (23 bit) saved in static table;
25. the MSS-Interworking layer extracts the Message type Information Element:
    - message type must be 0000 0101 (SETUP message);
    - change state of calling MEDIAN party from Null(U0) into Call initiated(U1) state in static table;
    - save internal in originated field in the static table;
26. the MSS-Interworking layer extracts the Message length Information Element:
    - extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself);

27. the MSS-Interworking layer extracts the ATM traffic descriptor Information Element:
    - deduce information about the forward/backward peak cell rate (CLP=0 and CLP=0+1) (if present);
    - deduce information about the forward/backward sustainable cell rate (CLP=0 and CLP=0+1) (if present);
    - deduce information about the forward/backward maximum burst size (CLP=0 and CLP=0+1) (if present);
    - save information in the static data base;
28. the MSS-Interworking layer extracts the Broadband bearer capability Information Element:
    - bearer class (BCOB-A, BCOB-C, BCOB-X(most likely to be used));
    - traffic type (CBR, VBR);
    - timing requirements (no indication, end-to-end timing required, end-to-end timing not required;
    - susceptibility to clipping (yes,no);
    - user plane connection (point-to-point, point-to-multipoint);
    - save information in the static data base;
29. the MSS-Interworking layer extracts the Called party Number Information Element and deduces the ATM end system address;
30. the MSS-Interworking layer extracts the QoS Information element, which can contain information concerning the:
    - cell error ratio;
    - severly-errored cell block ratio;
    - cell misinsertion ratio;
    - cell transfer delay;
    - mean cell transfer delay;
    - cell delay variation;
    - save information in the static data base;
31. the MSS-Interworking layer deduces if the end-user is:
    - an MEDIAN system end-user (is valid in this scenario);
    - not an MEDIAN system end-user;
32. the MSS-Interworking layer checks if message is errorless (Yes assumed);
33. the MSS-Interworking layer checks if requested QoS can be provided, using the extracted QoS information. There is additional information needed about the behaviour of the overall system for example BER, bursty characteristic of errors, propagation delay, transport delay of ATM cells in ATM layer, buffer capacity in Median Server Station etc. It should be possible to deduce the necessary information from the information maintained in the static data base;
34. the MSS-Interworking layer checks if able to support combination of traffic parameters in both the directions (MSS⇒ MPS and MPS⇒ MSS). The MSS-Interworking layer maintains the information of the static list, therefore having the data to derive the free slots in the uplink and the downlink;
35. the MSS-Interworking layer checks if a RVCI is available for the connection form the MSS to the calling party and if a RVCI is available for the connection form the MSS to the called party;
36. the previous checks are okay (because this is scenario III_Setup_c);
37. MSS-Interworking layer chooses a unused VPI/VCI combination. The VPI/VCI information is saved in the static list;
38. MSS-Interworking layer chooses a unused RVCI for the called MEDIAN party (saved in static list);
39. MSS-Interworking layer chooses a unused RVCI for the calling MEDIAN party (saved in static list);
40. MSS-Interworking layer: deduce time on which cells belonging to the selected RVCI (eq. VPCI/VCI) expire and retain this value in the static list (in row belonging to the calling and the called MEDIAN party). The needed values, forward and backward cell transfer delay and forward and backward mean cell transfer delay; can be found in the QoS parameter information element located in the SETUP message;
41. copy call reference value form called party row to calling party row;
42. copy VPCI/VCI value form called party row to calling party row;
43. state from called RVCI is changed from Null(U0) into Call Present (U6);

44. modification of the SETUP message:
    - generate Connection Identifier Information Element:
    - the MSS-Interworking layer inserts the chosen VPI/VCI (PVCI/RVCI) (on the same way the ATM switch would) in the default structure of the Connection Identifier Information Element;
    - this message is saved, because it also has to be added to the CALL PROCEEDING message which will be sent by the called UTE to the MSS (and which has to be sent by the MSS to the calling UTE);
45. MSS-Interworking layer starts timer T303;
46. IF no reaction to the SETUP message before expiry of timer T303 THEN initiate call/connection release;
47. signalling (SETUP) message is delivered to the MSS-SAAL layer;
48. after the MSS-SAAL layer has done the necessary actions and segmented the signalling message into ATM-information containers, it delivers the ATM-information containers to the MSS-ATM layer;
49. the MSS-ATM layer adds the ATM-header to the ATM-information container ;
50. the ATM cells and all the necessary information being of importance to the MSS-MAC are send to the MSS-MAC;
51. MSS-MAC delivers the bitstream to the MSS-PHY layer (MAC and PHY take care of the 'reliable' transportation of the SETUP message to the PRP);

52. 'SETUP message' is transported to the called-PRP using the radiochannel;

53. the CALLED-PRP-PHY layer delivers the constructed bitstream to the CALLED-PRP-MAC layer;
54. the CALLED-PRP-MAC delivers the broadcast cell to the CALLED-PRP-Interworking layer;
55. the CALLED-PRP-MAC delivers (if any) signalling cell(s) (VPI/VCI=0/5) to the Interworking layer;
56. the CALLED-PRP-Interworking layer buffers the broadcast cell;
57. the CALLED-PRP-Interworking layer buffers the signalling cell(s);
58. the CALLED-PRP-Interworking layer deduces signalling broadcast cell from the broadcast cell;
59. the CALLED-PRP-Interworking layer checks if signalling broadcast cell is nil;
60. if nil then delete broadcast cell and signalling broadcast cell, else continue (continue is assumed);
61. the CALLED-PRP-Interworking layer deduces the RVCI and the internal address (or ATM end system address or E.164 address) from the signalling broadcast cell;
62. the CALLED-PRP-Interworking layer checks if the internal address is the address of the CALLED-PRP (MPS)
    - if unequal then do not proceed and delete the buffered ATM signalling cells (if any) and the broadcast cell;
    - if yes then proceed (yes is assumed);
63. the CALLED-PRP-Interworking layer saves the RVCI in the static table;
64. the CALLED-PRP-Interworking layer sets the in-use field in the static table to yes;
65. the CALLED-PRP-Interworking layer starts timer concerning reaction of the UTE;
66. if timer expiry before answer of UTE then set the in-use field to no;
67. the CALLED-PRP-Interworking layer feeds the buffered ATM (signalling) cells to the ATM-layer (where, among other things, the ATM cells are serialised into a bitstream);
68. CALLED-PRP-ATM layer delivers bitstream to the CALLED-PRP-PHY layer;

69. 'SETUP message' is transported to the called-UTE using the optical channel;
70. the called-UTE decides to send a CALL PROCEEDING message to the calling UTE (through the MSS);
71. 'CALL PROCEEDING message' is transported to the called-PRP using the optical channel;

72. the PRP-PHY layer delivers the received bitstream to the PRP-ATM layer;
73. the PRP-ATM layer deserialises the bitstream into ATM cells;
74. the PRP-ATM layer delivers signalling ATM cell (if any) to the PRP-Interworking layer;
75. the PRP-Interworking layer buffers (in a receive buffer) the incoming signalling ATM cells (buffering is needed because the PRP-MAC ,maybe , first has to content for a free (signalling) slot);
76. the PRP-Interworking layer starts a timer to measure the time between two successive signalling cells;
77. on expiry of the timer, set the static table inuse value to false;
78. the PRP-Interworking layer checks if the static table inuse value is true or false;

79. if inuse is false then
    - ask the MAC layer to content for an uplink (signalling) timeslot and start timer for contention time;
    - on expiry of timer delete buffered signalling ATM cell
    - the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the PRP-Interworking layer stops the timer;
    - the PRP-Interworking layer changes the static table field inuse to true;
80. if inuse is true then the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
81. the PRP-Interworking layer stops the timer;
82. the PRP-Interworking layer checks if there is another signalling ATM cell in the de-serialiser (or better, waits until the PRP-ATM layer delivers a signalling cell before timer expiry);
83. if no signalling ATM cell available in de-serialiser then change static table field inuse to false (end);
84. if ATM cell available in de-serialiser then
    - the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the PRP-Interworking layer continues this procedure until there is no signalling ATM cell in the de-serialiser;
85. the PRP-MAC layer delivers the bitstream to the PRP-PHY layer;

86. the 'signalling (CALL PROCEEDING) message' is transported to the MSS over the radio channel;.

87. at the MSS the MSS-PHY layer delivers the constructed bitstream to the MSS-MAC layer;
88. the MSS-MAC layer delivers the 'signalling (CALL PROCEEDING) message' to the MSS-ATM layer;
89. the MSS-ATM layer removes the ATM cell header and delivers the ATM information cell to the MSS-SAAL;
90. the MSS-SAAL reassembles (among other things) the received signalling ATM information cell containers. The MSS-SAAL delivers the CALL PROCEEDING message to the MSS-Interworking layer;
91. the MSS-Interworking layer receives the incoming signalling (CALL PROCEEDING) message and buffers the message;
92. the MSS-Interworking layer extracts the Protocol discriminator Information Element:
    - must be 000010001 (Q.2931)
93. the MSS-Interworking layer extracts the Call reference Information Element:
    - flag must be 1 (message sent to side that originates the call reference (calling party));
94. the deduced call reference value is also saved in static table, and uniquely identifies the call the signalling message refers to;
95. the MSS-Interworking layer extracts the Message type Information Element:
    - message type must be 0000 0010 (CALL PROCEEDING message);
    - change state from calling party from Call Initiated(U1) into Incoming Call Proceeding(U9) state in static table;
    - change state from called party from Call Present(U6) into Outgoing Call Proceeding(U3)state in static table;
    - stop timer T303;
96. the MSS- Interworking layer adds the saved Connection identifier Information Element to the CALL PROCEEDING message;
97. the MSS-Interworking layer extracts the Message length Information Element;
98. the MSS-Interworking layer checks if message is errorless (Yes assumed);
99. the MSS-Interworking layer calculates the new length of the signalling message (old length + length (connection identifier Information element) and changes the Message length Information Element according to the calculated length;
100. the MSS-Interworking layer starts timer T310;
101. IF no reaction to the CALL PROCEEDING message before expiry of timer T310 THEN initiate call/connection release;
102. MSS-Interworking layer sends the changed signalling (CALL PROCEEDING) message to the MSS-SAAL (to the calling MEDIAN party);
103. after the MSS-SAAL layer has done the necessary actions and segmented the setup message into ATM-information containers, it delivers the ATM-information containers to the MSS-ATM layer;
104. the MSS-ATM layer adds the ATM-header to the ATM-information container;

105. the ATM cells and all the necessary information being of importance to the MSS-MAC are send to the MSS-MAC. There is a way of synchronisation needed between the MSS-MAC and the MSS-Interworking layer which can be accomplished by the use of the primitives: request and indication with there own message indication;
106. MSS-MAC delivers the bitstream to the MSS-PHY layer (MAC and PHY take care of the 'reliable' transportation of the CALL PROCEEDING message to the PRP);

107. CALL PROCEEDING message is transported to the calling-PRP using the radiochannel;

108. the CALLING-PRP-PHY layer delivers the constructed bitstream to the CALLING-PRP-MAC layer;
109. the CALLING-PRP-MAC delivers the broadcast cell to the CALLING-PRP-Interworking layer;
110. the CALLING-PRP-MAC delivers (if any) signalling cell(s) (VPI/VCI=0/5) to the Interworking layer;
111. the CALLING-PRP-Interworking layer buffers the broadcast cell;
112. the CALLING-PRP-Interworking layer buffers the signalling cell(s);
113. the CALLING-PRP-Interworking layer deduces signalling broadcast cell from the broadcast cell;
114. the CALLING-PRP-Interworking layer checks if signalling broadcast cell is nil;
115. if nil then delete broadcast cell and signalling broadcast cell, else continue (continue is assumed);
116. the CALLING-PRP-Interworking layer deduces the RVCI and the internal address (or ATM end system address or E.164 address) from the signalling broadcast cell;
117. the CALLING-PRP-Interworking layer checks if the internal address is the address of the CALLING-PRP (MPS)
    - if unequal then do not proceed and delete the buffered ATM signalling cells (if any) and the broadcast cell;
    - if yes then proceed (yes is assumed);
118. the CALLING-PRP-Interworking layer saves the RVCI in the static table;
119. the CALLING-PRP-Interworking layer sets the in-use field in the static table to yes;
120. the CALLING-PRP-Interworking layer starts timer concerning reaction of the UTE;
121. if timer expiry before answer of UTE then set the in-use field to no;
122. the CALLING-PRP-Interworking layer feeds the buffered ATM (signalling) cells to the ATM-layer (where, among other things, the ATM cells are serialised into a bitstream);
123. CALLING-PRP-ATM layer delivers bitstream to the CALLING-PRP-PHY layer;

124. 'CALL PROCEEDING message' is transported to the calling-UTE, using the optical channel;

125. the UTE wants to accept the call (Yes assumed because III_Setup_c), it sends a CONNECT message to the calling PRP;

126. the CALLED-PRP-PHY layer delivers the received bitstream to the CALLED-PRP-ATM layer;
127. the CALLED-PRP-ATM layer deserialises the bitstream into ATM cells;
128. the CALLED-PRP-ATM layer delivers signalling ATM cell (if any) to the CALLED-PRP-Interworking layer;
129. the CALLED-PRP-Interworking layer buffers (in a receive buffer) the incoming signalling ATM cells (buffering is needed because the CALLED-PRP-MAC ,maybe , first has to content for a free (signalling) slot);
130. the CALLED-PRP-Interworking layer starts a timer to measure the time between two successive signalling cells;
131. on expiry of the timer, set the static table inuse value to false;
132. the CALLED-PRP-Interworking layer checks if the static table inuse value is true or false;
133. if inuse is false then
    - ask the MAC layer to content for an uplink (signalling) timeslot and start timer for contention time;
    - on expiry of timer delete buffered signalling ATM cell
    - the CALLED-PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the CALLED-PRP-Interworking layer stops the timer;
    - the CALLED-PRP-Interworking layer changes the static table field inuse to true;
134. if inuse is true then the CALLED-PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
135. the CALLED-PRP-Interworking layer stops the timer;

136.the CALLED-PRP-Interworking layer checks if there is another signalling ATM cell in the de-serialiser (or better, waits until the CALLED-PRP-ATM layer delivers a signalling cell before timer expiry);
137.if no signalling ATM cell available in de-serialiser then change static table field inuse to false (end);
138.if ATM cell available in de-serialiser then
- the CALLED-PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
- the CALLED-PRP-Interworking layer continues this procedure until there is no signalling ATM cell in the de-serialiser;

139.the CALLED-PRP-MAC layer delivers the bitstream to the CALLED-PRP-PHY layer;

140.the 'CONNECT message' is transported to the MSS over the radio channel;

141.the MSS-PHY layer delivers the constructed bitstream to the MSS-MAC layer;
142.the MSS-MAC layer delivers the signalling (CONNECT message) to the MSS-ATM layer;
143.the MSS-ATM layer removes the ATM cell header and delivers the ATM information cell to the MSS-SAAL;
144.the MSS-SAAL reassembles (among other things) the received 'signalling message'. The SAAL delivers the signalling (CONNECT) message to the MSS-Interworking layer;
145.the MSS-Interworking layer receives the signalling message;
146.the MSS-Interworking layer receives the incoming signalling (CONNECT) message and buffers the message;
147.the MSS-Interworking layer extracts the Protocol discriminator Information Element:
- must be 000010001 (Q.2931)

148.the MSS-Interworking layer extracts the Call reference Information Element:
- flag must be 1 (message sent to side that originates the call reference (calling party));

149.the deduced call reference value is also saved in static table, and uniquely identifies the call the signalling message refers to;
150.the MSS-Interworking layer extracts the Message type Information Element:
- message type must be 0000 0111 (CONNECT message);
- change state of calling connection from Incoming Call Proceeding(U9) state into Active(U10) in static table;
- change state of called connection from Outgoing Call Proceeding(U3) state into Active(U10) in static table;
- stop T310;

151.the MSS-Interworking layer extracts the Message length Information Element;
152.extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself);
153.the MSS-Interworking layer checks if message is errorless (Yes assumed);
154.the MSS-Interworking layer starts timer T313;
155.IF no reaction to the CONNECT message before expiry of timer T313 THEN initiate call/connection release;
156.no modification of the CONNECT message, the MSS-Interworking layer delivers the signalling message to the MSS-SAAL;
157.after the MSS-SAAL layer has done the necessary actions and segmented the signalling (CONNECT) message into ATM-information containers, it delivers the ATM-information containers to the MSS-ATM layer;
158.the MSS-ATM layer adds the ATM-header to the ATM-information container;
159.the ATM cells and all the necessary information being of importance to the MSS-MAC are send to the MSS-MAC. There is a way of synchronisation needed between the MSS-MAC and the MSS-Interworking layer which can be accomplished by the use of the primitives: request and indication with there own message indication;
160.MSS-MAC delivers the bitstream to the MSS-PHY layer;

161.CONNECT message is transported to the calling-PRP using the radiochannel;

162.the CALLING-PRP-PHY layer delivers the constructed bitstream to the CALLING-PRP-MAC layer;
163.the CALLING-PRP-MAC delivers the broadcast cell to the CALLING-PRP-Interworking layer;
164.the CALLING-PRP-MAC delivers (if any) signalling cell(s) (VPI/VCI=0/5) to the Interworking layer;
165.the CALLING-PRP-Interworking layer buffers the broadcast cell;

166.the CALLING-PRP-Interworking layer buffers the signalling cell(s);
167.the CALLING-PRP-Interworking layer deduces signalling broadcast cell from the broadcast cell;
168.the CALLING-PRP-Interworking layer checks if signalling broadcast cell is nil;
169.if nil then delete broadcast cell and signalling broadcast cell, else continue (continue is assumed);
170.the CALLING-PRP-Interworking layer deduces the RVCI and the internal address (or ATM end system address or E.164 address) from the signalling broadcast cell;
171.the CALLING-PRP-Interworking layer checks if the internal address is the address of the CALLING-PRP (MPS)
- if unequal then do not proceed and delete the buffered ATM signalling cells (if any) and the broadcast cell;
- if yes then proceed (yes is assumed);

172.the CALLING-PRP-Interworking layer saves the RVCI in the static table;
173.the CALLING-PRP-Interworking layer sets the in-use field in the static table to yes;
174.the CALLING-PRP-Interworking layer starts timer concerning reaction of the UTE;
175.if timer expiry before answer of UTE then set the in-use field to no;
176.the CALLING-PRP-Interworking layer feeds the buffered ATM (signalling) cells to the ATM-layer (where, among other things, the ATM cells are serialised into a bitstream);
177.CALLING-PRP-ATM layer delivers bitstream to the CALLING-PRP-PHY layer;

178.the 'CONNECT' message is transported to the calling UTE;
179.the calling UTE reacts by sending a 'CONNECT ACKNOWLEDGE' message to the called UTE;

180.the CALLING-PRP-PHY layer delivers the received bitstream to the CALLING-PRP-ATM layer;
181.the CALLING-PRP-ATM layer deserialises the bitstream into ATM cells;
182.the CALLING-PRP-ATM layer delivers signalling ATM cell (if any) to the CALLING-PRP-Interworking layer;
183.the CALLING-PRP-Interworking layer buffers (in a receive buffer) the incoming signalling ATM cells (buffering is needed because the CALLING-PRP-MAC ,maybe , first has to content for a free (signalling) slot);
184.the CALLING-PRP-Interworking layer starts a timer to measure the time between two successive signalling cells;
185.on expiry of the timer, set the static table inuse value to false;
186.the CALLING-PRP-Interworking layer checks if the static table inuse value is true or false;
187.if inuse is false then
- ask the MAC layer to content for an uplink (signalling) timeslot and start timer for contention time;
- on expiry of timer delete buffered signalling ATM cell
- the CALLING-PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
- the CALLING-PRP-Interworking layer stops the timer;
- the CALLING-PRP-Interworking layer changes the static table field inuse to true;

188.if inuse is true then the CALLING-PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
189.the CALLING-PRP-Interworking layer stops the timer;
190.the CALLING-PRP-Interworking layer checks if there is another signalling ATM cell in the de-serialiser (or better, waits until the CALLING-PRP-ATM layer delivers a signalling cell before timer expiry);
191.if no signalling ATM cell available in de-serialiser then change static table field inuse to false (end);
192.if ATM cell available in de-serialiser then
- the CALLING-PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
- the CALLING-PRP-Interworking layer continues this procedure until there is no signalling ATM cell in the de-serialiser;

193.the CALLING-PRP-MAC layer delivers the bitstream to the CALLING-PRP-PHY layer;

194.at the MSS the MSS-PHY layer delivers the constructed bitstream to the MSS-MAC layer;
195.the MSS-MAC layer delivers the 'CONNECT ACKNOWLEDGE message' (segmented in ATM cells) to the MSS-ATM layer;
196.the MSS-ATM layer removes the ATM cell header and delivers the ATM information cell to the MSS-SAAL;
197.the MSS-SAAL reassembles (among other things) the 'CONNECT ACKNOWLEDGE message'. The MSS-SAAL delivers the CONNECT ACKNOWLEDGE message to the MSS-Interworking layer;

198. the MSS-Interworking layer receives the incoming signalling (CONNECT ACKNOWLEDGE) message and buffers the message;
199. the MSS-Interworking layer extracts the Protocol discriminator Information Element:
    - must be 000010001 (Q.2931)
200. the MSS-Interworking layer extracts the Call reference Information Element:
    - flag must be 0 (message sent from side that originates the call reference (calling party));
201. the deduced call reference value is also saved in static table, and identifies the call the signalling message refers to;
202. the MSS-Interworking layer extracts the Message type Information Element:
    - message type must be 0000 1111 (CONNECT ACKNOWLEDGE message);
    - stop timer T313;
203. the MSS-Interworking layer extracts the Message length Information Element;
204. the MSS-Interworking layer extracts the length of the message (excluding protocol discriminator, call reference, message type, message length indication itself);
205. the MSS-Interworking layer checks if message is errorless (Yes assumed);
206. no modification of the CONNECT ACKNOWLEDGE message, the MSS seems to be transparent to the CONNECT ACKNOWLEDGE message in this scenario;
207. CONNECT ACKNOWLEDGE message is delivered to the MSS-SAAL layer (it is possible to directly sent the buffered CONNECT ACKNOWLEDGE message (in received buffer) to the SAAL because no modification has taken place);
208. after the MSS-SAAL layer has done the necessary actions and segmented the setup message into ATM-information containers. The flow of ATM information containers is delivered to the MSS-ATM layer;
209. the MSS-ATM layer adds the ATM-header to the ATM-information container resulting in the same ATM cell corresponding to the arrived ATM-cell at the beginning;
210. the ATM cells and all the necessary information being of importance to the MSS-MAC are send to the MSS-MAC;
211. MSS-MAC delivers the bitstream to the MSS-PHY layer;

212. CONNECT ACKNOWLEDGE message is transported to the called-PRP using the radiochannel;

213. the CALLED-PRP-PHY layer delivers the constructed bitstream to the CALLED-PRP-MAC layer;
214. the CALLED-PRP-MAC delivers the broadcast cell to the CALLED-PRP-Interworking layer;
215. the CALLED-PRP-MAC delivers (if any) signalling cell(s) (VPI/VCI=0/5) to the Interworking layer;
216. the CALLED-PRP-Interworking layer buffers the broadcast cell;
217. the CALLED-PRP-Interworking layer buffers the signalling cell(s);
218. the CALLED-PRP-Interworking layer deduces signalling broadcast cell from the broadcast cell;
219. the CALLED-PRP-Interworking layer checks if signalling broadcast cell is nil;
220. if nil then delete broadcast cell and signalling broadcast cell, else continue (continue is assumed);
221. the CALLED-PRP-Interworking layer deduces the RVCI and the internal address (or ATM end system address or E.164 address) from the signalling broadcast cell;
222. the CALLED-PRP-Interworking layer checks if the internal address is the address of the CALLED-PRP (MPS)
    - if unequal then do not proceed and delete the buffered ATM signalling cells (if any) and the broadcast cell;
    - if yes then proceed (yes is assumed);
223. the CALLED-PRP-Interworking layer saves the RVCI in the static table;
224. the CALLED-PRP-Interworking layer sets the in-use field in the static table to yes;
225. the CALLED-PRP-Interworking layer starts timer concerning reaction of the UTE;
226. if timer expiry before answer of UTE then set the in-use field to no;
227. the CALLED-PRP-Interworking layer feeds the buffered ATM (signalling) cells to the ATM-layer (where, among other things, the ATM cells are serialised into a bitstream);
228. CALLED-PRP-ATM layer delivers bitstream to the CALLED-PRP-PHY layer;

229. CONNECT ACKNOWLEDGE message is transported to the called-UTE, using the optical channel;
230. the CONNECT ACKNOWLEDGE message arrives at the destination, MPS
231. connection is established;

IV_Release_a

1. the ATM end-user initiates call release by sending a RELEASE message to the ATM switch;
2. on receiptance of the RELEASE message, the ATM switch sends a RELEASE COMPLETE to the releasing ATM end-user and a RELEASE message to the MPS;

3. 'RELEASE message' is delivered to the MSS;

4. MSS-PHY layer receives information from switch, and delivers the bit stream to the MSS-ATM layer;
5. MSS-ATM layer delivers the signalling ATM cells without the header to the MSS-SAAL;
6. MSS-SAAL reassembles the signalling ATM cells and delivers the signalling message (in particular RELEASE message) to the MSS-Interworking layer;
7. the MSS-Interworking layer receives the incoming signalling (RELEASE) message and buffers the message;
8. the MSS-Interworking layer extracts the Protocol discriminator Information Element:
   - must be 000010001 (Q.2931)
9. the MSS-Interworking layer extracts the Call reference Information Element:
   - flag: - 0 = message sent from the side that originates the call reference;
     1 = message sent to the side that originates the call reference;
     the MSS-Interworking layer knows who asked to establish the connection;
   - call reference value: available in the static table, uniquely identifies the call/connection the signalling message refers to;;
10. the MSS-Interworking layer extracts the Message type Information Element:
    - message type must be 0101 1101 (RELEASE message);
    - change state of connection from Active(U10) into Release Indication(U12) state in static table (row indicated by the call reference value);
11. the MSS-Interworking layer extracts the Message length Information Element:
    - extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself);
12. the MSS-Interworking layer extracts the Cause Information Element and deduces statistical data for the static database, information maintained in the static database;
13. the MSS-Interworking layer checks if message is errorless (Yes assumed);
14. MSS-Interworking layer starts timer T308;
15. IF no reaction to the RELEASE message before expiry of timer T308 THEN initiate release all the reservations made concerning the call reference value;
16. MSS-Interworking layer locally releases all the reservation made for this particular call, only the connection identifier and the state remains in the static table;
17. no modification of the signalling (RELEASE) message, the Median Server Station seems to be transparent to the RELEASE message in this scenario;
18. RELEASE message is delivered to the MSS-SAAL layer (it is possible to directly sent the buffered RELEASE message (in received buffer) to the SAAL because no modification has taken place);
19. after the MSS-SAAL layer has done the necessary actions and segmented the signalling message into ATM-information containers, it delivers the ATM-information containers to the MSS-ATM layer;
20. the MSS-ATM layer adds the ATM-header to the ATM-information container resulting in the same ATM cell corresponding to the arrived ATM-cell at the beginning (meaning that the RELEASE message is transparently transported through the MSS-Interworking layer);
21. the ATM cells and all the necessary information being of importance to the MSS-MAC are send to the MSS-MAC;
22. MSS-MAC delivers the bitstream to the MSS-PHY layer (MAC and PHY take care of the 'reliable' transportation of the RELEASE message to the PRP);

23. 'RELEASE message' is transported to the PRP using the radiochannel;

24. the PRP-PHY layer delivers the constructed bitstream to the PRP-MAC layer;
25. the PRP-MAC delivers the broadcast cell to the PRP-Interworking layer;
26. the PRP-MAC delivers (if any) signalling cell(s) (VPI/VCI=0/5) to the Interworking layer;
27. the PRP-Interworking layer buffers the broadcast cell;

28. the PRP-Interworking layer buffers the signalling cell(s);
29. the PRP-Interworking layer deduces signalling broadcast cell from the broadcast cell;
30. the PRP-Interworking layer checks if signalling broadcast cell is nil;
31. if nil then delete broadcast cell and signalling broadcast cell, else continue (continue is assumed);
32. the PRP-Interworking layer deduces the RVCI and the internal address (or ATM end system address or E.164 address) from the signalling broadcast cell;
33. the PRP-Interworking layer checks if the internal address is the address of the PRP (MPS)
    - if unequal then do not proceed and delete the buffered ATM signalling cells (if any) and the broadcast cell;
    - if yes then proceed (yes is assumed);
34. the PRP-Interworking layer saves the RVCI in the static table;
35. the PRP-Interworking layer sets the in-use field in the static table to yes;
36. the PRP-Interworking layer starts timer concerning reaction of the UTE;
37. if timer expiry before answer of UTE then set the in-use field to no;
38. the PRP-Interworking layer feeds the buffered ATM (signalling) cells to the ATM-layer (where, among other things, the ATM cells are serialised into a bitstream);
39. PRP-ATM layer delivers bitstream to the PRP-PHY layer;

40. RELEASE message is transported to the UTE-PHY layer, using the optical channel;
41. the received bitstream arrives at the UTE side;

42. how the UTE exactly reacts to this 'message' is not of importance (because these functionality's are already implemented on the interface card mounted in the UTE);
43. the UTE decides to send a RELEASE COMPLETE message to the PRP[1] (which finally has to arrive at the ATM switch) when the requested call establishment has been initiated and no more call establishment information will be accepted;

44. 'RELEASE COMPLETE message' is transported to the PRP using the optical channel;

45. the PRP-PHY layer delivers the received bitstream to the PRP-ATM layer;
46. the PRP-ATM layer deserialises the bitstream into ATM cells;
47. the PRP-ATM layer delivers signalling ATM cell (if any) to the PRP-Interworking layer;
48. the PRP-Interworking layer buffers (in a receive buffer) the incoming signalling ATM cells (buffering is needed because the PRP-MAC ,maybe , first has to content for a free (signalling) slot);
49. the PRP-Interworking layer starts a timer to measure the time between two successive signalling cells;
50. on expiry of the timer, set the static table inuse value to false;
51. the PRP-Interworking layer checks if the static table inuse value is true or false;
52. if inuse is false then
    - ask the MAC layer to content for an uplink (signalling) timeslot and start timer for contention time;
    - on expiry of timer delete buffered signalling ATM cell
    - the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the PRP-Interworking layer stops the timer;
    - the PRP-Interworking layer changes the static table field inuse to true;
53. if inuse is true then the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
54. the PRP-Interworking layer stops the timer;
55. the PRP-Interworking layer checks if there is another signalling ATM cell in the de-serialiser (or better, waits until the PRP-ATM layer delivers a signalling cell before timer expiry);
56. if no signalling ATM cell available in de-serialiser then change static table field inuse to false (end);
57. if ATM cell available in de-serialiser then
    - the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the PRP-Interworking layer continues this procedure until there is no signalling ATM cell in the de-serialiser;
58. the PRP-MAC layer delivers the bitstream to the PRP-PHY layer;

59. the 'RELEASE COMPLETE message' is transported to the MSS over the radio channel;.

60. at the MSS the MSS-PHY layer delivers the constructed bitstream to the MSS-MAC layer;
61. the MSS-MAC layer delivers the 'signalling (RELEASE COMPLETE) message' to the MSS-ATM layer;

62. the MSS-ATM layer removes the ATM cell header and delivers the ATM information cell to the MSS-SAAL;
63. the MSS-SAAL reassembles (among other things) the signalling message. The MSS-SAAL delivers the signalling message to the MSS-Interworking layer;
64. the MSS-Interworking layer receives the incoming signalling (RELEASE COMPLETE) message and buffers the message;
65. the MSS-Interworking layer extracts the Protocol discriminator Information Element:
    - must be 000010001 (Q.2931)
66. the MSS-Interworking layer extracts the Call reference Information Element:
    - flag: - 0 = message sent from the side that originates the call reference;
      1 = message sent to the side that originates the call reference;
      the MSS-Interworking layer knows who asked to establish the connection;
    - call reference value: available in the static table, uniquely identifies the call/connection the signalling message refers to;
67. the MSS-Interworking layer extracts the Message type Information Element:
    - message type must be 0101 1010 (RELEASE COMPLETE message);
    - change state of connection from Release Indication(U12) into Null(U0) state in static table;
    - stop timer T308;
68. the MSS-Interworking layer extracts the Message length Information Element:
69. extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself);
70. the MSS-Interworking layer checks if message is errorless (Yes assumed);
71. the MSS-Interworking layer deduces information needed to construct the static data base (optional);
72. the MSS-Interworking layer locally releases the call reference value;
73. MSS-Interworking layer does not change the RELEASE COMPLETE message (only reading), and the RELEASE COMPLETE message is delivered to the MSS-SAAL;
74. the MSS-SAAL delivers the segmented RELEASE COMPLETE message to the MSS-ATM layer;
75. the MSS-ATM layer delivers the bitstream to the MSS-PHY layer (actions concerning RELEASE COMPLETE message within the MEDIAN system are finished);
76. the RELEASE COMPLETE message is transported to the ATM switch over the optical channel;
77. call/connection is released within the MEDIAN system;

IV_Release_b

1. the MPS decides to send a RELEASE message to the ATM switch;

2. 'RELEASE message' is transported to the PRP using the optical channel;

3. the PRP-PHY layer delivers the received bitstream to the PRP-ATM layer;
4. the PRP-ATM layer deserialises the bitstream into ATM cells;
5. the PRP-ATM layer delivers signalling ATM cell (if any) to the PRP-Interworking layer;
6. the PRP-Interworking layer buffers (in a receive buffer) the incoming signalling ATM cells (buffering is needed because the PRP-MAC ,maybe , first has to content for a free (signalling) slot);
7. the PRP-Interworking layer starts a timer to measure the time between two successive signalling cells;
8. on expiry of the timer, set the static table inuse value to false;
9. the PRP-Interworking layer checks if the static table inuse value is true or false;
10. if inuse is false then
    - ask the MAC layer to content for an uplink (signalling) timeslot and start timer for contention time;
    - on expiry of timer delete buffered signalling ATM cell
    - the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the PRP-Interworking layer stops the timer;
    - the PRP-Interworking layer changes the static table field inuse to true;
11. if inuse is true then the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
12. the PRP-Interworking layer stops the timer;
13. the PRP-Interworking layer checks if there is another signalling ATM cell in the de-serialiser (or better, waits until the PRP-ATM layer delivers a signalling cell before timer expiry);
14. if no signalling ATM cell available in de-serialiser then change static table field inuse to false (end);
15. if ATM cell available in de-serialiser then
    - the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the PRP-Interworking layer continues this procedure until there is no signalling ATM cell in the de-serialiser;
16. the PRP-MAC layer delivers the bitstream to the PRP-PHY layer;

17. the 'RELEASE message' is transported to the MSS over the radio channel;.

18. at the MSS the MSS-PHY layer delivers the constructed bitstream to the MSS-MAC layer;
19. the MSS-MAC layer delivers the 'RELEASE message' (segmented in ATM cells plus additional overhead) to the MSS-ATM layer;
20. the MSS-ATM layer removes the ATM cell header and delivers the ATM information cell to the MSS-SAAL;
21. the MSS-SAAL reassembles (among other things) the received signalling ATM information cell containers. The MSS-SAAL delivers the RELEASE message to the MSS-Interworking layer;
22. the MSS-Interworking layer receives the incoming signalling (RELEASE) message and buffers the message;
23. the MSS-Interworking layer extracts the Protocol discriminator Information Element:
    - must be 000010001 (Q.2931)
24. the MSS-Interworking layer extracts the Call reference Information Element:
    - flag: - 0 = message sent from the side that originates the call reference;
1 = message sent to the side that originates the call reference;
the MSS-Interworking layer knows who asked to establish the connection;
    - call reference value: available in the static table, uniquely identifies the call/connection the signalling message refers to;
25. the MSS-Interworking layer extracts the Message type Information Element:
    - message type must be 0101 1101 (RELEASE message);
    - change state of connection from Active(U10) into Release Request (U11) state in static table;
26. the MSS-Interworking layer extracts the Message length Information Element:
    - extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself);

27. the MSS-Interworking layer extracts the Cause Information Element and deduces statistical data for the static database, information maintained in the static database;
28. the MSS-Interworking layer checks if message is errorless (Yes assumed);
29. MSS-Interworking layer locally releases the reservation made for this particular call, only the call reference value and the state remains in the static table;
30. no modification of the signalling (RELEASE) message, the Median Server Station seems to be transparent to the RELEASE message in this scenario;
31. MSS-Interworking layer starts timer T308;
32. IF no reaction to the RELEASE message before expiry of timer T308 THEN initiate release all the reservations made concerning the call reference value;
33. no modification of the signalling (RELEASE) message, the Median Server Station seems to be transparent to the RELEASE message in this scenario;
34. RELEASE message is delivered to the MSS-SAAL layer (it is possible to directly sent the buffered RELEASE message (in received buffer) to the SAAL because no modification has taken place);
35. the MSS-SAAL delivers the segmented RELEASE message to the MSS-ATM layer (synchronised);
36. the MSS-ATM layer delivers the bitstream to the MSS-PHY layer

37. ATM-switch receives the RELEASE message and decides to send a RELEASE COMPLETE message to the MPS (the ATM switch is also involved in several other actions, which are not of our concern)(only information flows within the MEDIAN system are addressed);

38. MSS-PHY layer receives information from the ATM-switch, and delivers the bit stream to the MSS-ATM layer;
39. MSS-ATM layer delivers the signalling ATM cells without the header to the MSS-SAAL;
40. MSS-SAAL reassembles the signalling ATM cells and delivers the signalling message (in particular RELEASE COMPLETE message) to the MSS-Interworking layer;
41. the MSS-Interworking layer receives the incoming signalling (RELEASE COMPLETE) message and buffers the message;
42. the MSS-Interworking layer extracts the Protocol discriminator Information Element:
    - must be 000010001 (Q.2931)
43. the MSS-Interworking layer extracts the Call reference Information Element:
    - flag: - 0 = message sent from the side that originates the call reference;
      1 = message sent to the side that originates the call reference;
      the MSS-Interworking layer knows who asked to establish the connection;
    - call reference value: available in the static table, uniquely identifies the call/connection the signalling message refers to;
44. the MSS-Interworking layer extracts the Message type Information Element:
    - message type must be 0101 1010 (RELEASE COMPLETE message);
    - change state of connection from Release Request (U11) into Null(U0) state in static table;
    - stop timer T308
45. the MSS-Interworking layer extracts the Message length Information Element:
46. extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself);
47. the MSS-Interworking layer checks if message is errorless (Yes assumed);
48. the MSS-Interworking layer deduces information needed to construct the static data base (optional);
49. the MSS-Interworking layer locally releases the call reference value;
50. MSS-Interworking layer does not change the RELEASE COMPLETE message (only reading), and the RELEASE COMPLETE message is delivered to the MSS-SAAL;
51. after the MSS-SAAL layer has done the necessary actions and segmented the setup message into ATM-information containers. The flow of ATM information containers is delivered to the MSS-ATM layer;
52. the MSS-ATM layer adds the ATM-header to the ATM-information container resulting in the same ATM cell corresponding to the arrived ATM-cell at the beginning;
53. the ATM cells and all the necessary information being of importance to the MSS-MAC are send to the MSS-MAC;
54. MSS-MAC delivers the bitstream to the MSS-PHY layer;

55. RELEASE COMPLETE message is transported to the PRP using the radiochannel;

56. the PRP-PHY layer delivers the constructed bitstream to the PRP-MAC layer;
57. the PRP-MAC delivers the broadcast cell to the PRP-Interworking layer;
58. the PRP-MAC delivers (if any) signalling cell(s) (VPI/VCI=0/5) to the Interworking layer;
59. the PRP-Interworking layer buffers the broadcast cell;
60. the PRP-Interworking layer buffers the signalling cell(s);
61. the PRP-Interworking layer deduces signalling broadcast cell from the broadcast cell;
62. the PRP-Interworking layer checks if signalling broadcast cell is nil;
63. if nil then delete broadcast cell and signalling broadcast cell, else continue (continue is assumed);
64. the PRP-Interworking layer deduces the RVCI and the internal address (or ATM end system address or E.164 address) from the signalling broadcast cell;
65. the PRP-Interworking layer checks if the internal address is the address of the PRP (MPS)
    - if unequal then do not proceed and delete the buffered ATM signalling cells (if any) and the broadcast cell;
    - if yes then proceed (yes is assumed);
66. the PRP-Interworking layer saves the RVCI in the static table;
67. the PRP-Interworking layer sets the in-use field in the static table to yes;
68. the PRP-Interworking layer starts timer concerning reaction of the UTE;
69. if timer expiry before answer of UTE then set the in-use field to no;
70. the PRP-Interworking layer feeds the buffered ATM (signalling) cells to the ATM-layer (where, among other things, the ATM cells are serialised into a bitstream);
71. PRP-ATM layer delivers bitstream to the PRP-PHY layer;

72. RELEASE COMPLETE message is transported to the UTE, using the optical channel;

73. the RELEASE COMPLETE message arrives at the destination, UTE
74. connection is released and the connection identifier can be re-used;

IV_Release_c

1. ATM-switch decides to release the connection between the ATM end-use and the MPS;
2. ATM-switch initiate call release towards the ATM end-use and the UTE;

3. only in the UTE direction is considered;
4. ATM-switch sends 'RELEASE message' to the MPS;
5. 'RELEASE message' is delivered to the MSS;

6. MSS-PHY layer receives information from switch, and delivers the bit stream to the MSS-ATM layer;
7. MSS-ATM layer delivers the signalling ATM cells without the header to the MSS-SAAL;
8. MSS-SAAL reassembles the signalling ATM cells and delivers the signalling message (in particular RELEASE message) to the MSS-Interworking layer;
9. the MSS-Interworking layer receives the incoming signalling (RELEASE) message and buffers the message;
10. the MSS-Interworking layer extracts the Protocol discriminator Information Element:
    - must be 000010001 (Q.2931)
11. the MSS-Interworking layer extracts the Call reference Information Element:
    - flag: - 0 = message sent from the side that originates the call reference;
      1 = message sent to the side that originates the call reference;
      the MSS-Interworking layer knows who asked to establish the connection;
    - call reference value: available in the static table, uniquely identifies the call/connection the signalling message refers to;;
12. the MSS-Interworking layer extracts the Message type Information Element:
    - message type must be 0101 1101 (RELEASE message);
    - change state of connection from Active(U10) into Release Indication(U12) state in static table (row indicated by the call reference value);
13. the MSS-Interworking layer extracts the Message length Information Element:
    - extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself);
14. the MSS-Interworking layer extracts the Cause Information Element and deduces statistical data for the static database, information maintained in the static database;
15. the MSS-Interworking layer checks if message is errorless (Yes assumed);
16. MSS-Interworking layer starts timer T308;
17. IF no reaction to the RELEASE message before expiry of timer T308 THEN release all the reservations made concerning the call reference value;
18. MSS-Interworking layer locally releases all the reservation made for this particular call, only the call reference value and the state remains in the static table;
19. no modification of the signalling (RELEASE) message, the Median Server Station seems to be transparent to the RELEASE message in this scenario;
20. RELEASE message is delivered to the MSS-SAAL layer (it is possible to directly sent the buffered RELEASE message (in received buffer) to the SAAL because no modification has taken place);
21. after the MSS-SAAL layer has done the necessary actions and segmented the signalling message into ATM-information containers, it delivers the ATM-information containers to the MSS-ATM layer;
22. the MSS-ATM layer adds the ATM-header to the ATM-information container resulting in the same ATM cell corresponding to the arrived ATM-cell at the beginning (meaning that the RELEASE message is transparently transported through the MSS-Interworking layer);
23. the ATM cells and all the necessary information being of importance to the MSS-MAC are send to the MSS-MAC;
24. MSS-MAC delivers the bitstream to the MSS-PHY layer (MAC and PHY take care of the 'reliable' transportation of the RELEASE message to the PRP);

25. 'RELEASE message' is transported to the PRP using the radiochannel;

26. the PRP-PHY layer delivers the constructed bitstream to the PRP-MAC layer;
27. the PRP-MAC delivers the signalling cell (if any) in the broadcast cell to the PRP-Interworking layer;
28. the PRP-MAC delivers the signalling information cell(s) to the PRP-Interworking layer;

29. the PRP-Interworking layer deduces the RVCI and the internal address (or ATM end system address or E.164 address) from the signalling cell in the broadcast cell;
30. the PRP-Interworking layer checks if the internal address is the address of the PRP(UTE)
    - if not then do not proceed;
    - if yes then proceed (yes is assumed);
31. the PRP-Interworking layer saves the RVCI in the static table;
32. the PRP-Interworking layer buffers the signalling information cell(s) (of the different timeslot(s)) (s) belonging to the SRVCI;
33. the PRP-Interworking layer starts a timer for releasing the reservation of the SRVCI;
34. the PRP-Interworking layer feeds the buffered ATM (signalling) cells to the ATM-layer (where, among other things, the ATM cells are serialised into a bitstream);
35. PRP-ATM layer delivers bitstream to the PRP-PHY layer;

36. RELEASE message is transported to the UTE, using the optical channel;
37. the received bitstream arrives at the UTE side;

38. how the UTE exactly reacts to this 'message' is not of importance (because these functionality's are already implemented on the interface card mounted in the UTE);
39. the UTE decides to send a RELEASE COMPLETE message to the PRP (which finally has to arrive at the ATM switch) when the requested call establishment has been initiated and no more call establishment information will be accepted;

40. 'RELEASE COMPLETE message' is transported to the PRP using the optical channel;

41. the PRP-PHY layer delivers the received bitstream to the PRP-ATM layer;
42. the PRP-ATM layer deserialises the bitstream into ATM cells;
43. the PRP-ATM layer delivers signalling ATM cell (if any) to the PRP-Interworking layer;
44. the PRP-Interworking layer buffers (in a receive buffer) the incoming signalling ATM cells (buffering is needed because the PRP-MAC ,maybe , first has to content for a free (signalling) slot);
45. the PRP-Interworking layer starts a timer to measure the time between two successive signalling cells;
46. on expiry of the timer, set the static table inuse value to false;
47. the PRP-Interworking layer checks if the static table inuse value is true or false;
48. if inuse is false then
    - ask the MAC layer to content for an uplink (signalling) timeslot and start timer for contention time;
    - on expiry of timer delete buffered signalling ATM cell
    - the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the PRP-Interworking layer stops the timer;
    - the PRP-Interworking layer changes the static table field inuse to true;
49. if inuse is true then the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
50. the PRP-Interworking layer stops the timer;
51. the PRP-Interworking layer checks if there is another signalling ATM cell in the de-serialiser (or better, waits until the PRP-ATM layer delivers a signalling cell before timer expiry);
52. if no signalling ATM cell available in de-serialiser then change static table field inuse to false (end);
53. if ATM cell available in de-serialiser then
    - the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the PRP-Interworking layer continues this procedure until there is no signalling ATM cell in the de-serialiser;
54. the PRP-MAC layer delivers the bitstream to the PRP-PHY layer;

55. the 'RELEASE COMPLETE message' is transported to the MSS over the radio channel;.

56. at the MSS the MSS-PHY layer delivers the constructed bitstream to the MSS-MAC layer;
57. the MSS-MAC layer delivers the 'signalling (RELEASE COMPLETE) message' to the MSS-ATM layer;
58. the MSS-ATM layer removes the ATM cell header and delivers the ATM information cell to the MSS-SAAL;
59. the MSS-SAAL reassembles (among other things) the signalling message. The MSS-SAAL delivers the signalling message to the MSS-Interworking layer;

60. the MSS-Interworking layer receives the incoming signalling (RELEASE COMPLETE) message and buffers the message;
61. the MSS-Interworking layer extracts the Protocol discriminator Information Element:
    - must be 000010001 (Q.2931)
62. the MSS-Interworking layer extracts the Call reference Information Element:
    - flag: - 0 = message sent from the side that originates the call reference;
      1 = message sent to the side that originates the call reference;
      the MSS-Interworking layer knows who asked to establish the connection;
    - call reference value: available in the static table, uniquely identifies the call/connection the signalling message refers to;
63. the MSS-Interworking layer extracts the Message type Information Element:
    - message type must be 0101 1010 (RELEASE COMPLETE message);
    - change state of connection from Release Indication(U12) into Null(U0) state in static table;
    - stop timer T308;
64. the MSS-Interworking layer extracts the Message length Information Element:
65. extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself);
66. the MSS-Interworking layer checks if message is errorless (Yes assumed);
67. the MSS-Interworking layer deduces information needed to construct the static data base (optional);
68. the MSS-Interworking layer locally releases the call reference value;
69. MSS-Interworking layer does not change the RELEASE COMPLETE message (only reading), and the RELEASE COMPLETE message is delivered to the MSS-SAAL;
70. the MSS-SAAL delivers the segmented RELEASE COMPLETE message to the MSS-ATM layer;
71. the MSS-ATM layer delivers the bitstream to the MSS-PHY layer (actions concerning RELEASE COMPLETE message within the MEDIAN system are finished);
72. the RELEASE COMPLETE message is transported to the ATM switch over the optical channel;

73. call/connection is released within the MEDIAN system;

IV_Release_d

1. MSS decides to release the connection between the ATM end-use and the UTE;

2. MSS initiate call release towards the ATM end-use and the UTE;
3. the call/connection is identified by the call reference value, which is saved in the row of the static table corresponding to the call/connection which the MSS wants to release;
4. the MSS checks in the static table who initiated the call/connection establishment, needed to set the flag in the Call reference Information Element;

5. in the MSS towards the UTE direction:;
6. the MSS-Interworking layer generates the Protocol discriminator Information Element:
   - must be 000010001 (Q.2931)
7. the MSS-Interworking layer generates the Call reference Information Element:
   - flag: 0 = if ATM end-user originated the call reference;
1 = if UTE originated the call reference;
the MSS-Interworking layer knows who initiated the call/connection establishment;
   - call reference value: available in the static table, uniquely identifies the call/connection; call reference value (23 bit) of the connection which the MSS wants to release (is saved in static table);
8. the MSS-Interworking layer generates the Message type Information Element:
   - message type must be 0100 1101 (RELEASE message);
   - change state of connection from Active(U10) into Release Indication(U12) state in static table;
   - disconnect the virtual channel;
9. the MSS-Interworking layer generates the Cause Information Element:
   - Cause values are:
     - #16 ⇒ normal call clearing;
     - #41 ⇒ temporary failure;
     - #111 ⇒ protocol error, unspecified;
10. the MSS-Interworking layer calculates the message length:
    - extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself);
11. the MSS-Interworking layer generates the Message length Information Element:
12. the MSS-Interworking layer constructs the RELEASE message, and saves the RELEASE message (also needed in opposite direction) ;
13. the MSS-Interworking layer starts timer T308;
14. IF no reaction to the RELEASE message before expiry of timer T308 THEN release all the reservations made concerning the call reference value;
15. RELEASE message is delivered to the MSS-SAAL layer;
16. after the MSS-SAAL layer has done the necessary actions and segmented the setup message into ATM-information containers, it delivers the ATM-information containers to the MSS-ATM layer;
17. the MSS-ATM layer adds the ATM-header to the ATM-information container resulting in the same ATM cell corresponding to the arrived ATM-cell at the beginning;
18. the ATM cells and all the necessary information being of importance to the MSS-MAC are send to the MSS-MAC;
19. MSS-MAC delivers the bitstream to the MSS-PHY layer;

20. 'RELEASE message' is transported to the PRP using the radiochannel;

21. the PRP-PHY layer delivers the constructed bitstream to the PRP-MAC layer;
22. the PRP-MAC delivers the broadcast cell to the PRP-Interworking layer;
23. the PRP-MAC delivers (if any) signalling cell(s) (VPI/VCI=0/5) to the Interworking layer;
24. the PRP-Interworking layer buffers the broadcast cell;
25. the PRP-Interworking layer buffers the signalling cell(s);
26. the PRP-Interworking layer deduces signalling broadcast cell from the broadcast cell;
27. the PRP-Interworking layer checks if signalling broadcast cell is nil;
28. if nil then delete broadcast cell and signalling broadcast cell, else continue (continue is assumed);

29. the PRP-Interworking layer deduces the RVCI and the internal address (or ATM end system address or E.164 address) from the signalling broadcast cell;
30. the PRP-Interworking layer checks if the internal address is the address of the PRP (MPS)
    - if unequal then do not proceed and delete the buffered ATM signalling cells (if any) and the broadcast cell;
    - if yes then proceed (yes is assumed);
31. the PRP-Interworking layer saves the RVCI in the static table;
32. the PRP-Interworking layer sets the in-use field in the static table to yes;
33. the PRP-Interworking layer starts timer concerning reaction of the UTE;
34. if timer expiry before answer of UTE then set the in-use field to no;
35. the PRP-Interworking layer feeds the buffered ATM (signalling) cells to the ATM-layer (where, among other things, the ATM cells are serialised into a bitstream);
36. PRP-ATM layer delivers bitstream to the PRP-PHY layer;

37. 'RELEASE message' is transported to the UTE using the optical channel;

38. how the UTE exactly reacts to this 'message' is not of importance (because these functionality's are already implemented on the interface card mounted in the UTE);
39. the UTE decides to send a RELEASE COMPLETE message to the PRP (which finally has to arrive at the ATM switch) ;

40. 'RELEASE COMPLETE message' is transported to the PRP using the optical channel;

41. the PRP-PHY layer delivers the received bitstream to the PRP-ATM layer;
42. the PRP-ATM layer deserialises the bitstream into ATM cells;
43. the PRP-ATM layer delivers signalling ATM cell (if any) to the PRP-Interworking layer;
44. the PRP-Interworking layer buffers (in a receive buffer) the incoming signalling ATM cells (buffering is needed because the PRP-MAC ,maybe , first has to content for a free (signalling) slot);
45. the PRP-Interworking layer starts a timer to measure the time between two successive signalling cells;
46. on expiry of the timer, set the static table inuse value to false;
47. the PRP-Interworking layer checks if the static table inuse value is true or false;
48. if inuse is false then
    - ask the MAC layer to content for an uplink (signalling) timeslot and start timer for contention time;
    - on expiry of timer delete buffered signalling ATM cell
    - the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the PRP-Interworking layer stops the timer;
    - the PRP-Interworking layer changes the static table field inuse to true;
49. if inuse is true then the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
50. the PRP-Interworking layer stops the timer;
51. the PRP-Interworking layer checks if there is another signalling ATM cell in the de-serialiser (or better, waits until the PRP-ATM layer delivers a signalling cell before timer expiry);
52. if no signalling ATM cell available in de-serialiser then change static table field inuse to false (end);
53. if ATM cell available in de-serialiser then
    - the PRP-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the PRP-Interworking layer continues this procedure until there is no signalling ATM cell in the de-serialiser;
54. the PRP-MAC layer delivers the bitstream to the PRP-PHY layer;

55. the 'RELEASE COMPLETE message' is transported to the MSS over the radio channel;.

56. at the MSS the MSS-PHY layer delivers the constructed bitstream to the MSS-MAC layer;
57. the MSS-MAC layer delivers the 'RELEASE COMPLETE message' (segmented in ATM cells plus additional overhead) to the MSS-ATM layer;
58. the MSS-ATM layer removes the ATM cell header and delivers the ATM information cell to the MSS-SAAL;
59. the MSS-SAAL reassembles (among other things) the received signalling ATM information cell containers. The MSS-SAAL delivers the signalling (RELEASE COMPLETE) message to the MSS-Interworking layer;

60. the MSS-Interworking layer receives the incoming signalling (RELEASE COMPLETE) message and buffers the message;
61. the MSS-Interworking layer extracts the Protocol discriminator Information Element:
    - must be 000010001 (Q.2931)
62. the MSS-Interworking layer extracts the Call reference Information Element:
    - flag: 1 = if ATM end-user originated the call reference;
      0 = if UTE originated the call reference;
    - deduce call reference value;
63. the deduced call reference value has been saved in the static table, and uniquely identifies the call the signalling message refers to;
64. the MSS-Interworking layer extracts the Message type Information Element:
    - message type must be 0101 1010 (RELEASE COMPLETE message);
    - change state of connection from Release Indication (U12) into Release Request (U11) state in static table;
    - stop timer T308
65. the MSS-Interworking layer extracts the Message length Information Element:
66. extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself);
67. the MSS-Interworking layer checks if message is errorless (Yes assumed);
68. deduce information needed to construct the static data base (optional);
69. the MSS-Interworking layer locally releases all the reservations made concerning the connection between the MSS and the UTE (connection is identified by the call reference Information element);
70. connection between UTE and MSS is released

71. in the MSS towards the ATM end-user direction:;
72. the MSS-Interworking layer generates the Call reference Information Element:
    - flag: 1 = if ATM end-user originated the call reference;
      0 = if UTE originated the call reference;
      the MSS-Interworking layer knows who initiated the call/connection establishment;
73. the MSS-Interworking layer changes the flag bit (inversion) in the saved RELEASE message;
74. state of the connection remains unchanged (Release Request);
75. the MSS-Interworking layer starts timer T308;
76. IF no reaction to the RELEASE message before expiry of timer T308 THEN release all the reservations made concerning the call reference value;
77. the MSS-Interworking layer delivers the RELEASE message to the MSS-SAAL;
78. the MSS-SAAL delivers the segmented 'RELEASE message' to the MSS-ATM layer;
79. the MSS-ATM layer delivers the bitstream to the MSS-PHY layer
80. ATM switch receives the RELEASE message and decides to sent a RELEASE message to the ATM end-user (not of our concern) and a RELEASE COMPLETE message to the MSS;
81. the 'signalling (RELEASE COMPLETE) message' is transported to the MSS over the optical channel;.
82. at the MSS the MSS-PHY layer delivers the constructed bitstream to the MSS-ATM layer;
83. the MSS-ATM layer removes the ATM cell header and delivers the signalling ATM information cells to the MSS-SAAL;
84. the MSS-SAAL reassembles (among other things) the received signalling ATM information cell containers. The MSS-SAAL delivers the signalling (RELEASE COMPLETE) message to the MSS-Interworking layer;
85. the MSS-Interworking layer receives the incoming signalling (RELEASE COMPLETE) message and buffers the message;
86. the MSS-Interworking layer extracts the Protocol discriminator Information Element:
    - must be 000010001 (Q.2931)
87. the MSS-Interworking layer extracts the Call reference Information Element:
    - flag: 0 = if ATM end-user originated the call reference;
      1 = if UTE originated the call reference;
    - deduce call reference value;
88. the deduced call reference value has been saved in the static table, and uniquely identifies the call the signalling message refers to;

89. the MSS-Interworking layer extracts the Message type Information Element:
    - message type must be 0101 1010 (RELEASE COMPLETE message);
    - change state of connection from Release Request(U11) into Null(U0) state in static table;
    - stop timer T308;
90. the MSS-Interworking layer extracts the Message length Information Element:
    - extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself);
91. the MSS-Interworking layer checks if message is errorless (Yes assumed);
92. deduce information needed to construct the static data base (optional);
93. the MSS-Interworking layer locally releases all the reservations made concerning the connection (connection is identified by the call reference Information element);
94. connection between ATM switch and UTE is released;

95.connection between (MSS and ATM switch) and between (MSS and UTE) are released;

IV_Release_e

1. the UTE2 decides to send a RELEASE message to the MSS;

2. 'RELEASE message' is transported to the PRP2 using the optical channel;

3. the PRP2-PHY layer delivers the received bitstream to the PRP2-ATM layer;
4. the PRP2-ATM layer deserialises the bitstream into ATM cells;
5. the PRP2-ATM layer delivers signalling ATM cell (if any) to the PRP2-Interworking layer;
6. the PRP2-Interworking layer buffers (in a receive buffer) the incoming signalling ATM cells (buffering is needed because the PRP2-MAC ,maybe , first has to content for a free (signalling) slot);
7. the PRP2-Interworking layer starts a timer to measure the time between two successive signalling cells;
8. on expiry of the timer, set the static table inuse value to false;
9. the PRP2-Interworking layer checks if the static table inuse value is true or false;
10. if inuse is false then
    - ask the MAC layer to content for an uplink (signalling) timeslot and start timer for contention time;
    - on expiry of timer delete buffered signalling ATM cell
    - the PRP2-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the PRP2-Interworking layer stops the timer;
    - the PRP2-Interworking layer changes the static table field inuse to true;
11. if inuse is true then the PRP2-Interworking layer delivers the signalling ATM cell to the MAC layer;
12. the PRP2-Interworking layer stops the timer;
13. the PRP2-Interworking layer checks if there is another signalling ATM cell in the de-serialiser (or better, waits until the PRP2-ATM layer delivers a signalling cell before timer expiry);
14. if no signalling ATM cell available in de-serialiser then change static table field inuse to false (end);
15. if ATM cell available in de-serialiser then
    - the PRP2-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the PRP2-Interworking layer continues this procedure until there is no signalling ATM cell in the de-serialiser;
16. the PRP2-MAC layer delivers the bitstream to the PRP2-PHY layer;

17. the 'RELEASE message' is transported to the MSS over the radio channel;.

18. at the MSS the MSS-PHY layer delivers the constructed bitstream to the MSS-MAC layer;
19. the MSS-MAC layer delivers the 'RELEASE message' (segmented in ATM cells plus additional overhead) to the MSS-ATM layer;
20. the MSS-ATM layer removes the ATM cell header and delivers the ATM information cell to the MSS-SAAL;
21. the MSS-SAAL reassembles (among other things) the received signalling ATM information cell containers. The MSS-SAAL delivers the RELEASE message to the MSS-Interworking layer;
22. the MSS-Interworking layer receives the incoming signalling (RELEASE) message and buffers the message;
23. the MSS-Interworking layer extracts the Protocol discriminator Information Element:
    - must be 000010001 (Q.2931)
24. the MSS-Interworking layer extracts the Call reference Information Element:
    - flag: 1 = if ATM end-user originated the call reference;
      0 = if UTE originated the call reference;
      MSS-Interworking layer knows who initiated the call/connection establishment;
    - call reference value is also available in the static table and identifies the call/connection the signalling message refers to;
25. the MSS-Interworking layer extracts the Message type Information Element:
    - message type must be 0101 1101 (RELEASE message);
    - change state of connection of UTE2 from Active(U10) into Release Request(U11) state in static table;
26. the MSS-Interworking layer extracts the Message length Information Element:
    - extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself);

27. the MSS-Interworking layer extracts the Cause Information Element and deduces statistical data for the static database, information maintained in the static database;
28. the MSS-Interworking layer checks if message is errorless (Yes assumed);
29. MSS-Interworking layer saves the signalling message because is has to send a RELEASE message to UTE1;
30. MSS-interworking layer is now involved in two flows:
    - MSS sends RELEASE COMPLETE message towards release initiated UTE2;
    - MSS sends RELEASE message towards not release initiated UTE1;

31. first release call/connection towards the release initiated UTE2 (by doing so we avoid multiple uses of the same SVCI which could cause difficulties when both the MPS's want to make use of the SVCI (dedicated RVCI);
32. before the MSS-Interworking is able to send the RELEASE COMPLETE message, it first has to generate the message type, because:
    - protocol discriminator ⇒ 0000 1001 (same as in the RELEASE message);
    - call reference ⇒ same as in the RELEASE message;
    - message type ⇒ 0101 1010 (RELEASE COMPLETE message);
    - message length ⇒ number of octets of the message contents (same as in the RELEASE message);
    - cause ⇒ reason which caused the rejection of the call (same as in the RELEASE message);
33. MSS-Interworking layer deduces statistical data (what ever this maybe) for static database;
34. MSS-Interworking layer changes the state of the call/connection of UTE2 from Release Request (U11) to Null (U0);
35. MSS-Interworking layer locally releases all the reservation made for this particular call (in the direction towards the UTE2);
36. MSS-Interworking layer delivers the RELEASE COMPLETE message to the SAAL layer;
37. after the MSS-SAAL layer has done the necessary actions and segmented the setup message into ATM-information containers. The flow of ATM information containers is delivered to the MSS-ATM layer;
38. the MSS-ATM layer adds the ATM-header to the ATM-information container resulting in the same ATM cell corresponding to the arrived ATM-cell at the beginning;
39. the ATM cells and all the necessary information being of importance to the MSS-MAC are send to the MSS-MAC;
40. MSS-MAC delivers the bitstream to the MSS-PHY layer;

41. RELEASE COMPLETE message is transported to the PRP2 using the radiochannel;

42. the PRP2-PHY layer delivers the constructed bitstream to the PRP2-MAC layer;
43. the PRP2-MAC delivers the broadcast cell to the PRP2-Interworking layer;
44. the PRP2-MAC delivers (if any) signalling cell(s) (VPI/VCI=0/5) to the Interworking layer;
45. the PRP2-Interworking layer buffers the broadcast cell;
46. the PRP2-Interworking layer buffers the signalling cell(s);
47. the PRP2-Interworking layer deduces signalling broadcast cell from the broadcast cell;
48. the PRP2-Interworking layer checks if signalling broadcast cell is nil;
49. if nil then delete broadcast cell and signalling broadcast cell, else continue (continue is assumed);
50. the PRP2-Interworking layer deduces the RVCI and the internal address (or ATM end system address or E.164 address) from the signalling broadcast cell;
51. the PRP2-Interworking layer checks if the internal address is the address of the PRP2 (MPS)
    - if unequal then do not proceed and delete the buffered ATM signalling cells (if any) and the broadcast cell;
    - if yes then proceed (yes is assumed);
52. the PRP2-Interworking layer saves the RVCI in the static table;
53. the PRP2-Interworking layer sets the in-use field in the static table to yes;
54. the PRP2-Interworking layer starts timer concerning reaction of the UTE;
55. if timer expiry before answer of UTE then set the in-use field to no;
56. the PRP2-Interworking layer feeds the buffered ATM (signalling) cells to the ATM-layer (where, among other things, the ATM cells are serialised into a bitstream);
57. PRP2-ATM layer delivers bitstream to the PRP2-PHY layer;

58. RELEASE COMPLETE message is transported to the UTE2-PHY layer, using the optical channel;
59. the RELEASE COMPLETE message arrives at the destination, UTE2
60. connection between MSS and release initiated UTE2 is released;

61. then initiate call/connection release towards the not release initiated UTE1;
62. MSS-interworking layer changes the state of the connection/call (towards UTE1) from Active(U10) into Release Indication (U12);
63. MSS-interworking layer sends the (modified and saved) RELEASE message towards UTE1;
64. MSS-interworking layer starts T308;
65. IF no reaction to the RELEASE message before expiry of timer T308 THEN release all the reservations made concerning the call reference value and UTE1;
66. MSS-interworking layer delivers the (saved) RELEASE message to the MSS-SAAL;
67. after the MSS-SAAL layer has done the necessary actions and segmented the signalling (RELEASE) message into ATM-information containers, it delivers the ATM-information containers to the MSS-ATM layer;
68. the MSS-ATM layer adds the ATM-header to the ATM-information container;
69. the ATM cells and all the necessary information being of importance to the MSS-MAC are send to the MSS-MAC. There is a way of synchronisation needed between the MSS-MAC and the MSS-Interworking layer which can be accomplished by the use of the primitives: request and indication with there own message indication;
70. MSS-MAC delivers the bitstream to the MSS-PHY layer (MAC and PHY take care of the 'reliable' transportation of the RELEASE message to the PRP1);

71. RELEASE message is transported to the PRP1 using the radiochannel;

72. the PRP1-PHY layer delivers the constructed bitstream to the PRP1-MAC layer;
73. the PRP1-MAC delivers the broadcast cell to the PRP1-Interworking layer;
74. the PRP1-MAC delivers (if any) signalling cell(s) (VPI/VCI=0/5) to the Interworking layer;
75. the PRP1-Interworking layer buffers the broadcast cell;
76. the PRP1-Interworking layer buffers the signalling cell(s);
77. the PRP1-Interworking layer deduces signalling broadcast cell from the broadcast cell;
78. the PRP1-Interworking layer checks if signalling broadcast cell is nil;
79. if nil then delete broadcast cell and signalling broadcast cell, else continue (continue is assumed);
80. the PRP1-Interworking layer deduces the RVCI and the internal address (or ATM end system address or E.164 address) from the signalling broadcast cell;
81. the PRP1-Interworking layer checks if the internal address is the address of the PRP1 (MPS)
    - if unequal then do not proceed and delete the buffered ATM signalling cells (if any) and the broadcast cell;
    - if yes then proceed (yes is assumed);
82. the PRP1-Interworking layer saves the RVCI in the static table;
83. the PRP1-Interworking layer sets the in-use field in the static table to yes;
84. the PRP1-Interworking layer starts timer concerning reaction of the UTE;
85. if timer expiry before answer of UTE then set the in-use field to no;
86. the PRP1-Interworking layer feeds the buffered ATM (signalling) cells to the ATM-layer (where, among other things, the ATM cells are serialised into a bitstream);
87. PRP1-ATM layer delivers bitstream to the PRP1-PHY layer;

88. 'RELEASE message' is transported to the UTE1 using the optical channel;
89. the UTE1 reacts to the 'RELEASE message' by sending a 'RELEASE COMPLETE' message to the MSS (through the PRP1);

90. the PRP1-PHY layer delivers the received bitstream to the PRP1-ATM layer;
91. the PRP1-ATM layer deserialises the bitstream into ATM cells;
92. the PRP1-ATM layer delivers signalling ATM cell (if any) to the PRP1-Interworking layer;
93. the PRP1-Interworking layer buffers (in a receive buffer) the incoming signalling ATM cells (buffering is needed because the PRP1-MAC ,maybe , first has to content for a free (signalling) slot);
94. the PRP1-Interworking layer starts a timer to measure the time between two successive signalling cells;
95. on expiry of the timer, set the static table inuse value to false;

96. the PRP1-Interworking layer checks if the static table inuse value is true or false;
97. if inuse is false then
    - ask the MAC layer to content for an uplink (signalling) timeslot and start timer for contention time;
    - on expiry of timer delete buffered signalling ATM cell
    - the PRP1-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the PRP1-Interworking layer stops the timer;
    - the PRP1-Interworking layer changes the static table field inuse to true;
98. if inuse is true then the PRP1-Interworking layer delivers the signalling ATM cell to the MAC layer;
99. the PRP1-Interworking layer stops the timer;
100. the PRP1-Interworking layer checks if there is another signalling ATM cell in the de-serialiser (or better, waits until the PRP1-ATM layer delivers a signalling cell before timer expiry);
101. if no signalling ATM cell available in de-serialiser then change static table field inuse to false (end);
102. if ATM cell available in de-serialiser then
    - the PRP1-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the PRP1-Interworking layer continues this procedure until there is no signalling ATM cell in the de-serialiser;
103. the PRP1-MAC layer delivers the bitstream to the PRP1-PHY layer;

104. the 'RELEASE COMPLETE message' is transported to the MSS over the radio channel;.

105. at the MSS the MSS-PHY layer delivers the constructed bitstream to the MSS-MAC layer;
106. the MSS-MAC layer delivers the 'RELEASE COMPLETE message' (segmented in ATM cells) to the MSS-ATM layer;
107. the MSS-ATM layer removes the ATM cell header and delivers the ATM information cell to the MSS-SAAL;
108. the MSS-SAAL reassembles (among other things) the signalling (RELEASE COMPLETE) message'. The MSS-SAAL delivers the signalling (RELEASE COMPLETE) message to the MSS-Interworking layer;
109. the MSS-Interworking layer receives the incoming signalling (RELEASE COMPLETE) message and buffers the message;
110. the MSS-Interworking layer extracts the Protocol discriminator Information Element:
    - must be 000010001 (Q.2931)
111. the MSS-Interworking layer extracts the Call reference Information Element:
    - flag must be 1 (message sent to side that originates the call reference);
112. the deduced call reference value has been saved in the static table, and uniquely identifies the call the signalling message refers to;
113. the MSS-Interworking layer extracts the Message type Information Element:
    - message type must be 0101 1010 (RELEASE COMPLETE message);
    - change state of connection from Release Indication (U12) into Null (U0) state in static table;
    - stop T308;
114. the MSS-Interworking layer extracts the Message length Information Element:
    - extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself);
115. deduce information needed to construct the static data base (optional);
116. the MSS-Interworking layer checks if message is errorless (Yes assumed);
117. the MSS-Interworking layer locally releases all the reservations made concerning the connection towards the UTE1 (connection is identified by the call reference Information element);
118. connection between not release initiated UTE1 and MSS is released;
119. connection between MPS1 and MPS2 is released;

IV_Release_f

1. MSS decides to release the connection between two MPS's;

2. MSS initiate call release towards the MPS1 which has initiated the call/connection establishment;
3. the MSS-Interworking layer has to generate the RELEASE message;
4. the MSS-Interworking layer generates the Protocol discriminator Information Element:
    - must be 000010001 (Q.2931)
5. the MSS-Interworking layer generates the Call reference Information Element:
    - flag must be 1(message sent to the side that originates the call reference);
    - call reference value (23 bit) of the connection which the MSS wants to release (is saved in static table);
6. the MSS-Interworking layer generates the Message type Information Element:
    - message type must be 0100 1101 (RELEASE message);
    - change state of connection of UTE1 from Active(U10) into Release Indication(U12) state in static table;
7. the MSS-Interworking layer generates the Cause Information Element:
    - Cause values are:
        - #16 ⇒ normal call clearing;
        - #41 ⇒ temporary failure;
        - #111 ⇒ protocol error, unspecified;
8. the MSS-Interworking layer calculates the message length:
    - extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself);
9. the MSS-Interworking layer generates the Message length Information Element:
10. the MSS-Interworking layer constructs the RELEASE message, and saves the RELEASE message (also needed in other direction) ;
11. the MSS-Interworking layer starts timer T308;
12. IF no reaction to the RELEASE message before expiry of timer T308 THEN release all the reservations made concerning the call reference value and UTE1;
13. RELEASE message is delivered to the MSS-SAAL layer;
14. after the MSS-SAAL layer has done the necessary actions and segmented the signalling (RELEASE) message into ATM-information containers, it delivers the ATM-information containers to the MSS-ATM layer;
15. the MSS-ATM layer adds the ATM-header to the ATM-information container resulting in the same ATM cell corresponding to the arrived ATM-cell at the beginning;
16. the ATM cells and all the necessary information being of importance to the MSS-MAC are send to the MSS-MAC;
17. MSS-MAC delivers the bitstream to the MSS-PHY layer;

18. 'RELEASE message' is transported to the PRP1 using the radiochannel;

19. the PRP1-PHY layer delivers the constructed bitstream to the PRP1-MAC layer;
20. the PRP1-MAC delivers the broadcast cell to the PRP1-Interworking layer;
21. the PRP1-MAC delivers (if any) signalling cell(s) (VPI/VCI=0/5) to the Interworking layer;
22. the PRP1-Interworking layer buffers the broadcast cell;
23. the PRP1-Interworking layer buffers the signalling cell(s);
24. the PRP1-Interworking layer deduces signalling broadcast cell from the broadcast cell;
25. the PRP1-Interworking layer checks if signalling broadcast cell is nil;
26. if nil then delete broadcast cell and signalling broadcast cell, else continue (continue is assumed);
27. the PRP1-Interworking layer deduces the RVCI and the internal address (or ATM end system address or E.164 address) from the signalling broadcast cell;
28. the PRP1-Interworking layer checks if the internal address is the address of the PRP1 (MPS)
    - if unequal then do not proceed and delete the buffered ATM signalling cells (if any) and the broadcast cell;
    - if yes then proceed (yes is assumed);
29. the PRP1-Interworking layer saves the RVCI in the static table;

30. the PRP1-Interworking layer sets the in-use field in the static table to yes;
31. the PRP1-Interworking layer starts timer concerning reaction of the UTE;
32. if timer expiry before answer of UTE then set the in-use field to no;
33. the PRP1-Interworking layer feeds the buffered ATM (signalling) cells to the ATM-layer (where, among other things, the ATM cells are serialised into a bitstream);
34. PRP1-ATM layer delivers bitstream to the PRP1-PHY layer;

35. 'RELEASE message' is transported to the UTE1 using the optical channel;

36. the UTE1 decides to send a RELEASE COMPLETE message to the PRP1 (which finally has to arrive at the ATM switch) ;

37. 'RELEASE COMPLETE message' is transported to the PRP1 using the optical channel;

38. the PRP1-PHY layer delivers the received bitstream to the PRP1-ATM layer;
39. the PRP1-ATM layer deserialises the bitstream into ATM cells;
40. the PRP1-ATM layer delivers signalling ATM cell (if any) to the PRP1-Interworking layer;
41. the PRP1-Interworking layer buffers (in a receive buffer) the incoming signalling ATM cells (buffering is needed because the PRP1-MAC ,maybe , first has to content for a free (signalling) slot);
42. the PRP1-Interworking layer starts a timer to measure the time between two successive signalling cells;
43. on expiry of the timer, set the static table inuse value to false;
44. the PRP1-Interworking layer checks if the static table inuse value is true or false;
45. if inuse is false then
    - ask the MAC layer to content for an uplink (signalling) timeslot and start timer for contention time;
    - on expiry of timer delete buffered signalling ATM cell
    - the PRP1-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the PRP1-Interworking layer stops the timer;
    - the PRP1-Interworking layer changes the static table field inuse to true;
46. if inuse is true then the PRP1-Interworking layer delivers the signalling ATM cell to the MAC layer;
47. the PRP1-Interworking layer stops the timer;
48. the PRP1-Interworking layer checks if there is another signalling ATM cell in the de-serialiser (or better, waits until the PRP1-ATM layer delivers a signalling cell before timer expiry);
49. if no signalling ATM cell available in de-serialiser then change static table field inuse to false (end);
50. if ATM cell available in de-serialiser then
    - the PRP1-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the PRP1-Interworking layer continues this procedure until there is no signalling ATM cell in the de-serialiser;
51. the PRP1-MAC layer delivers the bitstream to the PRP1-PHY layer;

52. the 'RELEASE COMPLETE message' is transported to the MSS over the radio channel;.

53. at the MSS the MSS-PHY layer delivers the constructed bitstream to the MSS-MAC layer;
54. the MSS-MAC layer delivers the 'RELEASE COMPLETE message' (segmented in ATM cells plus additional overhead) to the MSS-ATM layer;
55. the MSS-ATM layer removes the ATM cell header and delivers the ATM information cell to the MSS-SAAL;
56. the MSS-SAAL reassembles (among other things) the received signalling ATM information cell containers. The MSS-SAAL delivers the signalling (RELEASE COMPLETE) message to the MSS-Interworking layer;
57. the MSS-Interworking layer receives the incoming signalling (RELEASE COMPLETE) message and buffers the message;
58. the MSS-Interworking layer extracts the Protocol discriminator Information Element:
    - must be 000010001 (Q.2931)
59. the MSS-Interworking layer extracts the Call reference Information Element:
    - flag must be 0 (message sent from side that originates the call reference);
60. the deduced call reference value has been saved in the static table, and uniquely identifies the call the signalling message refers to;

61. the MSS-Interworking layer extracts the Message type Information Element:
    - message type must be 0101 1010 (RELEASE COMPLETE message);
    - change state of connection of UTE1 from Release Indication(U12) into Null state in static table;
    - stop timer T308;
62. the MSS-Interworking layer extracts the Message length Information Element;
63. extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself);
64. the MSS-Interworking layer checks if message is errorless (Yes assumed);
65. deduce information needed to construct the static data base (optional);
66. the MSS-Interworking layer locally releases all the reservations made concerning the connection (connection is identified by the call reference Information element) between MSS and UTE1(the one which initiated the call establishment);

67. connection between MPS1 and MSS is released

68. MSS initiate call release towards the MPS2 which has not initiated the call/connection establishment;
69. the MSS-Interworking changes state of connection of UTE2 from Active(U10) into Release Indication(U12) state in static table;
70. the MSS-Interworking layer uses the saved RELEASE message;
71. the MSS-Interworking layer changes the Call reference Information Element:
    - flag must be 0(message sent to the side that did not originate the call reference);
72. the MSS-Interworking layer starts timer T308;
73. IF no reaction to the RELEASE message before expiry of timer T308 THEN release all the reservations made concerning the call reference value and UTE1;
74. RELEASE message is delivered to the MSS-SAAL layer;
75. after the MSS-SAAL layer has done the necessary actions and segmented the signalling (RELEASE) message into ATM-information containers, it delivers the ATM-information containers to the MSS-ATM layer;
76. the MSS-ATM layer adds the ATM-header to the ATM-information container resulting in the same ATM cell corresponding to the arrived ATM-cell at the beginning;
77. the ATM cells and all the necessary information being of importance to the MSS-MAC are send to the MSS-MAC;
78. MSS-MAC delivers the bitstream to the MSS-PHY layer;

79. 'RELEASE message' is transported to the PRP2 using the radiochannel;

80. the PRP2-PHY layer delivers the constructed bitstream to the PRP2-MAC layer;
81. the PRP2-MAC delivers the broadcast cell to the PRP2-Interworking layer;
82. the PRP2-MAC delivers (if any) signalling cell(s) (VPI/VCI=0/5) to the Interworking layer;
83. the PRP2-Interworking layer buffers the broadcast cell;
84. the PRP2-Interworking layer buffers the signalling cell(s);
85. the PRP2-Interworking layer deduces signalling broadcast cell from the broadcast cell;
86. the PRP2-Interworking layer checks if signalling broadcast cell is nil;
87. if nil then delete broadcast cell and signalling broadcast cell, else continue (continue is assumed);
88. the PRP2-Interworking layer deduces the RVCI and the internal address (or ATM end system address or E.164 address) from the signalling broadcast cell;
89. the PRP2-Interworking layer checks if the internal address is the address of the PRP2 (MPS)
    - if unequal then do not proceed and delete the buffered ATM signalling cells (if any) and the broadcast cell;
    - if yes then proceed (yes is assumed);
90. the PRP2-Interworking layer saves the RVCI in the static table;
91. the PRP2-Interworking layer sets the in-use field in the static table to yes;
92. the PRP2-Interworking layer starts timer concerning reaction of the UTE;
93. if timer expiry before answer of UTE then set the in-use field to no;
94. the PRP2-Interworking layer feeds the buffered ATM (signalling) cells to the ATM-layer (where, among other things, the ATM cells are serialised into a bitstream);
95. PRP2-ATM layer delivers bitstream to the PRP2-PHY layer;

96. 'RELEASE message' is transported to the UTE2 using the optical channel;

97. the UTE2 decides to send a RELEASE COMPLETE message to the PRP2 (which finally has to arrive at the ATM switch) ;

98. 'RELEASE COMPLETE message' is transported to the PRP2 using the optical channel;

99. the PRP2-PHY layer delivers the received bitstream to the PRP2-ATM layer;
100. the PRP2-ATM layer deserialises the bitstream into ATM cells;
101. the PRP2-ATM layer delivers signalling ATM cell (if any) to the PRP2-Interworking layer;
102. the PRP2-Interworking layer buffers (in a receive buffer) the incoming signalling ATM cells (buffering is needed because the PRP2-MAC ,maybe , first has to content for a free (signalling) slot);
103. the PRP2-Interworking layer starts a timer to measure the time between two successive signalling cells;
104. on expiry of the timer, set the static table inuse value to false;
105. the PRP2-Interworking layer checks if the static table inuse value is true or false;
106. if inuse is false then
    - ask the MAC layer to content for an uplink (signalling) timeslot and start timer for contention time;
    - on expiry of timer delete buffered signalling ATM cell
    - the PRP2-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the PRP2-Interworking layer stops the timer;
    - the PRP2-Interworking layer changes the static table field inuse to true;
107. if inuse is true then the PRP2-Interworking layer delivers the signalling ATM cell to the MAC layer;
108. the PRP2-Interworking layer stops the timer;
109. the PRP2-Interworking layer checks if there is another signalling ATM cell in the de-serialiser (or better, waits until the PRP2-ATM layer delivers a signalling cell before timer expiry);
110. if no signalling ATM cell available in de-serialiser then change static table field inuse to false (end);
111. if ATM cell available in de-serialiser then
    - the PRP2-Interworking layer delivers the signalling ATM cell to the MAC layer;
    - the PRP2-Interworking layer continues this procedure until there is no signalling ATM cell in the de-serialiser;
112. the PRP2-MAC layer delivers the bitstream to the PRP2-PHY layer;

113. the 'RELEASE COMPLETE message' is transported to the MSS over the radio channel;.

114. at the MSS the MSS-PHY layer delivers the constructed bitstream to the MSS-MAC layer;
115. the MSS-MAC layer delivers the 'RELEASE COMPLETE message' (segmented in ATM cells plus additional overhead) to the MSS-ATM layer;
116. the MSS-ATM layer removes the ATM cell header and delivers the ATM information cell to the MSS-SAAL;
117. the MSS-SAAL reassembles (among other things) the received signalling ATM information cell containers. The MSS-SAAL delivers the signalling (RELEASE COMPLETE) message to the MSS-Interworking layer;
118. the MSS-Interworking layer receives the incoming signalling (RELEASE COMPLETE) message and buffers the message;
119. the MSS-Interworking layer extracts the Protocol discriminator Information Element:
    - must be 000010001 (Q.2931)
120. the MSS-Interworking layer extracts the Call reference Information Element:
    - flag must be 1 (message sent from side that did not originate the call reference);
121. the deduced call reference value has been saved in the static table, and uniquely identifies the call the signalling message refers to;
122. the MSS-Interworking layer extracts the Message type Information Element:
    - message type must be 0101 1010 (RELEASE COMPLETE message);
    - change state of connection from Release Indication (U12) into Null (U0) state in static table;
    - stop timer T308;
123. the MSS-Interworking layer extracts the Message length Information Element;
124. extract length of message (excluding protocol discriminator, call reference, message type, message length indication itself);
125. deduce information needed to construct the static data base (optional);
126. the MSS-Interworking layer checks if message is errorless (Yes assumed);

127. the MSS-Interworking layer locally releases all the reservations made concerning the connection (connection is identified by the call reference Information element) between MSS and UTE2(the one which did not initiate the call establishment);
128. connection between UTE2 and MSS is released;
129. connection between MPS1 and MPS2 is released;

Protocol description UTE-PRP

```
RECEIVE signalling_ATM_cell FROM PRP_deserialiser;
BUFFER ATM_cell IN inp_buf;

start Tzzzz        {timer used to measure time between two succesive signalling cells}
ON expiry OF Tzzz
⇒
        CHANGE static_table.inuse ⇒ FALSE {signalling timeslot not reserved to this MPS}

IF static_table.inuse = TRUE
THEN    {PRP has to content for a signalling timeslot in uplink frame}
        activate_MAC_contention
        start Tuuu;        {timer used to measure contention time}

        ON expiry OF Tuuu
        ⇒
                DELETE inp_buf

ELSE    { timeslot reserved to this MPS}
        deliver(inp_buf, MAC);
        stop Tyyy;

LOOP
CHECK available_ATM_cell (deserialiser);
IF available_ATM_cell = FALSE
THEN    {no more ATM cells need to be transported from the MPS to the MSS}
        CHANGE static_table.inuse = FALSE;
ELSE
        RECEIVE signalling_ATM_cell FROM PRP_deserialiser;
        BUFFER ATM_cell IN inp_buf;
        deliver(inp_buf, MAC);
        CONTINUE WITH LOOP

END {Protocol UTE_PRP}
```

Protocol description MSS-PRP

```
RECEIVE broadcast_cell FROM PRP-MAC;
BUFFER broadcast_cell IN inp_buf_broadcast;

{if any}
RECEIVE signalling_ATM_cells;
BUFFER signalling_ATM_cells IN inp_buf_sign_cells;

DEDUCE sign_broadcast_cell FROM inp_buf_broadcast;
{first cell in broadcast cell}

IF sign_broadcast_cell = NIL
THEN
        DELETE inp_buf_broadcast;
        DELETE inp_buf_sign_cells;
        STOP;
ELSE
        DEDUCE internal_address FROM sign_broadcast_cell;      {internal address used}

IF internal_address ≠ PRP_internal_address
THEN
        {signalling ATM cells are not addressed to this PRP (UTE)}
        DELETE inp_buf_broadcast;
        DELETE inp_buf_sign_cells;
ELSE
        {signalling ATM cells are addressed to this PRP (UTE)}
        DEDUCE RVCI FROM sign_broadcast_cell;     {extract the RVCI of this call/connection}
        SAVE static_table.RVCI = RVCI             {mapping of RVCI for this call/connection}
        SAVE static_table.inuse = yes             {indicating that there is a signalling uplink path}
                                                  {reserved}
        START Tyyy                                {timer for reaction from UTE, if any}
        ON expiry OF Tyyy static_table.inuse = no {releasing the reservation of the SRVCI slot(s)}
        deliver(inp_buf_sign_cells, UTE);         {signalling cells delivered to de/serialiser}
                                                  {only serialising of the ATM cellstream into a }
                                                  {bitstream is needed}

END {protocol MSS_PRP}
```

Protocol description MSS call release

```
{call/connection is identified by the unique call reference value = CRV}
continue = FALSE;
MSS_initiated_call_release = TRUE;

IF static_table.CRV.CRV+ = empty
THEN
{external call/connection}
        {first release call/connection towards the MPS}
        IF static_table.CRV.origin = internal
        THEN
                {MPS originated the call/connection establishment}
                F = 1;
        ELSE
                {ATM end-user originated the call/connection establishment}
                F = 0;

        CHANGE static_table.CRV.state FROM U10 ⇒ U12          {state is Release Indication}

        create_message (RELEASE, CRV, cause#, F);             {create release message with}
                                                              {appropriate CRV, cause# and }
                                                              {flag bit}
        SAVE RELEASE ⇒ buffer_release;                        {release message saved because}
                                                              {message also needed in oppo-}
                                                              {site direction}
        GENERATE statistical data;
        SAVE statistical data IN static_data_base
        deliver(RELEASE, internal, CRV);
        START T308.CRV;
        on expiry of timer T308.CRV timer_call_releasing;

        WAIT UNTIL continue = TRUE;
        {then release call/connection towards the ATM end-user}

        CHANGE (inp_buf_CR.F : inverse F);                    {message send to side that did}
                                                              {not originated call reference}
                                                              {Use release message with}
                                                              {appropriate CRV, cause# but}
                                                              {flag bit is inversed}
        GENERATE statistical data;
        SAVE statistical data IN static_data_base
        deliver(RELEASE, external, CRV);
        START T308.CRV;
        on expiry of timer T308.CRV timer_call_releasing;


ELSE
{internal call/connection}
        {release call/connection towards first MPS1}
        IF static_table.CRV.origin = internal
        THEN
                {MPS1 originated the call/connection establishment}
                F = 1;
```

ELSE

{ MPS2 originated the call/connection establishment}
F = 0;

CHANGE static_table.CRV.state FROM U10 ⇒ U12 {state is Release Indication}

create_message (RELEASE, CRV, cause#, F); {create release message with} {appropriate CRV, cause# and } {flag bit}

SAVE RELEASE ⇒ buffer_release; {release message saved because} {message also needed in oppo-} {site direction}

GENERATE statistical data;
SAVE statistical data IN static_data_base
deliver(RELEASE, internal, CRV);
START T308.CRV;
on expiry of timer T308.CRV *timer_call_releasing;*

WAIT UNTIL continue = TRUE;
{then release call/connection towards the ATM end-user}

CRV := static_table.CRV.CRV+ {CRV of other party}
CHANGE (inp_buf_CR.CRV : CRV); {change CRV in RELEASE} {message}

CHANGE (inp_buf_CR.F : inverse F); {message send to side that did} {not originated call reference} {Use release message with} {appropriate CRV, cause# but} {flag bit is inversed}

GENERATE statistical data;
SAVE statistical data IN static_data_base
deliver(RELEASE, internal, CRV);
START T308.CRV;
on expiry of timer T308.CRV *timer_call_releasing;*

Protocol description PRP to MSS

```
RECEIVE signalling message  from SAAL                                {read/buffer received message }
BUFFER frame in inp_buf;                                             {into input buffer}


DEDUCE frame information elements :
        Protocol discriminator      ⇒ inp_buf_PD;
        Call reference              ⇒ inp_buf_CR;
        Message type                ⇒ inp_buf_MT;
        Message length              ⇒ inp_buf_ML;
END {DEDUCE}

DEDUCE info in information elements FROM:

        Call reference: inp_buf_CR
                length              ⇒ length;
                flag                ⇒ F;
                call reference value⇒ CRV;

        Message type : inp_buf_MT
                message type        ⇒ MT;

        Message length : inp_buf_ML
                message length      ⇒ ML;
END {DEDUCE};


CASE of MT

        SETUP :
                MAKE row in static_table(CRV)⇒ static_table.CRV;     {create row in static table}
                SAVE static_table.CRV.CRV=CRV;                       {CRV saved in static table}
                CHANGE static_table.CRV.state FROM U0 into U1;               {state is Call initiated}
                SAVE static_table.CRV.origin=internal;               {internal call/conn. request}

                DEDUCE frame information elements :
                        ATM traffic descriptor              ⇒ inp_buf_ATD;
                        Broadband Bearer Capability         ⇒ inp_buf_BBC;
                        Called party Number                 ⇒ inp_buf_CPN;
                        Quality of  Service parameter       ⇒ inp_buf_QOS;
                END {DEDUCE}

                error_control_of_message;

                DEDUCE info in information elements:
                        ATM traffic descriptor ⇒ inp_buf_ATD
                                {if available}
                                forward peak cell rate with CLP=0              ⇒ forw_peak_0;
                                forward peak cell rate with CLP=1              ⇒ forw_peak_1;
                                backward peak cell rate with CLP=0        ⇒ back_peak_0;
                                backward peak cell rate with CLP=1        ⇒ back_peak_1;
                                forward sustainable cell rate with CLP=0  ⇒ forw_sust_0;
                                forward sustainable cell rate with CLP=1  ⇒ forw_sust_1;
                                backward sustainable cell rate with CLP=0 ⇒ back_sust_0;
                                backward sustainable cell rate with CLP=1 ⇒ back_sust_1;
```

```
                forward maximum burst size CLP=0      ⇒ forw_max_0;
                forward maximum burst size CLP=1      ⇒ forw_max_1;
                backward maximum burst size CLP=0     ⇒ back_max_0;
                backward maximum burst size CLP=1     ⇒ back_max_1;
                SAVE info in static_data_base;

            Broadband bearer capability ⇒ inp_buf_BBC;
                bearer class                ⇒ bear_class;
                traffic type                ⇒ traffic_type;
                timing requirements         ⇒ timing_requir;
                susceptibility to clipping  ⇒ susc_clip;
                user plane connection       ⇒ user_con;
                SAVE info in static_data_base;

            Called party number ⇒ inp_buf_CPN;
                type of number              ⇒ typ_numb;
                Addressing/number plan      ⇒ add_numb;
                ATM endsystem address       ⇒ ATM_add;

            Quality of service ⇒ inp_buf_QoS;
                QoS class forward           ⇒ QoS_forw;
                QoS class backward          ⇒ QoS_back;
                SAVE info in static_data_base;
        END {DEDUCE}

        check_registered(ATM_add);

        IF check_registered = TRUE
        THEN {internal call/connection}
            CREATE(CRV) ⇒ CRV2;                       {choose a CRV for connection}
                                                      {MSS-called MEDIAN user}
            SAVE static_table.CRV.CRV+ = CRV2;
            CONTINUE with message_handling(SETUP, internal, CRV);

        ELSE {external call/connection}
            CONTINUE with message_handling(SETUP, external, CRV);

END{SETUP}


CALL PROCEEDING :
        STOP T303.CRV;
        DEDUCE frame information elements :
            Connection Identifier ⇒ inp_buf_CI;

        error_control_of_message;

        DEDUCE info FROM information elements:
            Connection identifier ⇒ inp_buf_CON;
                VPI/VCI ⇒ VPI_VCI;

        CHANGE static_table.CRV.state FROM U6 INTO U3;   {state is outgoing call }
                                                          {proceeding}
        IF static_table.CRV.CRV+ = not empty
        THEN
            {internal connection]
            message_handling(CALL PROCEEDING, internal, (static_table.CRV.CRV+));
```

```
        ELSE
                {external connection}
                message_handling(CALL PROCEEDING, external,CRV);

END {CALL PROCEEDING}


CONNECT:
        error_control_of_message;

        IF static_table.CRV.CRV+= not empty
        THEN
                {internal connection]
                STOP T310.(static_table.CRV.CRV+);
                CHANGE static_table.CRV.state FROM U3 into U10          {Active}
                message_handling(CONNECT, internal, (static_table.CRV.CRV+));

        ELSE
                {external connection}
                STOP T310.CRV;
                CHANGE static_table.CRV.state FROM U3 into U8           {Connect request}
                message_handling(CONNECT, external, CRV);

END{ CONNECT}


CONNECT ACKNOWLEDGE :
        error_control_of_message;

        IF static_table.CRV.CRV+= not empty
        THEN
                {internal connection]
                STOP T313.CRV;
                message_handling(CONNECT ACKNOWLEDGE, internal,
                (static_table.CRV.CRV+));

        ELSE
                {external connection}
                STOP Txxx.CRV;
                message_handling(CONNECT ACKNOWLEDGE, external, CRV);

END {CONNECT ACKNOWLEDGE}


RELEASE COMPLETE :
        DEDUCE frame information elements :
                Cause ⇒ inp_buf_CI;

        error_control_of_message;

        IF1 (static_table.CRV.state=U3) AND active(T310.(static_table.CRV.CRV+)
        THEN1
                {rejection of the SETUP}
                STOP T310.( static_table.CRV.state);
                CHANGE static_table.CRV.state FROM U3 INTO U0;  {state is null}
                IF2 static_table.CRV.CRV+= empty
                THEN2
```

```
                    {external connection}
                    message_handling(RELEASE COMPLETE, external, CRV);
              ELSE2
                    {internal connection}
                    CHANGE static_table.(static_table.CRV.CRV+).state FROM U9 INTO U0;
                    DELETE row static_table (CRV);
                    message_handling(RELEASE COMPLETE, internal,
                    (static_table.CRV.CRV+));

        ELSE1
              {normal call releasing}
              IF2 (static_tabel.CRV.state= U12) AND (static_tabel.CRV.CRV+= NOT empty)
AND
              active(T308.CRV)
              THEN2
                    {scen:IVe, Ivf}
                    IF3 MSS_intiated_call_releasing ;
                    THEN3
                          {scen Ivf}
                          STOP T308.CRV;
                          CHANGE static_table.CRV.state FROM U12 INTO U0;
                          DELETE row static_table (CRV);
                          continue = TRUE;
                    ELSE3
                          {scen IVe}
                          STOP T308.CRV;
                          CHANGE static_table.CRV.state FROM U12 INTO U0;
                          DELETE row static_table (CRV);

              ELSE2
                    {scen IV a,c,d}
                    IF3 MSS intiated call releasing
                    THEN3
                          {scen IVd}
                          STOP T308.CRV;
                          continue = TRUE;

                    ELSE3
                          {scen: IVa,c}
                          STOP T308.CRV
                          CHANGE static_table.CRV.state FROM U12 INTO U0;
                          message_handling(RELEASE COMPLETE, external, CRV);

  END {RELEASE COMPLETE}


  RELEASE :
        DEDUCE frame information elements :
              Cause ⇒ inp_buf_CI;

        error_control_of_message;

        IF static_table.CRV.CRV+= empty
        THEN
              {scenario IV_b, external call}
              CHANGE static_table.CRV.state from U10 INTO U11;{state is Release request}
              message_handling(RELEASE, external, CRV);
```

```
        ELSE
                {scenario IV_e, internal call}
                SAVE RELEASE message ⇒ inp_buf;
                generate(RELEASE COMPLETE,CRV, CAUSE#); {cause# same as in RELEASE}
                CHANGE static_table.CRV.state from U10 INTO U0;

                message_handling(RELEASE COMPLETE, internal, CRV);

                USE saved RELEASE message;
                CHANGE inp_buf_CR.CRV INTO inp_buf_CR.(static_table.CRV.CRV+)
                CHANGE static_table.(static_table.CRV.CRV+).state FROM U10 INTO U12;

                message_handling(RELEASE , internal, (static_table.CRV.CRV+));

    END {RELEASE}



message_handling;
{SETUP}
CASE of MT
    SETUP :

        CASE of initiator
                internal :
                check_QoS_forw(QoS_forw, QoS_forw_MSS, MEDIAN, MEDIAN);
                IF1 check_QoS_forw = FALSE
                THEN1
                        CHANGE static_table.CRV.state FROM U6 INTO U0
                        release_complete(qause#, internal, CRV);

                ELSE1
                check_QoS_backw(QoS_back, QoS_back_MSS, MEDIAN, MEDIAN);
                IF2 check_QoS_backw = FALSE
                THEN2
                        CHANGE static_table.CRV.state FROM U6 INTO U0
                        release_complete(qause#, internal, CRV);

                ELSE2

                check_FP_0(forw_peak_0, forw_peak_0_MSS, MEDIAN, MEDIAN);
                IF3 check_FP_0 = FALSE
                THEN3
                        CHANGE static_table.CRV.state FROM U6 INTO U0
                        release_complete(qause#, internal, CRV);

                ELSE3

                check_FP_1(forw_peak_1, forw_peak_1_MSS, MEDIAN, MEDIAN);
                IF4 check_FP_1 = FALSE
                THEN4
                        CHANGE static_table.CRV.state FROM U6 INTO U0
                        release_complete(qause#, internal, CRV);

                ELSE4

                check_BP_0(back_peak_0, back _peak_0_MSS, MEDIAN, MEDIAN);
                IF5 check_BP_0 = FALSE
```

```
THEN5
        CHANGE static_table.CRV.state FROM U6 INTO U0
        release_complete(qause#, internal, CRV);

ELSE5
check_BP_1(back_peak_1, back _peak_1_MSS, MEDIAN, MEDIAN);
IF6 check_BP_1 = FALSE
THEN6
        CHANGE static_table.CRV.state FROM U6 INTO U0
        release_complete(qause#, internal, CRV);

ELSE6
check_FS_0(forw_sust_0, forw_sust_0_MSS, MEDIAN, MEDIAN);
IF7 check_FS_0= FALSE
THEN7
        CHANGE static_table.CRV.state FROM U6 INTO U0
        release_complete(qause#, internal, CRV);

ELSE7
check_FS_1(forw_sust_1 forw_sust_1_MSS, MEDIAN, MEDIAN);
IF8 check_FS_1= FALSE
THEN8
        CHANGE static_table.CRV.state FROM U6 INTO U0
        release_complete(qause#, internal, CRV);

ELSE8
check_BS_0(back_sust_0, back_sust_0_MSS, MEDIAN, MEDIAN);
IF9 check_BS_0= FALSE
THEN9
        CHANGE static_table.CRV.state FROM U6 INTO U0
        release_complete(qause#, internal, CRV);

ELSE9
check_BS_1(back_sust_1, back_sust_1_MSS, MEDIAN, MEDIAN);
IF10 check_BS_1= FALSE
THEN10
        CHANGE static_table.CRV.state FROM U6 INTO U0
        release_complete(qause#, internal, CRV);

ELSE10
check_FM_0(forw_max_0, forw_max_0_MSS, MEDIAN, MEDIAN);
IF11 check_FM_0= FALSE
THEN11
        CHANGE static_table.CRV.state FROM U6 INTO U0
        release_complete(qause#, internal, CRV);

ELSE11
check_FM_1(forw_max_1, forw_max_1_MSS, MEDIAN, MEDIAN);
IF12 check_FM_1= FALSE
THEN12
        CHANGE static_table.CRV.state FROM U6 INTO U0
        release_complete(qause#, internal, CRV);

ELSE12
check_BM_0(back_max_0, back_max_0_MSS, MEDIAN, MEDIAN);
IF13 check_BM_0 = FALSE
THEN13
        CHANGE static_table.CRV.state FROM U6 INTO U0
```

```
        release_complete(qause#, internal, CRV);

ELSE13
check_BM_1(back_max_1, back_max_1_MSS, MEDIAN, MEDIAN);
IF14 check_BM_1 = FALSE

THEN14
        CHANGE static_table.CRV.state FROM U6 INTO U0
        release_complete(qause#, internal, CRV);

ELSE14
IF15 available(VPI_VCI) = FALSE
THEN15
        qause# = 35 AND CHANGE static_table.CRV.state FROM U6 INTO U0
        AND release_complete(qause#, CRV);

ELSE15
IF16 available(RVCI) = FALSE
THEN16
        qause# = 35 AND CHANGE static_table.CRV.state FROM U6 INTO U0
        AND release_complete(qause#, CRV);

ELSE16
check_registered (ATM_add) ⇒ CHECK_REG = (TRUE, FALSE);
IF17 CHECK_REG = FALSE
THEN17
        quase# = 89 AND CHANGE static_table.CRV.state FROM U6 INTO U0
        AND release_complete(qause#, CRV);

ELSE17
{previous checks are okay}

{PRP has to content for a free slot, and identifies itself with its internal address,MAC}
{delivers the internal address to the Interworking layer.}
SAVE static_table.CRV.int_address = internal_address;

select_VPI_VCI ⇒VPI_VCI
IF18 VPI_VCI =xxxx
THEN18
        qause# = 35 AND CHANGE static_table.CRV.state FROM U6 INTO U0
        AND release_complete(qause#);
ELSE18
        SAVE static_table.CRV.VPI_VCI=VPI_VCI;
        MAKE row static_table(static_table.CRV.CRV+) ⇒ static_table.CRV2
        SAVE static_table.CRV2.CRV=CRV2;
        SAVE static_table.CRV2.VPI_VCI=VPI_VCI;
        SAVE static_table.CRV.CRV+=CRV2;
        SAVE static_table.CRV2.CRV+=CRV;
        SAVE static_table.CRV2.originated= internal;
        RVCI_mapping(VPI_VCI) ⇒ RVCI1;        {RVCI1 calling-MSS}

IF19 RVCI1=xxxx
THEN19
        qause# = 45 AND CHANGE static_table.CRV.state FROM U1 INTO U0
        DELETE row static_tabel.CRV2;
        release_complete(qause#,CRV);
ELSE19
        SAVE static_table.CRV.RVCI=RVCI1;
```

```
RVCI_mapping(VPI_VCI) ⇒ RVCI2;                {RVCI2 called-MSS}
IF20 RVCI2=xxxx
THEN20
        qause# = 45 AND CHANGE static_table.CRV.state FROM U1 INTO U0
        DELETE row static_tabel.CRV2;
        release_complete(qause#,CRV);
ELSE20
        SAVE static_table.CRV2.RVCI=RVCI2;
        CHANGE static_table.CRV2.state FROM U0 into U6{Call present}

deduce_delay(CRV);
SAVE static_table.CRV.up=up;
SAVE static_table.CRV.dow=down;
SAVE static_table.CRV2.up=down;
SAVE static_table.CRV2.dow=up;

create_message(CONNECTION IDENTIFIER, VPI_VCI,CRV);
SAVE CONNECTION IDENTIFIER ⇒ inp_buf_CON;
assemble_message(SETUP, inp_buf_CON);
GENERATE statistical data;
SAVE statistical data IN static_data_base
deliver(SETUP, internal, CRV2);
START T303.CRV2;
ON expiry of T303.CRV2 initiate timer_call_releasing;


external:
check_QoS_forw(QoS_forw, QoS_forw_MSS, MEDIAN, SWITCH);
IF1 check_QoS_forw = FALSE
THEN1
        CHANGE static_table.CRV.state FROM U6 INTO U0
        release_complete(qause#, internal, CRV);

ELSE1
check_QoS_backw(QoS_back, QoS_back_MSS, MEDIAN, SWITCH);
IF2 check_QoS_backw = FALSE
THEN2
        CHANGE static_table.CRV.state FROM U6 INTO U0
        release_complete(qause#, internal, CRV);

ELSE2

check_FP_0(forw_peak_0, forw_peak_0_MSS, MEDIAN, SWITCH);
IF3 check_FP_0 = FALSE
THEN3
        CHANGE static_table.CRV.state FROM U6 INTO U0
        release_complete(qause#, internal, CRV);

ELSE3

check_FP_1(forw_peak_1, forw_peak_1_MSS, MEDIAN, SWITCH);
IF4 check_FP_1 = FALSE
THEN4
        CHANGE static_table.CRV.state FROM U6 INTO U0
        release_complete(qause#, internal, CRV);
```

```
ELSE4
check_BP_0(back_peak_0, back _peak_0_MSS, MEDIAN, SWITCH);
IF5 check_BP_0 = FALSE
THEN5
        CHANGE static_table.CRV.state FROM U6 INTO U0
        release_complete(qause#, internal, CRV);

ELSE5
check_BP_1(back_peak_1, back _peak_1_MSS, MEDIAN, SWITCH);
IF6 check_BP_1 = FALSE
THEN6
        CHANGE static_table.CRV.state FROM U6 INTO U0
        release_complete(qause#, internal, CRV);

ELSE6
check_FS_0(forw_sust_0, forw_sust_0_MSS, MEDIAN, SWITCH);
IF7 check_FS_0= FALSE
THEN7
        CHANGE static_table.CRV.state FROM U6 INTO U0
        release_complete(qause#, internal, CRV);

ELSE7
check_FS_1(forw_sust_1 forw_sust_1_MSS, MEDIAN, SWITCH);
IF8 check_FS_1= FALSE
THEN8
        CHANGE static_table.CRV.state FROM U6 INTO U0
        release_complete(qause#, internal, CRV);

ELSE8
check_BS_0(back_sust_0, back_sust_0_MSS, MEDIAN, SWITCH);
IF9 check_BS_0= FALSE
THEN9
        CHANGE static_table.CRV.state FROM U6 INTO U0
        release_complete(qause#, internal, CRV);

ELSE9
check_BS_1(back_sust_1, back_sust_1_MSS, MEDIAN, SWITCH);
IF10 check_BS_1= FALSE
THEN10
        CHANGE static_table.CRV.state FROM U6 INTO U0
        release_complete(qause#, internal, CRV);

ELSE10
check_FM_0(forw_max_0, forw_max_0_MSS, MEDIAN, SWITCH);
IF11 check_FM_0= FALSE
THEN11
        CHANGE static_table.CRV.state FROM U6 INTO U0
        release_complete(qause#, internal, CRV);

ELSE11
check_FM_1(forw_max_1, forw_max_1_MSS, MEDIAN, SWITCH);
IF12 check_FM_1= FALSE
THEN12
        CHANGE static_table.CRV.state FROM U6 INTO U0
        release_complete(qause#, internal, CRV);
```

```
ELSE12
check_BM_0(back_max_0, back_max_0_MSS, MEDIAN, SWITCH);
IF13 check_BM_0 = FALSE
THEN13
        CHANGE static_table.CRV.state FROM U6 INTO U0
        release_complete(qause#, internal, CRV);

ELSE13
check_BM_1(back_max_1, back_max_1_MSS, MEDIAN, SWITCH);
IF14 check_BM_1 = FALSE
THEN14
        CHANGE static_table.CRV.state FROM U6 INTO U0
        release_complete(qause#, internal, CRV);

check_registered(ATM_add)) ⇒ check_registered = (TRUE, FALSE);

IF check_registered= FALSE
THEN quase# = 89 AND CHANGE static_table.CRV.state FROM U6 INTO U0
        AND release_complete(qause#, CRV);
ELSE
{previous checks are okay}

{PRP has to content for a free slot, and identifies itself with its internal address,
MAC}
{delivers the internal address to the Interworking layer.}
SAVE static_table.CRV.int_address = int_address;

deduce_delay(CRV);
{previous checks are okay}
SAVE static_table.CRV.up=up;
SAVE static_table.CRV.dow=down;

RVCI_mapping(VPI_VCI) ⇒ RVCI;                          {for data transport}

IF15 RVCI=xxxx
THEN15
        qause# = 45 AND CHANGE static_table.CRV.state FROM U1 INTO U0
        AND release_complete(qause#,CRV);
ELSE15
        SAVE static_table.CRV.RVCI=RVCI;

GENERATE statistical data;
SAVE statistical data IN static_data_base
deliver(SETUP,external, CRV);
START T303.CRV;
ON expiry of T303.CRV initiate timer_call_releasing;

CALL PROCEEDING :
    CASE of initiator
        internal :
        {CRV=CRV called party}
        CHANGE static_table. static_table.CRV.state FROM U1 INTO U9;

        START T310.CRV
        ON expiry of T310.CRV initiate timer_call_releasing;
        GENERATE statistical data;
        SAVE statistical data IN static_data_base
        deliver(CALL PROCEEDING, internal, CRV);
```

```
                    external :
                    {CRV=CRV called party}
                    START T310.CRV
                    ON expiry of T310.CRV initiate timer_call_releasing;
                    GENERATE statistical data;
                    SAVE statistical data IN static_data_base
                    deliver(CALL PROCEEDING, external,CRV);

CONNECT:
          CASE of initiator
                    internal :
                    CHANGE static_table. static_table.CRV.state U9 into U10    {Active state}
                    GENERATE statistical data;
                    SAVE statistical data IN static_data_base
                    deliver(CONNECT, internal, CRV);
                    START T313.CRV;
                    ON expiry of T313.CRV initiate timer_call_releasing;

                    external :
                    CHANGE static_table. static_table.CRV.state U3 into U8     {Connect request state}
                    START T313.CRV;
                    GENERATE statistical data;
                    SAVE statistical data IN static_data_base
                    deliver(CONNECT, internal, CRV);
                    ON expiry of T313.CRV initiate timer_call_releasing;

CONNECT ACKNOWLEDGE :
          CASE of initiater
                    internal :
                    {CRV=CRV called party}
                    GENERATE statistical data;
                    SAVE statistical data IN static_data_base
                    deliver(CONNECT ACKNOWLEDGE, internal, CRV};

                    external :
                    GENERATE statistical data;
                    SAVE statistical data IN static_data_base
                    deliver(CONNECT ACKNOWLEDGE, external, CRV};

RELEASE :
          CASE of initiator
                    internal :
                    GENERATE statistical data;
                    SAVE statistical data IN static_data_base
                    deliver(RELEASE, internal, CRV);
                    START T308.CRV;
                    ON expiry of T308.CRV initiate timer_ call_ releasing;

                    external :

                    GENERATE statistical data;
                    SAVE statistical data IN static_data_base
                    deliver(RELEASE, external, CRV);
                    START T308.CRV;
                    ON expiry of T308.CRV initiate timer_call_releasing;
```

```
RELEASE COMPLETE :
        CASE of initiater
                internal :

                GENERATE statistical data;
                SAVE statistical data IN static_data_base
                deliver(RELEASE COMPLETE, internal, CRV);
                DELETE row static_table(CRV);

                external :

                GENERATE statistical data;
                SAVE statistical data IN static_data_base
                deliver(RELEASE COMPLETE,external, CRV);
                DELETE row static_table(CRV);
```

active;
{}
INPUT : timer;
OUTPUT:IF timer active THEN TRUE ELSE FALSE;
END {*active*}

assemble_message
{}
INPUT : message type;
OUTPUT : message information element;
END {*assemble_message*}

available
{}
INPUT : RVCI or VPI_VCI;
OUTPUT : IF AVAILABLE THEN TRUE ELSE FALSE;
END {*available*}

check_registered
{}
INPUT : ATM_addres;
OUTPUT : IF ATM_address registered in MSS THEN TRUE ELSE FALSE;
END {*check_registered*}

create_message
{}
INPUT : information element name, parameter(s) VPI_VCI. identifing the value of the specific message field;
OUTPUT : information element;
END {*create_message*}

deduce_delay
{SETUP}
INPUT :CRV
OUTPUT: maximum uplink(up) and downlink(down) delay;
END { *deduce_delay*)

deliver
{CONNECT, CONNECT ACKNOWLEDGE, RELEASE COMPLETE, SETUP}
INPUT : message type, direction, CRV;
OUTPUT : message delivered to correct layer;
END {*deliver*}

error_control_of_message
{protocol discriminator error}
IF inp_buf_PD ≠ 00001001
THEN DELETE frame IN inp_buf
ELSE
{message to short}
calculate_message_length(frame) ⇒ ML_calc
IF ML_calc ≠ ML
THEN DELETE frame IN inp_buf
ELSE
{call reference procedural error}
IF length ≠ 3
THEN DELETE frame IN inp_buf
ELSE
{call reference procedural error}
IF MT ≠ (SETUP or RELEASE COMPLETE or STATUS or STATUS ENQUIRY) and static_table.CRV = empty
THEN *release_complete*(cause#81, internal/externalCRV);
ELSE
{call reference procedural error}
IF MT = RELEASE COMPLETE and static_table.CRV.state ≠U10 or U1 or U3 or U6 or U9
THEN DELETE frame IN inp_buf
ELSE
{call reference procedural error}
IF MT = SETUP and static_table.CRV.state = U10 or U1 or U3 or U6 or U9
THEN DELETE frame IN inp_buf
ELSE
{global call reference}
IF MT ≠ (RESTART or RESTART ACKNOWLEDGE or STATUS) and CRV = 0)
THEN STATUS(cause#81) returned;
ELSE
{call reference procedural error}
IF MT = STATUS and static_table.CRV.state = U10 or U1 or U3 or U6 or U9
THEN *release_complete*(cause#101, internal/external, CRV)
ELSE
{message type or message sequence error}
IF static_table.CRV.state ≠ U0 and MT≠ (RELEASE or RELEASE COMPLETE) and (static_tabel.CRV.state not compatible with MT)
THEN STATUS(cause#97 or #101);
ELSE
{mandatory information element missing}
IF MT ≠ (SETUP or RELEASE or RELEASE COMPLETE) and information element missing
THEN
DELETE frame IN inp_buf;
create(STATUS, cause#96);
deliver(STATUS, internal/external, CRV);
ELSE
IF MT = SETUP and information element missing
THEN
create(RELEASE COMPLETE,cause#96);
deliver(RELEASE COMPLETE, internal/extenal, CRV);
ELSE
IF MT = RELEASE and inp_buf_CI = empty
THEN USE cause#96;
ELSE
IF MT = RELEASE COMPLETE and inp_buf_CI = empty
THEN USE cause#31;
ELSE

```
IF MT ≠ (SETUP or RELEASE or RELEASE COMPLETE) and information element with invalid content
THEN
create(STATUS, cause#100);
deliver(STATUS, internal/external, CRV);
ELSE
IF MT = SETUP and information element with invalid content
THEN
create(RELEASE COMPLETE,cause#100);
deliver(RELEASE COMPLETE, internal/extenal, CRV);
ELSE
IF MT = (RELEASE or RELEASE COMPLETE) and invalid(inp_buf_CI)
THEN
USE cause#31
create(RELEASE COMPLETE,cause#100);
deliver(RELEASE COMPLETE, internal/extenal, CRV);
ELSE
IF unrecorgized information element THEN
        IF MT ≠ (RELEASE or RELEASE COMPLETE)
        THEN
                create(STATUS, cause#99);
                deliver(STATUS, internal/external, CRV);

        IF MT = RELEASE
        THEN
                create(STATUS, cause#99);
                deliver(STATUS, internal/external, CRV);

        IF MT = RELEASE COMPLETE
        THEN {no action}
ELSE
IF unexpected recorgnized information element THEN
        IF MT ≠ (RELEASE or RELEASE COMPLETE)
        THEN
                create(STATUS, cause#99);
                deliver(STATUS, internal/external, CRV);

        IF MT = RELEASE
        THEN
                create(RELEASE COMPLETE,cause#99);
                deliver(RELEASE COMPLETE, internal/extenal, CRV);

        IF MT = RELEASE COMPLETE
        THEN {no action}
ELSE CONTINUE
END{error_control_of_message}
```

```
release_complete(quase#,CRV);
{SETUP}
USE inp_buf_PD;
USE inp_buf_CR;
CHANGE inp_buf_MT(MT = 0101 1010);        {RELEASE COMPLETE}
location = 0101;                          {private network serving the remote user}
CREATE cause(cause#, location);
CALCULATE Message length ⇒ ML;
CREATE message length;
ASSEMBLE RELEASE COMPLETE;
GENERATE statistical data;
SAVE statistical data IN static_data_base
deliver(release complete,internal);
DELETE row static_table.CRV;
END {release_complete }

RVCI_mapping
{SETUP}
INPUT : VPI_VCI;
OUTPUT : RVCI;
END {RVCI_mapping}

select_VPI_VCI
{SETUP}
END { select_VPI_VCI }

timer_call_releasing;
{SETUP}
END {timer_call_releasing};


check_QoS_forw
{SETUP}
        CHECK_QoS(QoS_forw, QoS_forw_MSS, MEDIAN, SWITCH)
        ⇒ CHECK_QoS = (TRUE, FALSE);

        IF CHECK_QoS = FALSE
        THEN RETURN (qause# = 49 AND CHECK_QoS = FALSE)
        ELSE RETURN CHECK_QoS = TRUE
END {check_QoS}


check_QoS_backw
{SETUP}
        CHECK_QoS(QoS_backw, QoS_backw_MSS, MEDIAN, SWITCH)
        ⇒ CHECK_QoS = (TRUE, FALSE);

        IF CHECK_QoS = FALSE
        THEN RETURN (qause# = 49 AND CHECK_QoS = FALSE)
        ELSE RETURN CHECK_QoS = TRUE
END {check_QoS_backw}
```

```
check_FP_0
{SETUP}
        CHECK_FP_0(forw_peak_0, forw_peak_0_MSS, MEDIAN, SWITCH)
        ⇒ CHECK_FP_0 =  (TRUE, FALSE);

        IF CHECK_FP_0 = FALSE
        THEN RETURN (qause# = 37 OR qause#=73)
        ELSE RETURN CHECK_FP_0 = TRUE;
END {check_FP_0}


check_FP_1
{SETUP}
        CHECK_FP_1(forw_peak_1, forw_peak_1_MSS, MEDIAN, SWITCH)
        ⇒ CHECK_FP_1 =  (TRUE, FALSE);

        IF CHECK_FP_1 = FALSE
        THEN RETURN (qause# = 37 OR qause#=73)
        ELSE RETURN CHECK_FP_1 = TRUE;
END {check_FP_1}

check_BP_0
{SETUP}
CHECK_BP_0(back_peak_0, back _peak_0_MSS, MEDIAN, SWITCH)
        ⇒ CHECK_BP_0 =  (TRUE, FALSE);

        IF CHECK_BP_0 = FALSE
        THEN RETURN (qause# = 37 OR qause#=73)
        ELSE RETURN CHECK_BP_0 = TRUE;
END {check_BP_0}


check_BP_1
{SETUP}
CHECK_BP_1(back_peak_1, back _peak_1_MSS, MEDIAN, SWITCH)
        ⇒ CHECK_BP_1 =  (TRUE, FALSE);

        IF CHECK_BP_1 = FALSE
        THEN RETURN (qause# = 37 OR qause#=73)
        ELSE RETURN CHECK_BP_1 = TRUE;
END {check_BP_1}


check_FS_0
{SETUP}
CHECK_FS_0((forw_sust_0, forw_sust_0_MSS, MEDIAN, SWITCH)
        ⇒ CHECK_FS_0 =  (TRUE, FALSE);

        IF CHECK_FS_0 = FALSE
        THEN RETURN (qause# = 37 OR qause#=73)
        ELSE RETURN CHECK_FS_0 = TRUE;
END {check_FS_0}
```

```
check_FS_1
{SETUP}
CHECK_FS_1(forw_sust_1, forw_sust_1_MSS, MEDIAN, SWITCH)
        ⇒ CHECK_FS_1 =  (TRUE, FALSE);

        IF CHECK_FS_1 = FALSE
        THEN RETURN (qause# = 37 OR qause#=73)
        ELSE RETURN CHECK_FS_1 = TRUE;
END {check_FS_1}


check_BS_0
(SETUP}
CHECK_BS_0(back_sust_0, back_sust_0_MSS, MEDIAN, SWITCH)
        ⇒ CHECK_BS_0 =  (TRUE, FALSE);

        IF CHECK_BS_0 = FALSE
        THEN RETURN (qause# = 37 OR qause#=73)
        ELSE RETURN CHECK_BS_0 = TRUE;
END {check_BS_0}


check_BS_1
(SETUP}
CHECK_BS_1(back_sust_1, back_sust_1_MSS, MEDIAN, SWITCH)
        ⇒ CHECK_BS_1 =  (TRUE, FALSE);

        IF CHECK_BS_1 = FALSE
        THEN RETURN (qause# = 37 OR qause#=73)
        ELSE RETURN CHECK_BS_1 = TRUE;
END {check_BS_1}


check_FM_0
(SETUP}
CHECK_FM_0(forw_max_0, forw_max_0_MSS, MEDIAN, SWITCH)
        ⇒ CHECK_FM_0 =  (TRUE, FALSE);

        IF CHECK_FM_0 = FALSE
        THEN RETURN (qause# = 37 OR qause#=73)
        ELSE RETURN CHECK_FM_0 = TRUE;
END {check_FM_0}


check_FM_1
(SETUP}
CHECK_FM_1(forw_max_1, forw_max_1_MSS, MEDIAN, SWITCH)
        ⇒ CHECK_FM_1 =  (TRUE, FALSE);

        IF CHECK_FM_1 = FALSE
        THEN RETURN (qause# = 37 OR qause#=73)
        ELSE RETURN CHECK_FM_1 = TRUE;
END {check_FM_1}
```

check_BM_0
(SETUP}
CHECK_BM_0(back_max_0, back_max_0_MSS, MEDIAN, SWITCH)
⇒ CHECK_BM_0 = (TRUE, FALSE);

IF CHECK_BM_0 = FALSE
THEN RETURN (qause# = 37 OR qause#=73)
ELSE RETURN CHECK_BM_0 = TRUE;
END {*check_BM_0*}

check_BM_1
(SETUP}
CHECK_BM_1(back_max_1, back_max_1_MSS, MEDIAN, SWITCH)
⇒ CHECK_BM_1 = (TRUE, FALSE);

IF CHECK_BM_1 = FALSE
THEN RETURN (qause# = 37 OR qause#=73)
ELSE RETURN CHECK_BM_1 = TRUE;
END {*check_BM_1*}

Appendix G

Protocol description ATM switch to MSS

```
RECEIVE signalling message  FROM SAAL                 {read/buffer received message }
BUFFER frame IN inp_buf;                              {into input buffer}

DEDUCE frame information elements FROM inp_buf:
        Protocol discriminator    ⇒ inp_buf_PD;
        Call reference            ⇒ inp_buf_CR;
        Message type              ⇒ inp_buf_MT;
        Message length            ⇒ inp_buf_ML;
END {DEDUCE}

DEDUCE info in information elements FROM:

        Call reference: inp_buf_CR
                length            ⇒ length;
                flag              ⇒ F;
                call reference value⇒ CRV;

        Message type : inp_buf_MT
                message type       ⇒ MT;

        Message length : inp_buf_ML
                message length     ⇒ ML;
END {DEDUCE}

CASE of MT

        SETUP :
                MAKE row in static_table: static_table.CRV;          {create row in static table}
                SAVE static_table.CRV.CRV=CRV;                       {CRV saved in static table}
                CHANGE static_table.CRV.state FROM U0 INTO U6;       {state is Call present}
                SAVE static_table.CRV.originated=externall;          {external call/conn. request}

                DEDUCE frame information elements :
                        ATM traffic descriptor          ⇒ inp_buf_ATD;
                        Broadband Bearer Capability     ⇒ inp_buf_BBC;
                        Called party Number             ⇒ inp_buf_CPN;
                        Connection Identifier           ⇒ inp_buf_CI;
                        Quality of  Service parameter   ⇒ inp_buf_QOS;

                error_control_of_message;

                DEDUCE info IN information elements FROM:
                        ATM traffic descriptor ⇒ inp_buf_ATD
                                {if available}
                                forward peak cell rate with CLP=0                ⇒ forw_peak_0;
                                forward peak cell rate with CLP=1                ⇒ forw_peak_1;
                                backward peak cell rate with CLP=0         ⇒ back_peak_0;
                                backward peak cell rate with CLP=1         ⇒ back_peak_1;
                                forward sustainable cell rate with CLP=0   ⇒ forw_sust_0;
                                forward sustainable cell rate with CLP=1   ⇒ forw_sust_1;
                                backward sustainable cell rate with CLP=0  ⇒ back_sust_0;
                                backward sustainable cell rate with CLP=1  ⇒ back_sust_1;
                                forward maximum burst size CLP=0           ⇒ forw_max_0;
                                forward maximum burst size CLP=1           ⇒ forw_max_1;
                                backward maximum burst size CLP=0          ⇒ back_max_0;
```

```
                backward maximum burst size CLP=1      ⇒ back_max_1;
                SAVE info in static data base;

        Broadband bearer capability ⇒ inp_buf_BBC;
                bearer class                ⇒ bear_class;
                traffic type                ⇒ traffic_type;
                timing requirements         ⇒ timing_requir;
                susceptibility to clipping  ⇒ susc_clip;
                user plane connection       ⇒ user_con;
                SAVE info in static data base;

        Called party number ⇒ inp_buf_CPN;
                type of number              ⇒ typ_numb;
                Addressing/number plan      ⇒ add_numb;
                ATM endsystem address       ⇒ ATM_add;

        Connection identifier ⇒ inp_buf_CON;
                VPI/VCI                     ⇒ VPI_VCI;
                SAVE static_table.CRV.VPI_VCI = VPI_VCI:   {VPI/VCI mapped to
                CRV}

        Quality of service ⇒ inp_buf_QoS;
                QoS class forward           ⇒ QoS_forw;
                QoS class backward          ⇒ QoS_back;
                SAVE info in static data base;

        CONTINUE WITH message_handling(SETUP, internal, CRV);

END{SETUP}


CONNECT :
        STOP T310.CRV

        error_control_of_ message;

        CHANGE static_table.CRV.state FROM U3 INTO U10;     {state is Active}

        message_handling(CONNECT, internal, CRV);

END{CONNECT}


CONNECT ACKNOWLEDGE :
        error_control_ of_message;

        CHANGE static_table.CRV.state FROM U8 INTO U10;     {state is Active}
        STOP T313.CRV;

        message_handling(CONNECT ACKNOWLEDGE, internal, CRV);

END{CONNECT ACKNOWLEDGE}
```

```
CALL PROCEEDING :
        STOP T303.CRV;

        DEDUCE frame information elements :
                Connection Identifier ⇒ inp_buf_CI;

        error_control_of_message;

        DEDUCE info FROM information elements:
                Connection identifier ⇒ inp_buf_CON;
                        VPI/VCI ⇒ VPI_VCI;
                        SAVE static_table.CRV.VPI_VCI = VPI_VCI: {VPI/VCI mapped to CRV}
        CHANGE static_table.CRV.state FROM U1 INTO U3;          {state is Outgoing call }
                                                                {proceeding}
        message_handling(CALL PROCEEDING, internal, CRV);

END {CALL PROCEEDING}


RELEASE
        DEDUCE frame information elements :
                Cause ⇒ inp_buf_CI;

        error_control_of_message;

        CHANGE static_table.CRV.state from U10 INTO U12;     {state is Release indication}

        message_handling(RELEASE, internal, CRV);

END {RELEASE}


RELEASE COMPLETE
        IF active(T308.CRV)
        THEN STOP T308.CRV              {stop timer T308 or T310}
        ELSE STOP T310.CRV;

        DEDUCE frame information elements :
                Cause ⇒ inp_buf_CI;

        error_control_ of_message;

        CHANGE static_table.CRV.state INTO U0;                  {state is Null}

        message_ handling(RELEASE COMPLETE,internal, CRV);

END {RELEASE COMPLETE}

END {CASE}
```

message_handling
{SETUP,CALL PROCEEDING, CONNECT, CONNECT ACKNOWLEDGE, RELEASE COMPLETE, RELEASE}

```
CASE of MT
    SETUP :
        CASE of initiator
            internal :
            check_QoS_forw(QoS_forw, QoS_forw_MSS, MEDIAN, SWITCH);
            IF check_QoS_forw = FALSE
            THEN
            CHANGE static_table.CRV.state FROM U6 INTO U0
            release_complete(qause#, internal, CRV);

            ELSE
            check_QoS_backw(QoS_back, QoS_back_MSS, MEDIAN, SWITCH);
            IF check_QoS_backw = FALSE
            THEN
            CHANGE static_table.CRV.state FROM U6 INTO U0
            release_complete(qause#, internal, CRV);

            ELSE

            check_FP_0(forw_peak_0, forw_peak_0_MSS, MEDIAN, SWITCH);
            IF check_FP_0 = FALSE
            THEN
            CHANGE static_table.CRV.state FROM U6 INTO U0
            release_complete(qause#, internal, CRV);

            ELSE

            check_FP_1(forw_peak_1, forw_peak_1_MSS, MEDIAN, SWITCH);
            IF check_FP_1 = FALSE
            THEN
            CHANGE static_table.CRV.state FROM U6 INTO U0
            release_complete(qause#, internal, CRV);

            ELSE

            check_BP_0(back_peak_0, back _peak_0_MSS, MEDIAN, SWITCH);
            IF check_BP_0 = FALSE
            THEN
            CHANGE static_table.CRV.state FROM U6 INTO U0
            release_complete(qause#, internal, CRV);

            ELSE

            check_BP_1(back_peak_1, back _peak_1_MSS, MEDIAN, SWITCH);
            IF check_BP_1 = FALSE
            THEN
            CHANGE static_table.CRV.state FROM U6 INTO U0
            release_complete(qause#, internal, CRV);

            ELSE

            check_FS_0(forw_sust_0, forw_sust_0_MSS, MEDIAN, SWITCH);
            IF check_FS_0= FALSE
            THEN
```

CHANGE static_table.CRV.state FROM U6 INTO U0
release_complete(qause#, internal, CRV);

ELSE

check_FS_1(forw_sust_1 forw_sust_1_MSS, MEDIAN, SWITCH);
IF *check_FS_1*= FALSE
THEN
CHANGE static_table.CRV.state FROM U6 INTO U0
release_complete(qause#, internal, CRV);

ELSE

check_BS_0(back_sust_0, back_sust_0_MSS, MEDIAN, SWITCH);
IF *check_BS_0*= FALSE
THEN
CHANGE static_table.CRV.state FROM U6 INTO U0
release_complete(qause#, internal, CRV);

ELSE

check_BS_1(back_sust_1, back_sust_1_MSS, MEDIAN, SWITCH);
IF *check_BS_1*= FALSE
THEN
CHANGE static_table.CRV.state FROM U6 INTO U0
release_complete(qause#, internal, CRV);

ELSE

check_FM_0(forw_max_0, forw_max_0_MSS, MEDIAN, SWITCH);
IF *check_FM_0*= FALSE
THEN
CHANGE static_table.CRV.state FROM U6 INTO U0
release_complete(qause#, internal, CRV);

ELSE

check_FM_1(forw_max_1, forw_max_1_MSS, MEDIAN, SWITCH);
IF *check_FM_1*= FALSE
THEN
CHANGE static_table.CRV.state FROM U6 INTO U0
release_complete(qause#, internal, CRV);

ELSE

check_BM_0(back_max_0, back_max_0_MSS, MEDIAN, SWITCH);
IF *check_BM_0* = FALSE
THEN
CHANGE static_table.CRV.state FROM U6 INTO U0
release_complete(qause#, internal, CRV);

ELSE

check_BM_1(back_max_1, back_max_1_MSS, MEDIAN, SWITCH);
IF *check_BM_1* = FALSE
THEN
CHANGE static_table.CRV.state FROM U6 INTO U0

```
            release_complete(qause#, internal, CRV);
            ELSE

            IF available(VPI_VCI) = FALSE
            THEN qause# = 35 AND CHANGE static_table.CRV.state FROM U6 INTO U0
                    AND release_complete(qause#, CRV);

            ELSE

            IF available(RVCI) = FALSE
            THEN qause# = 35 AND CHANGE static_table.CRV.state FROM U6 INTO U0
                    AND release_complete(qause#, CRV);

            ELSE

            check_registered(ATM_add)) ⇒ check_registered = (TRUE, FALSE);

            IF check_registered= FALSE
            THEN quase# = 89 AND CHANGE static_table.CRV.state FROM U6 INTO U0
                    AND release_complete(qause#, CRV);

            ELSE
            {previous checks are okay}

            internal_mapping(ATM-address) ⇒ int_address     {large ATM-address mapped}
            SAVE static_table.CRV.int_address = int_address  {on internal address}
            RVCI_mapping(VPI_VCI) ⇒ RVCI;                    {choosing an unused RVCI}
            SAVE static_table.CRV.RVCI = RVCI;               {RVCI mapped to VPI/VCI}
            deduce_delay(CRV);                               {calculates the allowable up-}
                                                             {and downlink  delays}
            SAVE static_table.CRV.up = up;
            SAVE static_table.CRV.down = down;
            GENERATE statistical data;
            SAVE statistical data IN static_data_base
            deliver(SETUP, internal, CRV);
            START T303.CRV
            ON expiry of T303.CRV initiate timer_call_releasing;

END {SETUP}


CALL PROCEEDING :
        CASE of initiator
            internal :
            START T310.CRV
            ON expiry of T310.CRV initiate timer_call_releasing;
            GENERATE statistical data;
            SAVE statistical data IN static_data_base
            deliver(CALL PROCEEDING, internal, CRV);

END {CALL PROCEEDING}
```

```
CONNECT :
        CASE of initiator
                internal :
                GENERATE statistical data;
                SAVE statistical data IN static_data_base
                deliver(CONNECT, internal, CRV);
                START Txxx.CRV;                          {timer for receipt of CON.ACK}
                ON expiry of Txxx.CRV initiate timer_call_releasing;

END {CONNECT}


CONNECT ACKNOWLEDGE :
        CASE of initiator
                internal :
                GENERATE statistical data;
                SAVE statistical data IN static_data_base
                deliver(CONNECT ACKNOWLEDGE, internal, CRV);

END {CONNECT ACKNOWLEDGE}


RELEASE :
        CASE of initiator
                internal :
                START T308.CRV;
                ON expiry of T308.CRV initiate timer_call_releasing;
                GENERATE statistical data;
                SAVE statistical data IN static_data_base
                deliver(RELEASE, internal, CRV);

END {RELEASE}


RELEASE COMPLETE :
        CASE of initiator
                internal :
                IF MSS_initiated_call_release = FALSE
                THEN
                        GENERATE statistical data;
                        SAVE statistical data IN static_data_base
                        deliver(RELEASE COMPLETE, internal, CRV);

                DELETE row static_table.CRV;

END {RELEASE COMPLETE}

END {MESSAGE HANDLING}
```

active;
{}
INPUT : timer;
OUTPUT: TRUE IF timer ACTIVE ELSE FALSE;
END {*active*}

assemble_message
{}
INPUT : message type;
OUTPUT : message information element;
END { *assemble_message*}

available
{}
INPUT : RVCI or VPI_VCI;
OUTPUT : IF AVAILABLE THEN TRUE ELSE FALSE;
END {*available*}

check_registered
{}
INPUT : ATM_addres;
OUTPUT : IF ATM_address registered in MSS THEN TRUE ELSE FALSE;
END {*check_registered*}

create_message
{}
INPUT : information element name, parameter(s) VPI_VCI. identifing the value of the specific message field;
OUTPUT : information element;
END {*create_message*}

deliver
{CONNECT, CONNECT ACKNOWLEDGE, RELEASE COMPLETE}
INPUT : message type, direction, CRV;
OUTPUT : message delivered to correct layer;
END {*deliver*}

deduce_delay
{SETUP}
INPUT : CRV;
OUTPUT: maximum uplink(up) and downlink(down) delay;
END { *deduce_delay*)

error_control_of_message
{protocol discriminator error}
IF inp_buf_PD ≠ 00001001
THEN DELETE frame IN inp_buf
ELSE
{message to short}
calculate_message_length(frame) ⇒ ML_calc
IF ML_calc ≠ ML
THEN DELETE frame IN inp_buf

ELSE
{call reference procedural error}
IF length ≠ 3
THEN DELETE frame IN inp_buf
ELSE
{call reference procedural error}
IF MT ≠ (SETUP or RELEASE COMPLETE or STATUS or STATUS ENQUIRY) and static_table.CRV = empty
THEN *release_complete*(cause#81, internal/externalCRV);
ELSE
{call reference procedural error}
IF MT = RELEASE COMPLETE and static_table.CRV.state ≠U10 or U1 or U3 or U6 or U9
THEN DELETE frame IN inp_buf
ELSE
{call reference procedural error}
IF MT = SETUP and static_table.CRV.state = U10 or U1 or U3 or U6 or U9
THEN DELETE frame IN inp_buf
ELSE
{global call reference}
IF MT ≠ (RESTART or RESTART ACKNOWLEDGE or STATUS) and CRV = 0)
THEN STATUS(cause#81) returned;
ELSE
{call reference procedural error}
IF MT = STATUS and static_table.CRV.state = U10 or U1 or U3 or U6 or U9
THEN *release_complete*(cause#101, internal/external, CRV)
ELSE
{message type or message sequence error}
IF static_table.CRV.state ≠ U0 and MT≠ (RELEASE or RELEASE COMPLETE) and (static_tabel.CRV.state not compatible with MT)
THEN STATUS(cause#97 or #101);
ELSE
{mandatory information element missing}
IF MT ≠ (SETUP or RELEASE or RELEASE COMPLETE) and information element missing
THEN
DELETE frame IN inp_buf;
create(STATUS, cause#96);
deliver(STATUS, internal/external, CRV);
ELSE
IF MT = SETUP and information element missing
THEN
create(RELEASE COMPLETE,cause#96);
deliver(RELEASE COMPLETE, internal/extenal, CRV);
ELSE
IF MT = RELEASE and inp_buf_CI = empty
THEN USE cause#96;
ELSE
IF MT = RELEASE COMPLETE and inp_buf_CI = empty
THEN USE cause#31;
ELSE
IF MT ≠ (SETUP or RELEASE or RELEASE COMPLETE) and information element with invalid content
THEN
create(STATUS, cause#100);
deliver(STATUS, internal/external, CRV);
ELSE
IF MT = SETUP and information element with invalid content
THEN
create(RELEASE COMPLETE,cause#100);
deliver(RELEASE COMPLETE, internal/extenal, CRV);

```
ELSE
IF MT = (RELEASE or RELEASE COMPLETE) and invalid(inp_buf_CI)
THEN
USE cause#31
create(RELEASE COMPLETE,cause#100);
deliver(RELEASE COMPLETE, internal/extenal, CRV);
ELSE
IF unrecorgized information element THEN
        IF MT ≠ (RELEASE or RELEASE COMPLETE)
        THEN
                create(STATUS, cause#99);
                deliver(STATUS, internal/external, CRV);

        IF MT = RELEASE
        THEN
                create(STATUS, cause#99);
                deliver(STATUS, internal/external, CRV);

        IF MT = RELEASE COMPLETE
        THEN {no action}
ELSE
IF unexpected recorgnized information element THEN
        IF MT ≠ (RELEASE or RELEASE COMPLETE)
        THEN
                create(STATUS, cause#99);
                deliver(STATUS, internal/external, CRV);

        IF MT = RELEASE
        THEN
                create(RELEASE COMPLETE,cause#99);
                deliver(RELEASE COMPLETE, internal/extenal, CRV);

        IF MT = RELEASE COMPLETE
        THEN {no action}
ELSE CONTINUE
END{error_control_of_message}


internal_mapping
{}
INPUT : ATM_address;
OUTPUT : corresponding internal address (int_address);
END { internal_mapping}


release_complete
{SETUP}
USE inp_buf_PD;
USE inp_buf_CR;
CHANGE inp_buf_MT(MT = 0101 1010);
location = 0101;                                {private network serving the remote user}
create_message(Cause, cause#, location, CRV);
CALCULATE Message length ⇒ ML;
create_message(message length, ML, CRV);
ASSEMBLE RELEASE COMPLETE;
GENERATE statistical data;
SAVE statistical data IN static_data_base
deliver(release complete,external);
```

DELETE row static_table.CRV;
END { *release_complete* }
RVCI_mapping
{SETUP}
INPUT : VPI_VCI;
OUTPUT : RVCI;
END {*RVCI_mapping*}

timer_call_releasing
{CONNECT ACK}
END { *timer_call_releasing* }

check_QoS_forw
{SETUP}

```
        CHECK_QoS(QoS_forw, QoS_forw_MSS, MEDIAN, SWITCH)
        ⇒ CHECK_QoS = (TRUE, FALSE);

        IF CHECK_QoS = FALSE
        THEN RETURN (qause# = 49 AND CHECK_QoS = FALSE)
        ELSE RETURN CHECK_QoS = TRUE
```

END {*check_QoS}*

check_QoS_backw
{SETUP}

```
        CHECK_QoS(QoS_backw, QoS_backw_MSS, MEDIAN, SWITCH)
        ⇒ CHECK_QoS = (TRUE, FALSE);

        IF CHECK_QoS = FALSE
        THEN RETURN (qause# = 49 AND CHECK_QoS = FALSE)
        ELSE RETURN CHECK_QoS = TRUE
```

END {*check_QoS_backw}*

check_FP_0
{SETUP}

```
        CHECK_FP_0(forw_peak_0, forw_peak_0_MSS, MEDIAN, SWITCH)
        ⇒ CHECK_FP_0 =  (TRUE, FALSE);

        IF CHECK_FP_0 = FALSE
        THEN RETURN (qause# = 37 OR qause#=73)
        ELSE RETURN CHECK_FP_0 = TRUE;
```

END {*check_FP_0*}

check_FP_1
{SETUP}

```
        CHECK_FP_1(forw_peak_1, forw_peak_1_MSS, MEDIAN, SWITCH)
        ⇒ CHECK_FP_1 =  (TRUE, FALSE);

        IF CHECK_FP_1 = FALSE
        THEN RETURN (qause# = 37 OR qause#=73)
        ELSE RETURN CHECK_FP_1 = TRUE;
```

END {*check_FP_1*}
check_BP_0
{SETUP}
CHECK_BP_0(back_peak_0, back _peak_0_MSS, MEDIAN, SWITCH)
⇒ CHECK_BP_0 = (TRUE, FALSE);

IF CHECK_BP_0 = FALSE
THEN RETURN (qause# = 37 OR qause#=73)
ELSE RETURN CHECK_BP_0 = TRUE;

END {*check_BP_0*}

check_BP_1
{SETUP}
CHECK_BP_1(back_peak_1, back _peak_1_MSS, MEDIAN, SWITCH)
⇒ CHECK_BP_1 = (TRUE, FALSE);

IF CHECK_BP_1 = FALSE
THEN RETURN (qause# = 37 OR qause#=73)
ELSE RETURN CHECK_BP_1 = TRUE;

END {*check_BP_1*}

check_FS_0
{SETUP}
CHECK_FS_0((forw_sust_0, forw_sust_0_MSS, MEDIAN, SWITCH)
⇒ CHECK_FS_0 = (TRUE, FALSE);

IF CHECK_FS_0 = FALSE
THEN RETURN (qause# = 37 OR qause#=73)
ELSE RETURN CHECK_FS_0 = TRUE;

END {*check_FS_0*}

check_FS_1
{SETUP}
CHECK_FS_1(forw_sust_1, forw_sust_1_MSS, MEDIAN, SWITCH)
⇒ CHECK_FS_1 = (TRUE, FALSE);

IF CHECK_FS_1 = FALSE
THEN RETURN (qause# = 37 OR qause#=73)
ELSE RETURN CHECK_FS_1 = TRUE;

END {*check_FS_1*}

check_BS_0
(SETUP)
CHECK_BS_0(back_sust_0, back_sust_0_MSS, MEDIAN, SWITCH)
⇒ CHECK_BS_0 = (TRUE, FALSE);

IF CHECK_BS_0 = FALSE
THEN RETURN (qause# = 37 OR qause#=73)
ELSE RETURN CHECK_BS_0 = TRUE;

END {*check_BS_0*}

check_BS_1
{SETUP}
CHECK_BS_1(back_sust_1, back_sust_1_MSS, MEDIAN, SWITCH)
⇒ CHECK_BS_1 = (TRUE, FALSE);

IF CHECK_BS_1 = FALSE
THEN RETURN (qause# = 37 OR qause#=73)
ELSE RETURN CHECK_BS_1 = TRUE;

END {*check_BS_1*}

check_FM_0
{SETUP}
CHECK_FM_0(forw_max_0, forw_max_0_MSS, MEDIAN, SWITCH)
⇒ CHECK_FM_0 = (TRUE, FALSE);

IF CHECK_FM_0 = FALSE
THEN RETURN (qause# = 37 OR qause#=73)
ELSE RETURN CHECK_FM_0 = TRUE;

END {*check_FM_0*}

check_FM_1
{SETUP}
CHECK_FM_1(forw_max_1, forw_max_1_MSS, MEDIAN, SWITCH)
⇒ CHECK_FM_1 = (TRUE, FALSE);

IF CHECK_FM_1 = FALSE
THEN RETURN (qause# = 37 OR qause#=73)
ELSE RETURN CHECK_FM_1 = TRUE;

END {*check_FM_1*}

check_BM_0
{SETUP}
CHECK_BM_0(back_max_0, back_max_0_MSS, MEDIAN, SWITCH)
⇒ CHECK_BM_0 = (TRUE, FALSE);

IF CHECK_BM_0 = FALSE
THEN RETURN (qause# = 37 OR qause#=73)
ELSE RETURN CHECK_BM_0 = TRUE;

END {*check_BM_0*}

ELSE

check_BM_1
{SETUP}
CHECK_BM_1(back_max_1, back_max_1_MSS, MEDIAN, SWITCH)
⇒ CHECK_BM_1 = (TRUE, FALSE);

IF CHECK_BM_1 = FALSE
THEN RETURN (qause# = 37 OR qause#=73)
ELSE RETURN CHECK_BM_1 = TRUE;

Appendix H

Description signalling messages

Used messages

The messages that will be used are composed of several information elements. Every message consits, at least, of the following four information elements:

- protocol discriminator;
- call reference;
- message type;
- message length.

Depending on the used message that will be generated additional information elements can be added.

The protocol discriminator

The purpose of the protocol discriminator is to distinguish messages for user-network call control from other messges. The protocol discrimiator is the first part of every message. The protocol discriminator is coded according to Table 1.

Table 1: protocol discriminator

Bits **8765 4321**	**Meaning**
0000 0000 through	assigned in Q.931, not available for use in the
0000 0111	message protocol discrminator
0000 1000	Q.931 user-network call control messages
0000 1001	Q.93B user-network call control messages
0001 0000 through	reserved for other network layer or layer 3 protocols,
0011 1111	including recommendation X.25
0001 0000 through	national use
0001 1111	
0101 0000 through	reserved for other network layer or layer 3 protocols,
1111 1110	including recommendation X.25

The call reference

The purpose of the call reference is to identify the call at the local user-network interface to which the particular message applies. The call reference does not have end-to-end significance across ATM networks.

Call reference values are assigned by the originating side of the interface for a call. These values are unique to the originating side only within a particular signalling virtual channel. The call reference value is assigned at the beginning of a call and remains fixed for the lifetime of a call. To avoid race conditions in certain SAAL error scenarios, it is suggested that implementors avoid immediate reuse of the call reference values after they are released.

The call reference is the second part of every message.

Message type

The purpose is to identify the function of the message being sent and to allow the sender of a message to indicate explicitly the way the receiver should handle unrecorginzed messages. The message type is the third part of every message The possible message types are listed in Table 2.

Table 2: Message type

Bits 8765 4321	Meaning
0 0 0	*Call establishment message:*
0 0 0 1 0	CALL PROCEEDING
0 0 1 1 1	CONNECT
0 1 1 1 1	CONNECT ACKNOWLEDGE
0 0 1 0 1	SETUP
0 1 0	*Call clearing message:*
0 1 1 0 1	RELEASE
1 1 0 1 0	RELEASE COMPLETE
0 0 1 1 0	RESTART
0 1 1 1 0	RESTART ACKNOWLEDGE
0 1 1	*Miscellaneous messages:*
1 1 1 0 1	STATUS
1 0 1 0 1	STATUS ENQUIRY
1 0 0	*Point-to-Multipoint messages:*
0 0 0 0 0	ADD PARTY
0 0 0 0 1	ADD PARTY ACKNOWLEDGE
0 0 0 1 0	ADD PARTY REJECT
0 0 0 1 1	DROP PARTY
0 0 1 0 0	DROP PARTY ACKNOWLEDGE

Message length

Identifies the length of the contents of a message. It is the binary coding of the number of octets of the message contents, excluding the octets used for

- protocol discriminator;
- call reference;
- message type;

and for the message length indication itself.

Continuing with the different messages concerning call establishment and call clearing.

SETUP message

The SETUP message consists of the following information elements (only the mandatory information elements are listed):

- *protocol discriminator*

  Length 1 octet, both directions (calling user to network and network to called user).

0	0	0	0	1	0	0	1

  Q.2931 User-Network CALL CONTROL message

- *call reference*

  Length 4 octets, both directions

0	0	0	0	length call reference value
F	call reference value			
call reference value				
call reference value				

F Flag, 0 ⇒ message is sent from the side that originates the call reference;
1 ⇒ message is sent to the side that originates the call reference;
The call reference flag identifies who allocated the call reference value and the only purpose of the call reference flag is to resolve simultaneous attempts to allocate the same call reference value.

- *message type*

  Purpose is to identify the function of the message being sent and to allow the sender of a message to indicate explicitly the way the receiver should handle unrecorginzed messages.

  Length 2 octets, both directions.

message type							
0	0	0	0	0	1	0	1
1 ext	0 spare	0 spare	F	0 spare	0 spare	Action indicator	

F Flag, 0 ⇒ message instruction field not significant (regular error handling procedures apply)
1 ⇒ follow explicit instructions

Action indicator
00 ⇒ clear call
01 ⇒ discard and ignore
00 ⇒ discard and report status
00 ⇒ reserved

- message length

  Length 2 octets, both directions.

message length
message length (continued)

- ATM traffic descriptor
  Length between 12 and 30 octets, both directions.
  The purpose of the ATM traffic descriptor information element is to specify the set of traffic parameters which, specify a traffic control capability.
  It consists of the following traffic description information elements:
  - *ATM user cell rate Information element identifier* ⇒ 01011001, length 1 octet;
  - an octet containing among other things an IE instruction field and the coding standard
  - length of ATM user cell rate contents, length 2 octets;
  - *forward peak cell rate identifier* when the CLP=0 (no congestion or other ATM-flow disturbance has occurred)⇒ 10000010;
  - forward peak cell rate, length 3 octets;
  - *backward peak cell rate identifier* when the CLP=0 (no congestion or other ATM-flow disturbance has occurred) ⇒ 10000011;
  - backward peak cell rate, length 3 octets;
  - *forward peak cell rate identifier* when the CLP=0 or 1 (congestion or other ATM-flow disturbance could have occurred)⇒ 10000100;
  - forward peak cell rate, length 3 octets;
  - *backward peak cell rate identifier* when the CLP=0 or 1(congestion or other ATM-flow disturbance could have occurred) ⇒ 10000101;
  - backward peak cell rate, length 3 octets;
  - *forward sustainable cell rate identifier* when the CLP=0 ⇒ 10001000
  - forward sustainable cell rate, length 3 octets
  - *backward sustainable cell rate identifier* when the CLP=0 ⇒ 10001001
  - backward sustainable cell rate, length 3 octets
  - *forward sustainable cell rate identifier* when the CLP=0 or 1⇒ 10010000
  - forward sustainable cell rate, length 3 octets
  - *backward sustainable cell rate identifier* when the CLP=0 or 1⇒ 10010001
  - backward sustainable cell rate, length 3 octets
  - *forward maximum burst size identifier* when CLP=0 ⇒ 10100000
  - forward maximum burst size, length 3 octets
  - *backward maximum burst size identifier* when CLP=0 ⇒ 10100001
  - backward maximum burst size, length 3 octets
  - *forward maximum burst size identifier* when CLP=0 or 1 ⇒ 10110000
  - forward maximum burst size, length 3 octets
  - *backward maximum burst size identifier* when CLP=0 ⇒ 10110001
  - backward maximum burst size, length 3 octets
  - *best effort indicator*, QoS class 0 is used with the best effort indication, ⇒ 10111110
  - *traffic management options identifier* ⇒ 10111111
  - one octet consisting of 6 bits which are reserved and set to 0, one tagging backward bit and one tagging forward bit

The term forward indicates the direction from calling user to called user.
The term backward indicates the direction from called user to calling user.
In the traffic description both the operation and management traffic and the user traffic are included.
Not all of the above mentioned traffic descriptors must be used during call SETUP.
All the mentioned cell rate parameters indicate the specifically rate (for example forward sustainable cell rate) in cells per second.
The tagging parameter is coded as one (1) when the tagging is requested and is coded as a zero (0) when tagging is not requested.

The valid combinations of the traffic descriptor subfields in the ATM traffic descriptor information element are shown in table 2. Table 2 shows the valid combinations of traffic parameter subfield in a given direction (the set of traffic parameter subfields used in the forward direction may differ from the ones used in the backward direction).

Table 3:allowable combinations of traffic parameters in a given direction

Allowable combinations of traffic parameter subfields in the ATM user cell rate information element for a given direction
Peak cell rate CLP=0 Peak cell rate CLP=0+1
Peak cell rate CLP=0 Peak cell rate CLP=0+1 Tagging= tagging requested
Peak cell rate CLP=0+1 Sustainable cell rate CLP=0 Maximum burst size CLP=0
Peak cell rate CLP=0+1 Sustainable cell rate CLP=0 Maximum burst size CLP=0 Tagging= tagging requested
Peak cell rate CLP=0+1
Peak cell rate CLP=0+1 Sustainable cell rate CLP=0+1 Maximum burst size CLP=0+1

<table>
<tr><td colspan="3">ATM user cell rate information element identifier</td></tr>
<tr><td>ext
1</td><td>coding
standard</td><td>IE instruction
field</td></tr>
<tr><td colspan="3">Length of ATM user cell
rate contents</td></tr>
<tr><td colspan="3">forward peak cell rate identifier
(CLP=0)</td></tr>
<tr><td colspan="3">forward Peak Cell Rate</td></tr>
<tr><td colspan="3">backward peak cell rate identifier
(CLP=0)</td></tr>
<tr><td colspan="3">backward Peak Cell Rate</td></tr>
<tr><td colspan="3">forward peak cell rate identifier (CLP=0+1)</td></tr>
<tr><td colspan="3">forward peak cell rate</td></tr>
<tr><td colspan="3">backward peak cell rate identifier
(CLP=0+1)</td></tr>
<tr><td colspan="3">backward peak cell rate</td></tr>
<tr><td colspan="3">forward sustainable cell rate identifier (CLP=0)</td></tr>
<tr><td colspan="3">forward sustainable cell rate</td></tr>
<tr><td colspan="3">backward sustainable cell rate
identifier (CLP=0)</td></tr>
<tr><td colspan="3">backward sustainable cell rate</td></tr>
</table>

Continued on next page

<table>
<tr><td colspan="3">forward sustainable cell rate identifier (CLP=0+1)</td></tr>
<tr><td colspan="3">forward sustainable cell rate</td></tr>
<tr><td colspan="3">backward sustainable cell rate identifier (CLP=0+1)</td></tr>
<tr><td colspan="3">backward sustainable cell rate</td></tr>
<tr><td colspan="3">forward maximum burst size identifier (CLP=0)</td></tr>
<tr><td colspan="3">forward maximum burst size</td></tr>
<tr><td colspan="3">backward maximum burst size identifier (CLP=0)</td></tr>
<tr><td colspan="3">backward maximum burst size</td></tr>
<tr><td colspan="3">forward maximum burst size identifier (CLP=0+1)</td></tr>
<tr><td colspan="3">forward maximum burst size</td></tr>
<tr><td colspan="3">backward maximum burst size identifier (CLP=0+1)</td></tr>
<tr><td colspan="3">backward maximum burst size</td></tr>
<tr><td colspan="3">best effort indicator</td></tr>
<tr><td colspan="3">traffic management options identifier</td></tr>
<tr><td>reserved
tagging</td><td>tagging</td><td></td></tr>
</table>

Appendix H

- *Broadband bearer capability*

  The purpose of the Broadband bearer capability information element is to indicate a requested broadband connection oriented bearer service to be provided by the network. It consists of the following octets:
  - broadband bearer capability information element identifier ⇒ 01011110
  - an octet contains among other things an IE instruction field and the coding standard, length 1 octet
  - length of the Broadband bearer capability contents, 2 octets
  - three octets containing among other things the bearer class, traffic type, timing requirements.

  It is used in both directions.

  There are three different bearer classes defined, for each class of service A, C and X. There are three different possibilities concerning the traffic type:
  1. no indication ⇒ 000;
  2. constant bit rate ⇒ 001;
  3. variable bit rate ⇒ 010.

  If we look at the timing requirements four different possibilities are distinghuised:
  1. no timing requirements indicated ⇒ 00;
  2. end-to-end timing required ⇒ 01;
  3. end-to-end timing not required ⇒ 10;
  4. reserved ⇒ 11.

  Concerning the user plane connection two possible configurations are supported:
  1. point-to-point ⇒ 00;
  2. point-to-multipoint ⇒ 01.

<table>
<tr><td colspan="4">broadband bearer capability information element identifier
0 1 0 1 1 1 1 0</td></tr>
<tr><td>ext</td><td>coding
standard</td><td colspan="2">IE instruction
field</td></tr>
<tr><td colspan="4">length of broadband bearer
capability contents</td></tr>
<tr><td>0/1
ext</td><td>0 0
spare</td><td colspan="2">bearer class</td></tr>
<tr><td>1
ext</td><td>0 0
spare</td><td>traffic
type</td><td>timing
requirements</td></tr>
<tr><td>1
ext</td><td>susceptibility
to clipping</td><td>0 0 0
spare</td><td>user plane connec
tion configuration</td></tr>
</table>

- *called party number*

  The purpose of the called party number information element is to identify the called party of a call and it is used in both directions.

  The format of the called party number information element depends on the numbering plan. An ATM end-system address uniquely identifies an ATM endpoint. The format of an ATM address for end-points in private networks is modelled after the format of an OSI Network Service Access Point. There are three different structures of an ATM address. These three structures are:
  1. DCC ATM format (DCC= Data Country Code);
  2. ICD ATM format (ICD= International Code Designator);
  3. E.164 ATM format.

  Every address format start with the AFI field. The AFI field (Authority and Format Identifier) identifies the authority allocating the DCC, ICD, or E.164 number.

Table 4: AFI identifies authority

AFI	*Format of IDI and DSP*
39	DCC ATM format
47	ICD ATM format
45	E.164 ATM format

The format of the called party number information element is illustrated below.

Called party number information number							
0	1	1	1	0	0	0	0
1 ext	coding standard		IE instruction field				
length of called party							
number contents							
1 ext	type of number		addressing/numbering plan identification				
0	address/number digits (IA5 characters)						
ATM endsystem address octets							

- *Connection identifier*

  The connection identifier information element identifies the local ATM connection resources on the interface. See below for a format of the connection identifier information element, its length is 9 octets and is used between network and called user during SETUP initiating.

connection identifier information element identifier							
0	1	0	1	1	0	1	0
1 ext	coding standard		IE instruction field				
length of connection							
identifier contents							
1 ext	0	0 spare	VP associated signalling		preferred/ exclusive		
virtual path connection identifier							
continued							
virtual channel identifier							
continued							

In the implementation agreement of Forum version 3.1 the network always assigns the VPCI and the VCI. The virtual path connection identifier consists of a two octet binary number assigned to the ATM connection representing the identifier of the virtual path connection. For this implementation agreement, the VCPI value is numerically equivalent to the VPI value assigned for the call. The VPI value is coded in the continued virtual path connection identifier. The bits of the first octet of the virtual path connection identifier are coded to all “0’s”

- *QoS parameter*
  The purpose of the Quality of Service parameter information element is to request and indicate the Quality of Service Class for a connection. If information about requested QoS is not available at the terminating interface, the network will generate the default value (unspecified QoS) for the QoS parameter information element for the transfer to the called user. The length of the Quality of Service parameter information element is 6 octets and it is used in both directions.

<table>
<tr><td colspan="8">Quality of Service parameter information element identifier</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td>1
ext</td><td colspan="3">coding
standard</td><td colspan="4">IE instruction
field</td></tr>
<tr><td colspan="8">length of quality of service</td></tr>
<tr><td colspan="8">parameter contents</td></tr>
<tr><td colspan="8">QoS class forward</td></tr>
<tr><td colspan="8">QoS class backward</td></tr>
</table>

There is distinguished between 5 different QoS classes:

Table 5: QoS forward and backward

bits	*meaning*
0000 0000	QoS class 0 (unspecified class)
0000 0001	QoS class 1
0000 0010	QoS class 2
0000 0011	QoS class 3
0000 0100	QoS class 4
1111 1111	reserved

Call proceeding

This message is sent by the called user to the network or by the network to the calling user to indicate that the requested call establishment has been initiated and no more call establishment information will be accepted. The sending of this message is optional but receiptance is required.
The first nine octets of all the messages are always the protocol discriminator; the call reference, the message type; and the message length. The message type makes it possible to distinguish between the different messages. After these four information elements to optional information fields can be added.
The first one is the connection identifier with a length between 4 and 9 octets. It is mandatory in the network-to-user direction if this message is the first message in response to a SETUP message. It is mandatory in the user-to-network direction if this message is the first response to a SETUP message, unless the user accepts the connection identifier indicated in the SETUP message
The second optional message is mandatory if an endpoint was included in the SETUP message, its length is between 4 and 9 octets.

Connect

This message is sent by the called user to the network and by the network to the calling user to indicate call acceptance by the called user.
The additional information elements are:

- AAL parameters;
- Broadband low layer information;
- Connection identifier, it is mandatory in the network-to-user direction if this message is the first message in response to a SETUP message. It is mandatory in the user-to-network direction if this message is the first message in response to a SETUP message, unless the user accepts the connection identifier indicated in the SETUP message;
- Endpoint reference, is mandatory if the endpoint reference was included in the SETUP message.

Connect acknowledge

This message is sent by the network to the called user to indicate the user has been awarded the call. It is also sent by the calling user to the network to allow symmetrical call control procedures. There is no possibility to add extra information elements to the connect acknowledge message.

Release

This message is sent by the user to request the network to clear the end-to-end connection (if any) or is sent by the network to indicate that the en-to-end connection is cleared and that the receiving equipment should release the virtual channel and prepare to release the call reference after sending a RELEASE COMPLETE. When this message is sent a additional Cause field must be added. The cause field indicates the reason why the connection will be released, the length of the cause field is in the range between 6 and 34 octets.

Release complete

This message is sent by the user or the network to indicate that the equipment sending the message has released the virtual channel and call reference, the virtual channel is available for reuse, and the receiving equipment shall release the call reference. When this message is sent a additional Cause field must be added. The cause field indicates the reason why the connection will be released, the length of the cause field is in the range between 4 and 34 octets. The adding of the cause information element is mandatory in the first call clearing message; including when the RELEASE COMPLETE is sent as a result of an error condition.

Status

This message is sent by the user or the network in response to a STATUS REQUIRY message or at any time to report certain error conditions. Two additional fields are mandatory and two are optional.
The two mandatory fields are the call state and the cause field. The call state can be one of the states listed below:

- null;
- call initiated;
- outgoing call proceeding;
- call delivered;
- call present;
- call received;
- connect request;
- incoming call proceeding;
- active;
- release request;
- release indication;
- call abort;
- restart request;
- restart.

Status enquiry

The status enquiry message is sent by the user or the network at any time to solicit a STATUS message from the peer layer 3 entity. Besides the four usual information elements the use of the endpoint reference information element is optional.

Restart
This message is sent by the user or the network to request the recipient to restart (i.e. release all resources associated with) the indicated virtual channel or all virtual channels controlled by the signalling virtual channel. Two information fields are added. The first is the connection identifier which is optional and has a length in the range between 4 and 9 octets, the other one is the restart indicator which has a length of 5 octets and is mandatory.

Restart acknowledge
This message is sent to acknowledge the receipt of a RESTART message and to indicate that the requested restart is complete. Two information fields are added. The first is the connection identifier which is optional and has a length in the range between 4 and 9 octets, the other one is the restart indicator which has a length of 5 octets and is mandatory.