On a Call Connection Interface Switch Design

Call Processor Inter-operability between UNI Signaling and IS-136 TDMA Air Interface Standard

T. Vrind, A. Jain
Summer Project
HCL Info Systems Limited India

Introduction

This project specifies the procedures for dynamically establishing, maintaining and clearing ATM to Mobile connections at the User Network Interface (UNI). This project is defined in terms of messages and information elements used to characterize the ATM to Mobile connection and ensure interoperability.

The purpose of this project is to design a call/connection control interface switch. It is required to develop a Call-Processor, which can establish connection between an ATM user and a mobile user. The Call-Processor is required to operate with UNI signals as well as IS-136 Wireless Loop (Air Interface) standard.

The proposed system is able to establish the call/connection from a single ATM user to a single Mobile user. The call established can be disconnected from the ATM side after the connection has been sustained for a fixed time slice as included in the design. The system design includes three different processes

- ATM Process
- Call- Processor Process
- Mobile Process

The processes communicate amongst themselves using UNIX - System V message queues. The software includes the use of Timers thus terminating the processes in case of a system error or unavailability of the messages on the message queues.

SYSTEM ARCHITECTURE

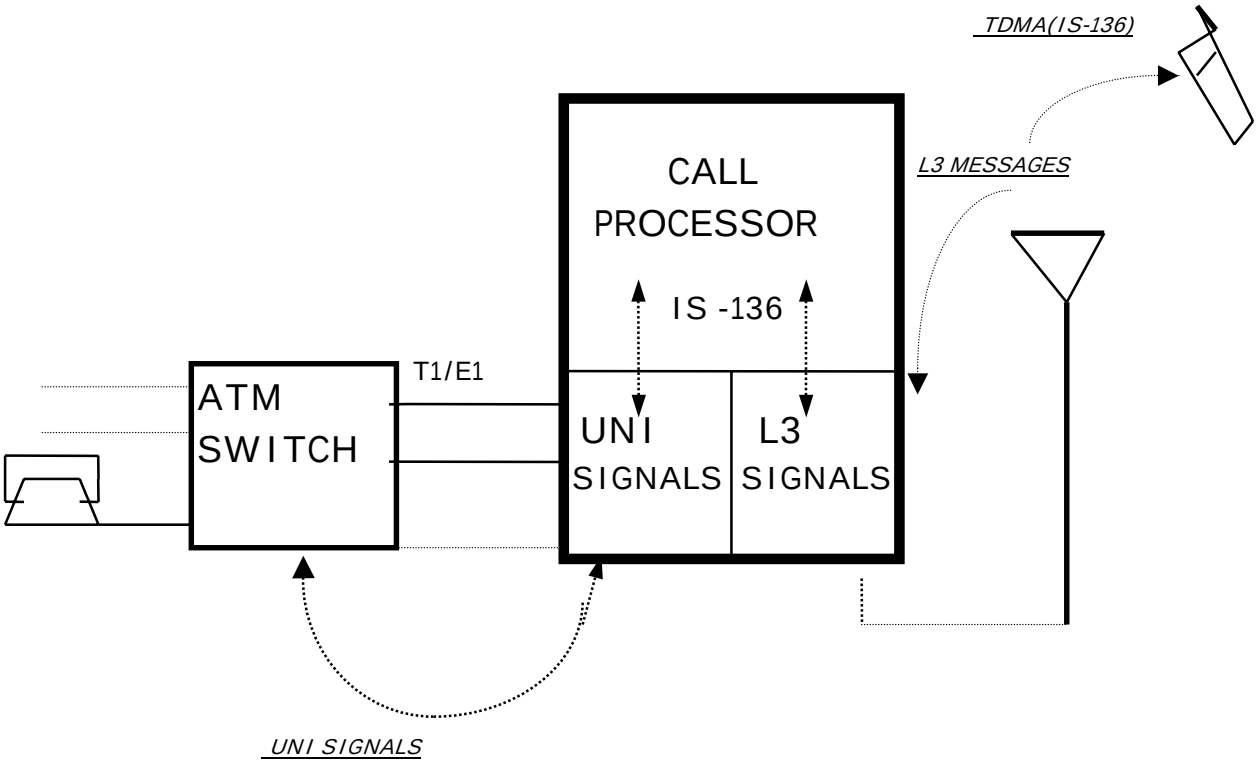


Figure 1. IS - 136 Wireless Loop ATM - Call Processor Design.

ATM SWITCH:

The System consists of an ATM Switch to which a number of land users (ATM users)can be connected to by normal cables and ports when they require to establish connection. The number of phones, which can be hooked up to the ATM switch, is fixed. There is a physical link between the two. It communicates with the Call - Processor using T1/E1 line and uses UNI signals for communicating.

CALL - PROCESSOR (CP):

Call - Processor is a protocol converter, which is able to receive UNI signals from ATM switch and transmit L3 messages to the mobile and vice-versa. Its basic function is switching messages from ATM landline phone to the mobile.

MOBILE:

Mobiles are not directly connected to the Call-Processor that is there is no physical link between the two. Mobile communicates with the Call -Processor by the help of an antenna. The mobile and the antenna are tuned to the same frequency band (Digital Channel) and hence they can transfer messages between themselves. When the mobile is

not busy it is tuned to the Digital Control Channel (DCCH) on which it can receive broadcast messages and call originating / set-up messages. When the mobile is busy it is tuned to the Digital Traffic Channel (DTC) or the voice channel. Control channel is unique for al the mobiles in the wireless loop, however the voice channel is different for all the mobiles which are busy at the same time.

OVERVIEW

The following section summarizes the messages for ATM & Mobile point - to - point call and connection control.

UNI SIGNALLING :

CALL PROCEEDING

Information Element	Reference	Direction	Type	Length
Protocol discriminator	5.4.2	both	M	1
Call reference	5.4.3	both	M	4
Message type	5.4.4.1	both	M	2
Message length	5.4.4.2	both	M	2
Connection identifier	5.4.5.16	both	O (1)	4-9
Endpoint reference	5.4.8.1	both	O(2)	4-7

Note 1- Mandatory in the network-to-user direction if this message is the first message in response to a SETUP message. Its 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.

Note 2 - Mandatory if an Endpoint reference was included in the SETUP message.

CONNECT

Information Element	Reference	Direction	Type	Length
Protocol discriminator	5.4.2	both	M	1
Call reference	5.4.3	both	M	4
Message type	5.4.4.1	both	M	2
Message length	5.4.4.2	both	M	2
AAL parameters	5.4.5.5	both	O (1)	4-11
Broadband low layer information	5.4.5.9	both	O (2)	4-17
Connection identifier	5.4.5.16	both	O (3)	4-9
Endpoint reference	5.4.8.1	both	O(4)	4-7

Note 1- Included in the user to network direction when the called user wants to pass ATM adaptation layer parameters information to the calling user, and the ATM adaptation layer parameters information element was present in the SETUP message. Included in the network to user direction if the called user included an ATM adaptation layer parameters information element in the CONNECT message. The ATM adaptation layer parameters information element shall not be present when the endpoint reference information element was present in the SETUP message and contained a non-zero value.

Note 2 -Included in the user-to-network direction when the answering user wants to return low layer information to the calling user. Included in the network-to-user direction if the user awarded the call included a Broadband low layer information element in the CONNECT message. Optionally included for Broadband low layer information negotiation, but some networks may not transport this information element to the calling user (see Annex C).

Note3 - Mandatory in the network-to-user direction if this message is the first message in response to a SETUP message. Its 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.

Note4 - Mandatory if the Endpoint reference was included in the SETUP message.

RELEASE

Information Element	Reference	Direction	Type	Length
Protocol discriminator	5.4.2	both	M	1
Call reference	5.4.3	both	M	4
Message type	5.4.4.1	both	M	2
Message length	5.4.4.2	both	M	2
Cause	5.4.5.15	both	M	6-34

RELEASE COMPLETE

Information Element	Reference	Direction	Type	Length
Protocol discriminator	5.4.2	both	M	1
Call reference	5.4.3	both	M	4
Message type	5.4.4.1	both	M	2
Message length	5.4.4.2	both	M	2
Cause	5.4.5.15	both	O(2)	4-34

SETUP

Information Element	Reference	Direction	Type	Length
Protocol discriminator	5.4.2	both	M	1
Call reference	5.4.3	both	M	4
Message type	5.4.4.1	both	M	2
Message length	5.4.4.2	both	M	2
AAL parameters	5.4.5.5	both	O(1)	4-20
ATM traffic descriptor	5.4.5.6	both	M	12-30
Broadband bearer capability	5.4.5.7	both	M	6-7
Broadband high layer information	5.4.5.8	both	O(2)	4-13
Broadband repeat indicator	5.4.5.19	both	O(3)	4-5
Broadband low layer information	5.4.5.9	both	O(4)	4-17
Called party number	5.4.5.11	both	M	(5)
Called party subaddress	5.4.5.12	both	O(6)	4-25
Calling party number	5.4.5.13	both	O(7)	4-26
Calling party subaddress	5.4.5.14	both	O(8)	4-25
Connection identifier	5.4.5.16	N -> U	M	9
QoS parameter	5.4.5.18	both	M	6
Broadband sending complete	5.4.5.21	both	O(9)	4-5
Transit network selection	5.4.5.22	U -> N	O(10)	4-8
Endpoint reference	5.4.8.1	both	O(11)	4-7

Note 1 - Included in the user-to-network direction when the calling user wants to pass ATM adaptation layer parameters information to the called user. Included in the network-to-user direction if the calling user included an ATM adaptation layer parameters information element in the SETUP message.

Note 2 - Included in the user-to-network direction when the calling user wants to pass broadband high layer information to the called user. Included in the network-to-user direction if the calling user included a Broadband high layer information element in the SETUP message.

Note 3 - Included when two or more Broadband low layer information elements are included for Broadband low layer information negotiation.

Note 4 - Included in the user-to-network direction when the calling user wants to pass broadband low layer information to the called user. Included in the network-to-user direction if the calling user included a Broadband low layer information element in the SETUP message. Two or three information elements may be included in descending order of priority, i.e., highest priority first, if the Broadband low layer information negotiation procedures are used (see Annex C).

Note 5 - Minimum length depends on the numbering plan. Maximum length is 25 octets.

Note 6 - Included in the user-to-network direction when the calling user wants to indicate the called party sub address. Included in the network-to-user direction if the calling user included a Called party sub address information element in the SETUP message.

Note 7 - May be included by the calling user or by the network to identify the calling user.

NoteÊ8 - Included in the user-to-network direction when the calling user wants to indicate the calling party subaddress. Included in the network-to-user direction if the calling user included a Calling party subaddress information element in the SETUP message.

Note 9 - It is optional for the user to include the Broadband sending complete information element when enbloc sending procedures (i.e., complete address information is included) are used; its interpretation by the network is optional. It is optional for the network to include the Broadband sending complete information element when enbloc receiving procedures (i.e., complete address information is included) are used.

Note 10 -Included by the calling user to select a particular transit network (see Annex D.)

IS-136 (L3 MESSAGES)

DCCH

Digital Traffic Channel Designation (DTCD)

Information Element	Length (Bits)
Protocol Discriminator	2
Message Type	6
DVCC	8
DMAC	4
CHAN	11
ATS	4
SB	1
Protocol Version	4
Time Alignment	5
Delay Interval Compensation Mode	1
Voice Mode	10
Subaddress	20 - 180
Message Encryption Mode	13
Hyperband Info	6
Display	12 - 668

Page

Information Element	Length (Bits)
Protocol Discriminator	2
Message Type	6
Service Code	4

Called Party Subaddress	20 - 180
Signal	16
Calling Party Number Presentation Indicator	8
Calling Party Number	20 - *
Calling Party Subaddress	20 - 180
Display	12 - 668
Called Party	20 - *

Page Response

Information Element	Length (Bits)
Protocol Discriminator	2
Message Type	6
Protocol Version	4
Last Try	1
SCM	5
Service Code	4
Voice Mode	10
Data Mode	16
Message Encryption Mode	13
Bandwidth	7
User Group	28,32,42,58
Subaddress	20 - 180

DTC

Alert With Info

Information Element	Length (bits)
Protocol Discriminator	2
Message Type	8
Remaining Length	6
Signal	10 - 18
Calling Party Number	26 - 146
Calling Party Name	26 - 514

Release

Information Element	Length (bits)
Protocol Discriminator	2
Message Type	8
Remaining Length	6

Digital Control Channel Information	31 - 73

Connect

Information Element	Length (bits)
Protocol Discriminator	2
Message Type	8

DVCC Present

Information Element	Length (bits)
Protocol Discriminator	2
Message Type	8
CHAN	11

Mobile Ack

Information Element	Length (bits)
Protocol Discriminator	2
Message Type	8
Ack Message Type	8
Remaining Length	6
Last Decoded Parameter	14

SYSTEM DESIGN

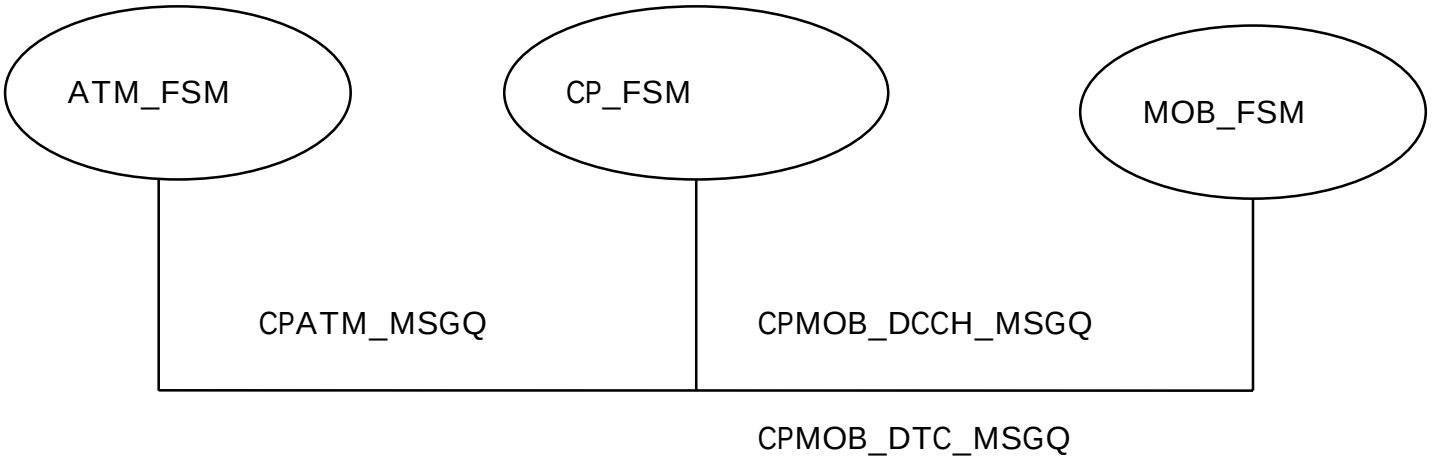


Figure 2. System Design.

The system consists of three finite state machines

- ATM_FSM
- CP_FSM
- MOB_FSM

All the finite state machines have a unique initial state and a unique final state. The state transition occurs by receiving and/or sending messages on the UNIX System V message queues.

Message Queues:

Message queue is a type of UNIX System V IPC(Inter Process Communication). In System V implementation of messages, all messages are stored in the kernel and have an associated *message queue identifier*. This identifier is called as *msqid,* that identifies a particular queue of messages. Processes read and write messages to arbitrary queues. There is no requirement that any process be waiting for a message to arrive on a queue before some other process is allowed to write a message to that queue.
Every message on the queue has the following attributes:

- long integer *type;*
- *length* of the data portion of the message (can be zero);
- *data* (if the *length* is greater than zero).

For every message queue in the system, the kernel maintains a structure of information given in *"sys/ipc.h"*. A new message queue is created, or an existing message queue is accessed with the *msgget* system call with the following template.

```
int msgget( key_t , int msgflag );
```

Keys

The function *ftok* is provided by the System V standard 'C' library to convert a pathname and project identifier into a System V IPC key. These keys are used to identify message queues, shared memory and semaphores.
The file <sys/types.h> defines the *key_t* data-type, which is typically a 32-bit integer. The template for *ftok* is :

```
key_t ftok(char *pathname , char proj );
```

The normal implementation of *ftok* is as follows.
It combines the 8-bit *proj* value , along with the numeric I-node number of the disk file corresponding to the specified *pathname* , along with the minor device number of the filesystem on which the disk file resides. The combination of these three values produce a 32-bit key.

Message Flag

The *msgflag* value is a combination of the constants shown below

Numeric	Symbolic	Description
0400	MSG_R	Read by owner
0200	MSG_W	Write by owner
0040	MSG_R >> 3	Read by group
0020	MSG_W>> 3	Write by group
0004	MSG_R >> 6	Read by world
0002	MSG_W >> 6	Write by world
	IPC_CREAT	Creates new entry if does not exist , else returns the existing entry
	IPC_EXCL	It has no meaning if used alone but if used with IPC_CREAT (i.e. both are set) it creates a new entry if it does not exist ,else an error occurs.

Table 1. Msgflag values.

Related System Calls

Once a message queue is opened with *msgget* ,a message can be put on the queue using the *msgsnd* system call with the following template:

int msgsnd(int *msqid* ,struct msgbuf **ptr*, int *length*, int *flag*);

The *ptr* argument is a pointer to a structure with the template defined in<sys/msg.h>. The *length* argument to *msgsnd* specifies the length of the message in bytes. The *flag* argument can be specified as either *IPC_NOWAIT* or as 0. The *IPC_NOWAIT* value allows the system call to return immediately if there is no room on the message queue for the new message. In this case the *msgsnd* returns -1. If the system call is successful , *msgsnd* returns zero.

A message is read from the message queue using the *msgrcv* system call with the following template:

int msgrcv(int *msqid* , struct msgbuf * *ptr* , int *length,* long *msgtype,* int *flag*);

All the fields are same as in case of *msgsnd.* The long integer *msgtype* argument specifies which message on the queue is desired.

- If *msgtype* is zero, the first message on the queue is returned. This specifies that the oldest message on the queue is to be returned.
- If *msgtype* is greater than zero, the first message with a type equal to *msgtype* is returned.
- If *msgtype* is less than zero, the first message with the lowest type that is less than or equal to the absolute value of *msgtype* is returned.

The *msgctl* system call provides a variety of control operations on the message queue. It has the following template:

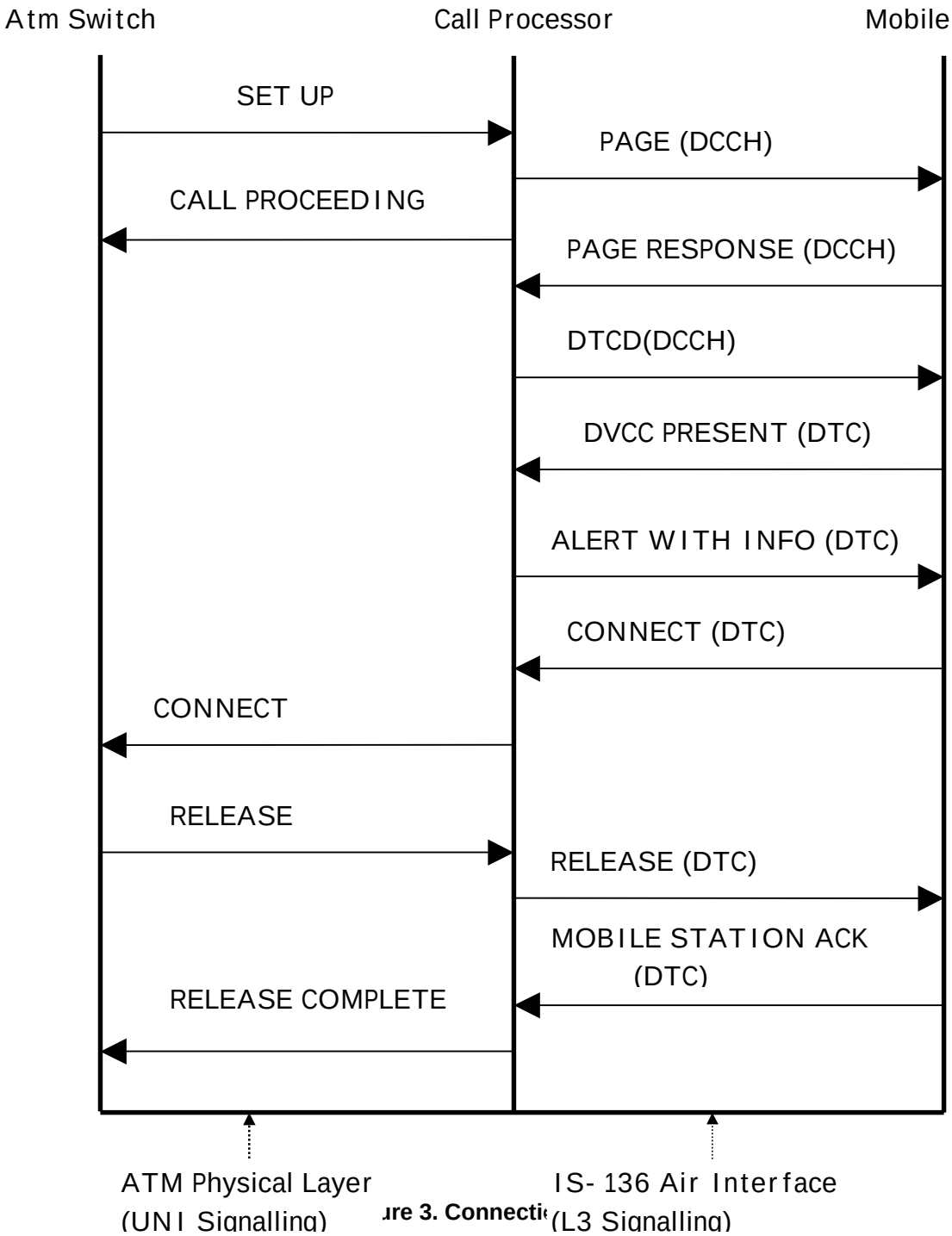
int msgctl(int *msqid,* int *cmd,* struct msqid_ds **buff*);

The only use of this system call in the project is for removing message queues from the system using a *cmd* of *IPC_RMID* .

The message queue identifiers used in the proposed design of the system are :

- CPATM_MSGQ - for communication between Call-Processor and ATM switch.
- CPMOB_DCCH_MSGQ - for communication between Call-Processor and Mobile on the Digital Control Channel.
- CPMOB_DTC_MSGQ - for communication between Call - Processor and Mobile on the Digital Traffic Channel.

CALL PROCESSING:



ıre 3. Connecti

The procedure for establishing connection from the ATM switch to the Mobile is as follows:

- The ATM Switch sends a SETUP message to the Call-Processor on CPATM_MSGQ.

- The Call-Processor in turn sends a PAGE message to the Mobile on CPMOB_DCCH_MSGQ and a CALL PROCEEDING message to the ATM Switch on CPATM_MSGQ.
- The Mobile then sends a PAGE RESPONSE message to the Call-Processor on the CPMOB_DCCH_MSGQ.
- The Call-Processor in turn sends a DTCD message to the Mobile ON CPMOB_DCCH_MSGQ.
- The Mobile then sends a DVCC PRESENT message to the Call-Processor on CPMOB_DTC_MSGQ.
- The Call-Processor in turn sends a ALERT WITH INFO message to the Mobile on CPMOB_DTC_MSGQ.
- The Mobile then sends a CONNECT message to the Call Processor on CPMOB_DTC_MSGQ .
- The Call Processor in turn sends a CONNECT message to the ATM Switch on CPATM_MSGQ.
- The ATM Switch sends a RELEASE message to the Call Processor on CPATM_MSGQ after the pop-up of the designated timer for call sustenance period.
- The Call Processor in turn sends a RELEASE message to the Mobile on CPMOB_DTC_MSGQ.
- The Mobile then sends a MOBILE ACK to the Call Processor on CPMOB_DTC_MSGQ.
- The Call Processor in turn sends a RELEASE COMPLETE to the ATM Switch on CPATM_MSGQ ,thus terminating the call.

LEGEND

The State diagrams given in the following sections use the following meanings for the structures:

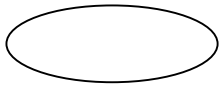
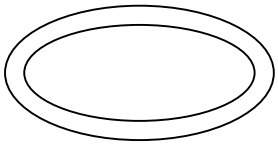
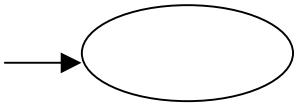
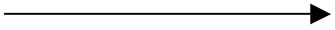
FIGURE	MEANING
	STATE
	FINAL STATE
	INITIAL STATE
	STATE TRANSITIONS
	MESSAGE RECEIVED
	ACTION TAKEN

Table 2. Legends

ATM-FSM

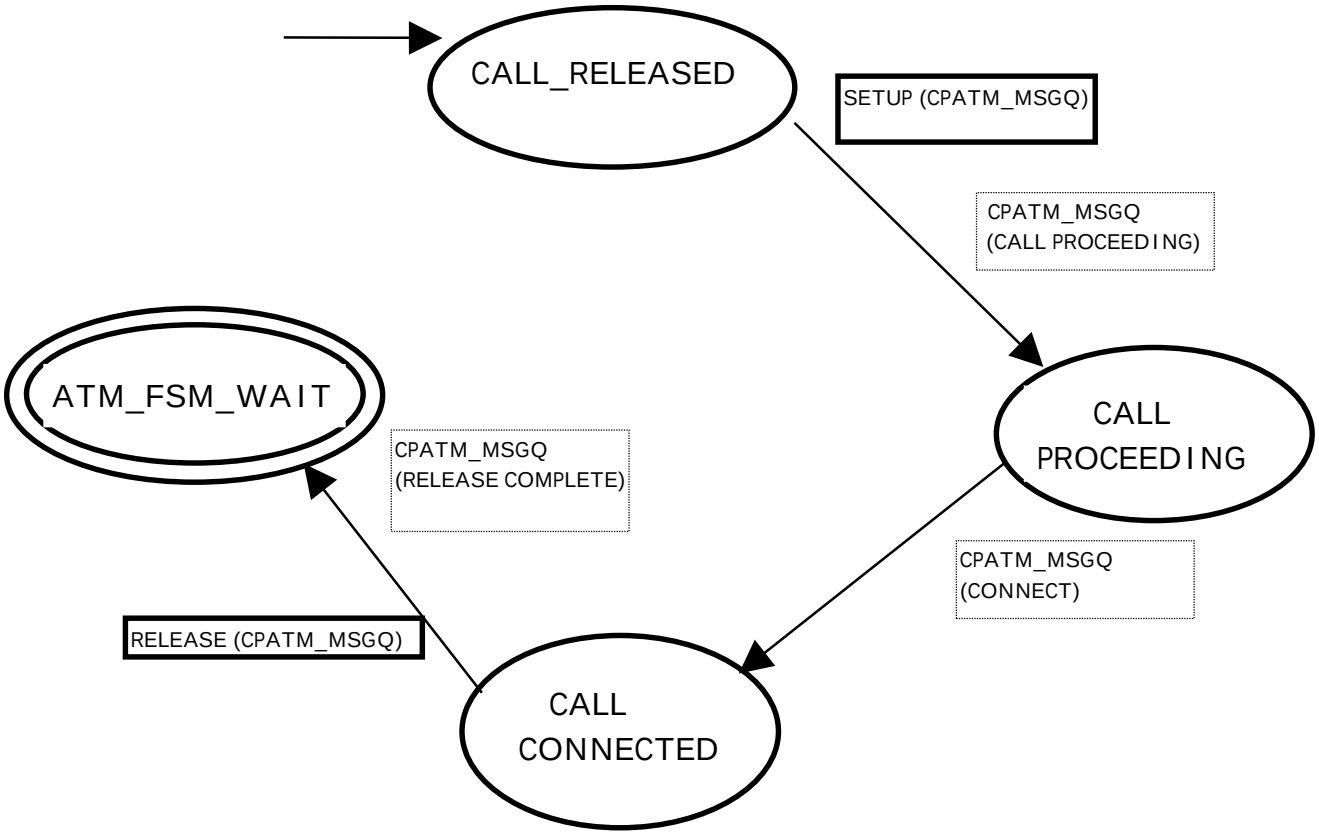


Figure 4. ATM Finite State Machine

The ATM Finite State Machine works on the following lines:

- The initial state of the machine is CALL RELEASED state. It sends a SETUP message to the CPATM_MSGQ ,and changes its state to CALL PROCEEDING when it receives CALL PROCEEDING message on the CPATM_MSGQ.
- In the CALL PROCEEDING state it waits for a CONNECT message on CPATM_MSGQ in order to change its state to CALL CONNECTED.
- In the CALL CONNECTED state it sleeps for a fixed time period during which connection has to be sustained. After this it sends a RELEASE message on CPATM_MSGQ, and goes to its final state ATM_FSM_WAIT on receiving RELEASE COMPLETE message on CPATM_MSGQ.

Note: After sending any message on CPATM_MSGQ the ATM_FSM waits for a fixed time period in order to receive a message. If it is unable to receive any message then it goes back to the CALL RELEASED state.

STATE	ACTION TAKEN	MESSAGE RECEIVED	NEXT STATE
CALL RELEASED	SET UP	CALL PROCEEDING	CALL PROCEEEDING
CALL PROCEEDING	NONE	CONNECT	CALL CONNECTED
CALL CONNECTED	RELEASE	RELEASE COMPLETE	ATM_FSM_WAIT

Table 3. ATM State Transitions.

CP-FSM:

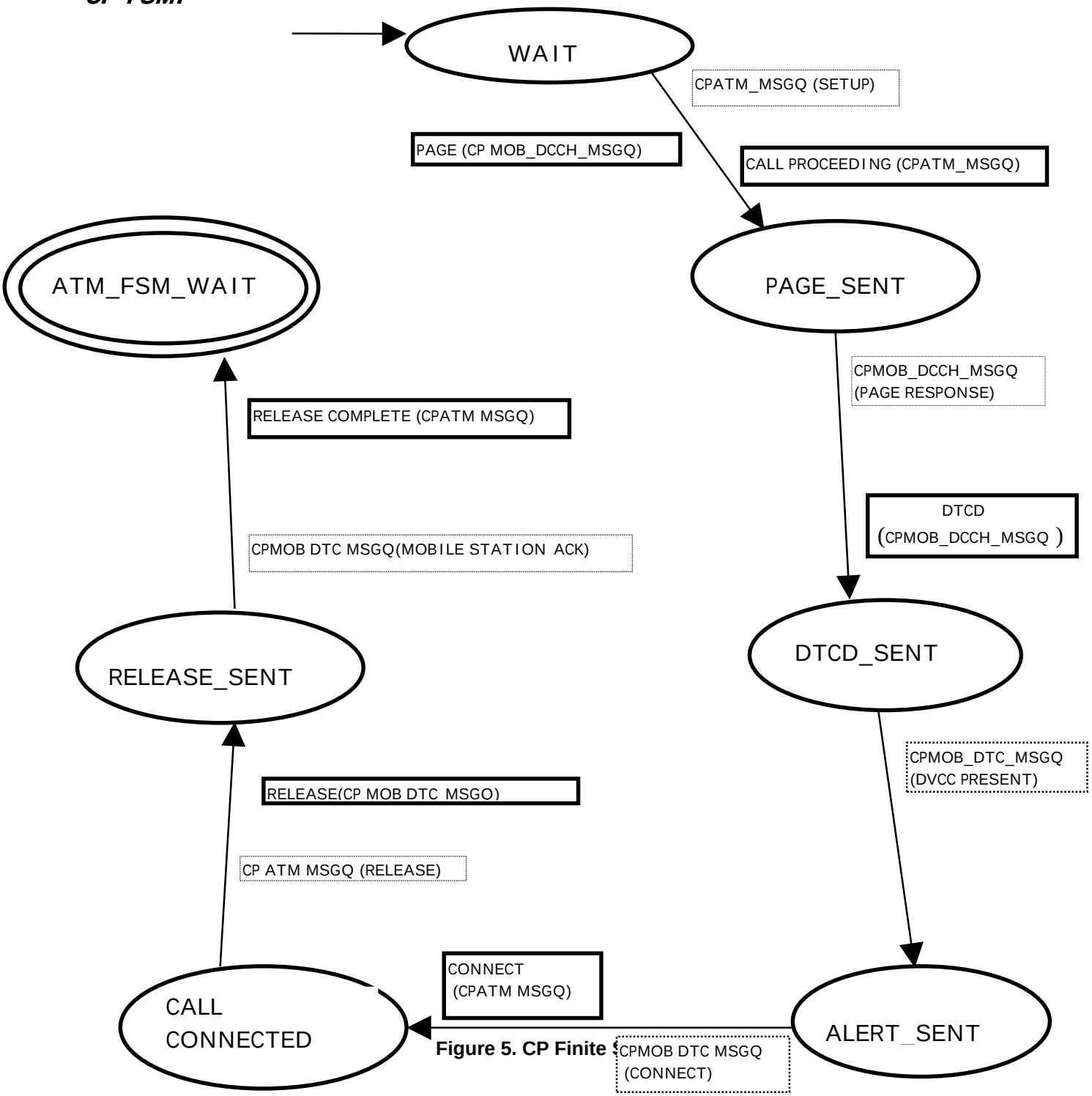


Figure 5. CP Finite S

The CP Finite State Machine works on the following lines :

- The initial state of CP_FSM is WAIT. The machine receives SETUP message on CPATM_MSGQ, sends PAGE message on CPMOB_DCCH_MSGQ and then sends

CALL PROCEEDING message on the CPATM_MSGQ. It then changes it state to PAGE_SENT.

- In the PAGE_SENT state it receives PAGE_RESPONSE on CPMOB_DCCH_MSGQ and sends DTCD on CPMOB_DCCH_MSGQ. It then changes its state to DTCD_SENT.
- In DTCD_SENT state it receives DVCC_PRESENT on CPMOB_DTC_MSGQ and sends ALERT_WITH_INFO on CPMOB_DTC_MSGQ. It then changes its state to ALERT_SENT.
- In ALERT_SENT state it receives CONNECT on CPMOB_DTC_MSGQ and sends CONNECT on CPATM_MSGQ. It then changes its state to CALL CONNECTED.
- In the CALL_CONNECTED state it receives RELEASE on CPATM_MSGQ and sends RELEASE on CPMOB_DTC_MSGQ. It then changes its state to RELEASE_SENT.
- In the RELEASE_SENT state it receives MOBILE_STATION_ACK on CPMOB_DTC_MSGQ and sends RELEASE_COMPLETE on CPATM_MSGQ. It then goes to its final state CP_FSM_WAIT.

STATE	MESSAGE RECEIVED	ACTION TAKEN	NEXT STATE
WAIT	SET UP	PAGE , CALL PROCEEDING	PAGE_SENT
PAGE_ SENT	PAGE RESPONSE	DTCD	DTCD_SENT
DTCD_SENT	DVCC PRESENT	ALERT WITH INFO	ALERT_SENT
ALERT_SENT	CONNECT	CONNECT	CALL_CONNECTED
CALL_CONNECTED	RELEASE	RELEASE	RELEASE_SENT
RELEASE_SENT	MOBILE STATION ACK	RELEASE COMPLETE	CP_FSM_WAIT

Table 4. CP State Transitions.

MOB-FSM:

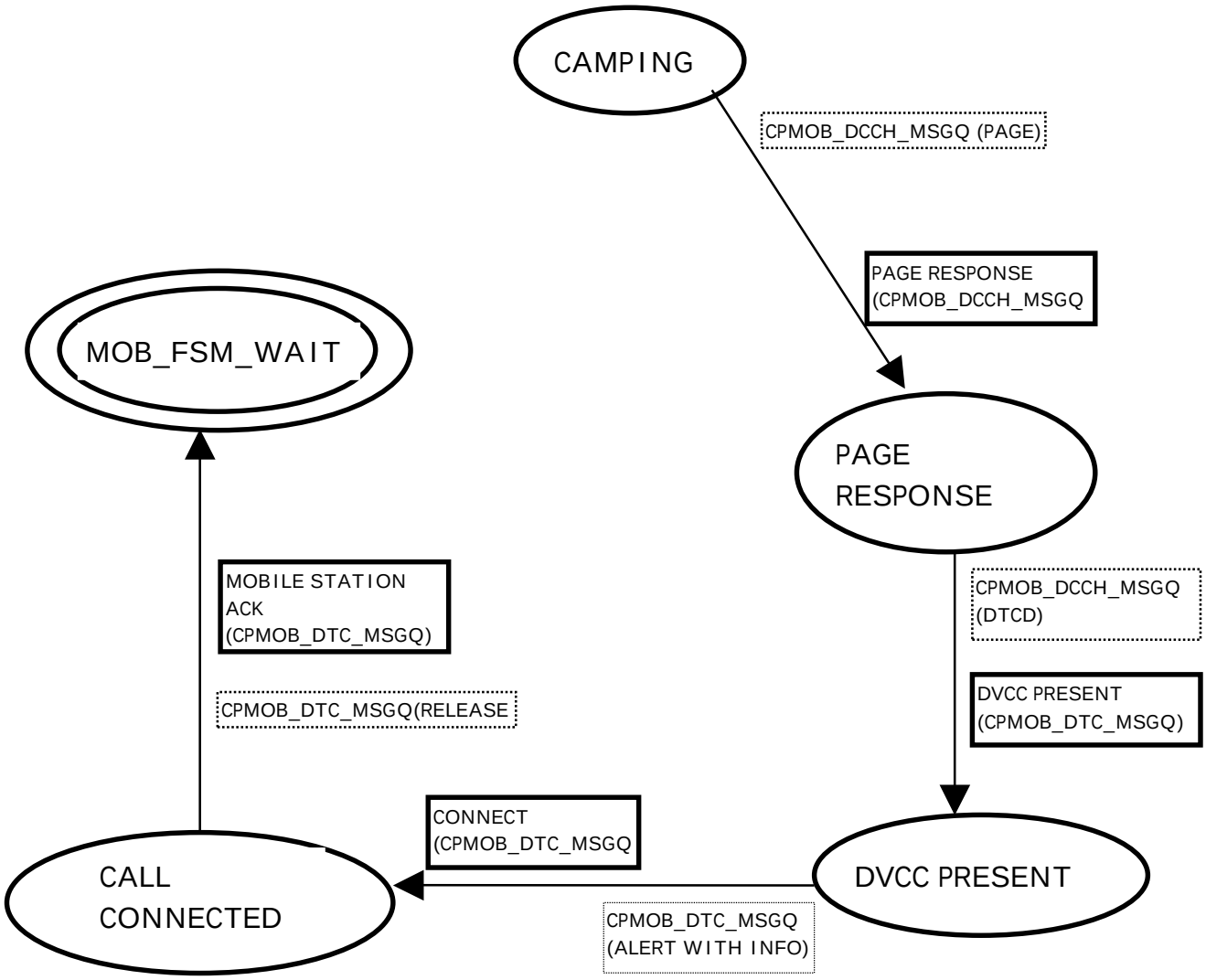


Figure 6. Mobile Finite State Machine.

The Mobile Finite State Machine works on the following lines :

- The initial state of MOB_FSM is CAMPING. It receives PAGE on CPMOB_DCCH_MSGQ and sends PAGE_RESPONSE on CPMOB_DCCH_MSGQ. It changes its state to PAGE_RESPONSE_SENT.
- In PAGE_RESPONSE_SENT state it receives DTCD on CPMOB_DCCH_MSGQ and sends DVCC_PRESENT on CPMOB_DTC_MSGQ. It then changes its state to DVCC_PRESENT_SENT.
- In DVCC_PRESENT_SENT state it receives ALERT_WITH_INFO on CPMOB_DTC_MSGQ and sends CONNECT on CPMOB_DTC_MSGQ. It then changes its state to CALL_CONNECTED.

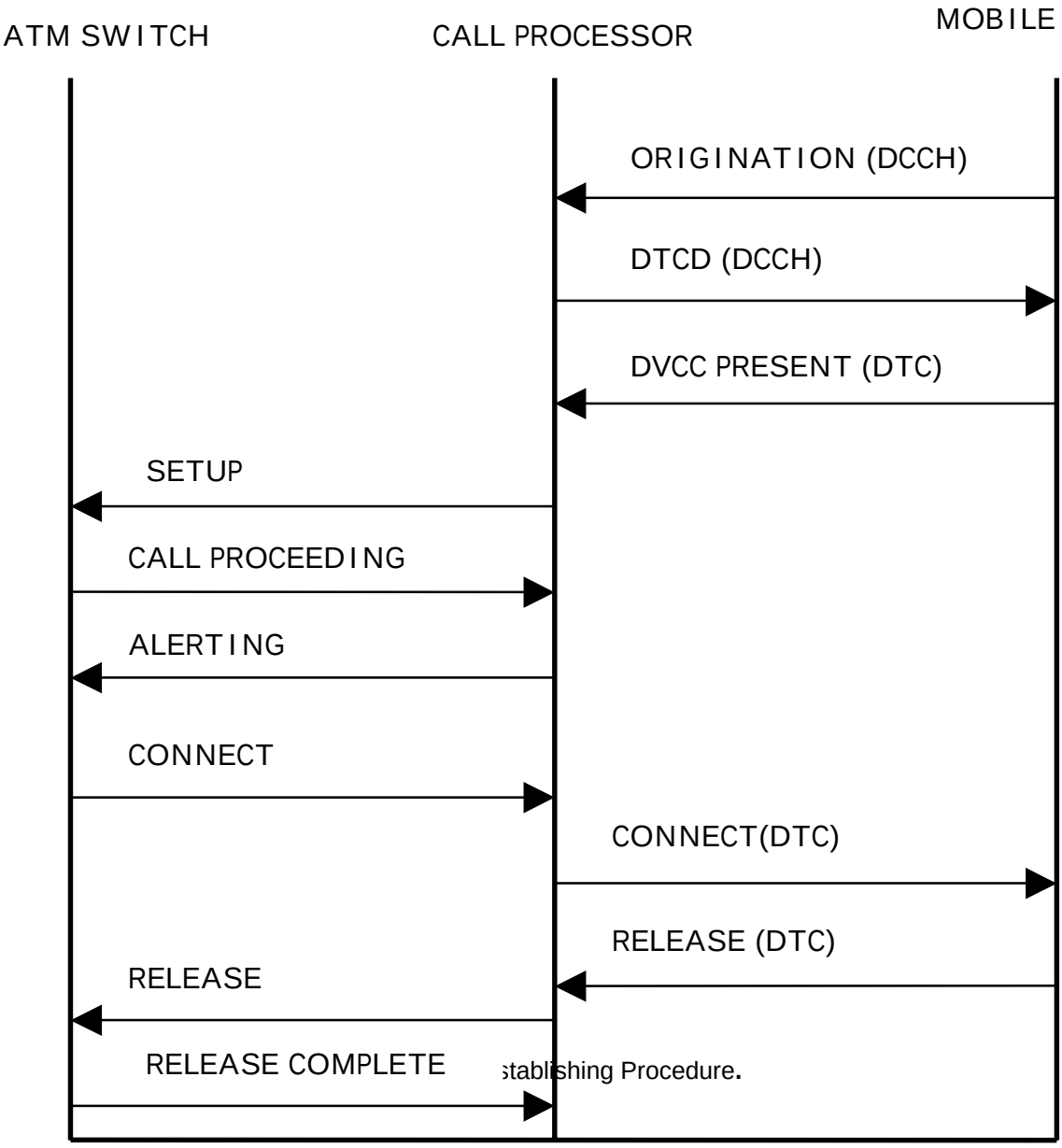
- In CALL_CONNECTED state it receives RELEASE on CPMOB_DTC_MSGQ and sends MOBILE_STATION_ACK on CPMOB_DTC_MSGQ. It then changes its state to MOB_FSM_WAIT.

STATES	MESSAGE RECEIVED	ACTION TAKEN	NEXT STATE
CAMPING	PAGE	PAGE RESPONSE	PAGE_RESPONSE_SENT
PAGE_RESPONSE_SENT	DTCD	DVCC PRESENT	DVCC_PRESENT_SENT
DVCC_PRESENT_SENT	ALERT WITH INFO	CONNECT	CALL_CONNECTED
CALL_CONNECTED	RELEASE	MOBILE STATION ACK	MOB_FSM_WAIT

Table 5. Mobile State Transitions.

SCOPE OF FUTURE WORK

The proposed design in the project is able to establish and terminate call/connection from the ATM Switch only. This feature can be extended further so that call/connection can be established and terminated from Mobile side too. The procedure for call establishment for such a system is given below.



stablishing Procedure.

The same procedure can be followed for drawing the State diagrams and tabulating the Procedures.
The procedure for establishing connection from the Mobile side to the ATM side is as follows:

- Mobile sends an ORIGINATING message on CPMOB_DCCHMSGQ to the Call Processor
- The Call Processor in turn sends a DTCD message on CPMOB_DCCH_MSGQ to the Mobile.
- Mobile then sends a DVCC PRESENT message to the Call Processor on COMOB_DTC_MSGQ.
- The Call Processor in turn sends a SETUP message to the ATM Switch on CPATM_MSGQ.
- ATM Switch then sends a CALL PROCEEDING message to the Call Processor on CPATM_MSGQ.
- The Call Processor in turn sends an ALERTING message to the ATM Switch on CPATM_MSGQ.
- ATM Switch then sends a CONNECT message to the Call Processor on CPATM_MSGQ.
- The Call Processor in turn sends a CONNECT message to the Mobile on CPMOB_DTC_MSGQ.
- Mobile sends a RELEASE message to the Call Processor on CPMOB_DTC_MSGQ when the period for call sustenance gets expired.
- The Call Processor in turn sends a RELEASE message to the ATM Switch on CPATM_MSGQ.
- ATM Switch then sends a RELEASE COMPLETE message to the Call Processor on CPATM_MSGQ ,thus indicating the termination of the call.

The proposed design is compatible with a single ATM user and a single Mobile user. This feature can be extended to produce a multi user design in which the Call Processor can maintain a number of calls/connection at the same time. In such a design the Call Processor should maintain a Call Record for each connection and changes of state should occur in the records.

ABBREVIATIONS

ATM	Asynchronous Transfer Mode
UNI	User Network Interface
L3	Layer 3
DCCH	Digital Control Channel
DTC	Digital Traffic Channel
DTCD	Digital Traffic Channel Designation
IPC	Inter Process Communication
CP	Call Processor
DVCC	Digital Verification Colour Code

References

1. ATM User Network Interface Specification (v3.1) Section 5: UNI Signaling
2. TIA/EIA IS - 136 – Layer 3 - Messages: Message Formats