

NAGRAVISION SMS GATEWAY

Interface Specification

for Native Aladin 1.x Systems

ISSUE 2.7.6

SAS

CUSTOMER NAME

NAGRAVISION S.A.
is a member of the KUDELSKI GROUP OF COMPANIES.

This document contains confidential and privileged information.
The reproduction of any part of this document is strictly
prohibited without the prior written consent of Nagravision S.A.

Copyright ©2001-2003 Nagravision S.A. All rights reserved.
CH-1033 Cheseaux, Vaud, Switzerland.

First published, August 1999.
Revised, June 2004.
Part number: SasGwyStdSpe2.7.6

Nagravision S.A. is a member of the Kudelski Group of Companies.
Tel.: (41) (21) 732-0311 Fax: (41) (21) 732-0300

Security Policy of Nagravision S.A. (Kudelski Group)

Any recipient of this document, without exception is subject to a
Non Disclosure Agreement (NDA) of Nagravision S.A. (Kudelski Group) prior to delivery.

NOTICE

This document is supplied with an understanding that the notice(s) herein or any other contractual agreement(s) made that instigated the delivery of a hard copy, electronic copy, facsimile or file transfer of this document are strictly observed and maintained.

Polite notice and request to an unintended recipient

Should this document come into your possession and you are not the intended recipient: Nagravision kindly requests and thanks you in advance for making contact at your earliest convenience for instructions on how to proceed with its disposal.

CUSTOMER NAME

Contents–Summary

Contents–Full listing	v
Conventions used in this guide.....	x
Acknowledgements	xi
Printing or viewing online.....	xi
Your comments	xi
Acronyms and abbreviations.....	xii
Data formats	xiii
1. Introduction	1
2. System overview	3
3. SMS-SMSgw connections.....	9
4. SMSgw commands.....	13
5. Error codes	85
6. UA and CA-S/N checksum	93
7. ASCII Table	95
8. Examples	97

Contents–Full listing

Contents–Full listing	v
List of figures	ix
List of tables	ix
Conventions used in this guide.....	x
Pull-quotes	x
Convention for Windows	x
Acknowledgements	xi
Printing or viewing online.....	xi
Your comments	xi
Acronyms and abbreviations.....	xii
Data formats	xiii
1. Introduction	1
1.1 Audience	1
1.2 How to use this specification	1
1.3 License notice	1
1.4 Related documents	1
2. System overview	3
2.1 SMSgw purpose	3
2.2 Communication protocols	3
2.2.1 Layers.....	3
2.2.2 TCP/IP protocol	3
2.2.3 Device_IO protocol.....	4
2.2.4 SMSgw protocol	4
2.3 Device_IO communication	4
2.3.1 Overview.....	4
2.3.2 Establishing a connection with a Device_IO server	5
2.3.3 Data exchange between client and server.....	6
2.3.4 Closing a connection with a Device_IO server	6
2.3.5 Rules of use	6
2.3.6 Messages format.....	6
2.3.7 Message_1 (connect to DeviceIO server)	7
2.3.8 Message_2 (answer from DeviceIO server)	7
2.3.9 Message_3 (answer from SMSgw).....	8
2.3.10 Message_5 (message from SMS or message from SMSgw) ...	8
3. SMS-SMSgw connections.....	9
3.1 Overview.....	9
3.2 EMM and Control command flow	10
3.3 Feedback commands flow.....	10
3.4 Feedback commands routing.....	10

3.5	Rules of use	11
3.5.1	Source identifier	11
3.5.2	Transaction number	11
3.5.3	SMS connection is alive	11
3.5.4	SMS connection establishment	11
4.	SMSgw commands	13
4.1	Specification.....	13
4.1.1	Command-response	13
4.1.2	Asynchronous by nature	13
4.1.3	Feedback.....	13
4.2	Metrics	14
4.3	Time and date.....	14
4.4	Identifiers	14
4.5	Headers.....	15
4.5.1	Root header	15
4.5.2	Address header – EMM cmd	16
4.5.3	Address header – Control cmd	16
4.5.4	Address header – Feedback cmd	17
4.5.5	Address header – Operation cmd.....	17
4.5.6	List of errors applicable to the header sections.....	18
4.6	EMM commands (Onn)	19
4.6.1	Command 2: Add Product	19
4.6.2	Command 4: Product Suspension	21
4.6.3	Command 5: Product Reactivation	22
4.6.4	Command 6: Product Cancellation	23
4.6.5	Command 7: All products cancellation.....	24
4.6.6	Command 8: Credit management	25
4.6.7	Command 9: Update Credit Threshold	26
4.6.8	Command 10: Add Event Product	27
4.6.9	Command 13: Create Credit for Impulse Purchase.....	29
4.6.10	Command 14: Suspend impulse purchase	30
4.6.11	Command 15: Reactivate impulse purchase	31
4.6.12	Command 20: Suspend Subscriber ICC	32
4.6.13	Command 21: Reactivate Subscriber ICC	33
4.6.14	Command 35: Create Operator	34
4.6.15	Command 36: Cancel Operator	35
4.6.16	Command 48: Set Zip code	36
4.6.17	Command 49: Set Callback phone number.....	37
4.6.18	Command 50: Cancel ICC.....	38
4.6.19	Command 51: Initialize SmartCard	39
4.6.20	Command 52: Pair the ICC with the STB	40
4.6.21	Command 54: Set Callback IP address.....	41
4.6.22	Command 55: Set Callback Parameters.....	42
4.6.23	Command 56: Set PIN code	43
4.6.24	Command 60: Immediate Call Back.....	44

SMS Gateway Interface Specification

COMPANY CONFIDENTIAL

4.6.25	Command 61: Enable Automatic Call Back.....	45
4.6.26	Command 62: Disable Automatic Call Back.....	46
4.6.27	Command 69: Send Generic IRD Command	47
4.6.28	Command 71: Get Products	49
4.6.29	Command 96: Purge PPV and IPPV Records	50
4.6.30	Command 97: Set IPPV Records as Reported	52
4.7	MACRO EMM commands (9nn).....	53
4.7.1	Command 901: Activate SmartCard.....	53
4.7.2	Command 902: Activate SmartCard Without Return Path.....	56
4.7.3	Command 903: Add List of Products.....	58
4.8	CONTROL commands (1nn).....	60
4.8.1	Command 100: Redefine Credit Limit	60
4.8.2	Command 101: Set Authorized Phone Number	62
4.8.3	Command 105: Cancel ICC On Call-Collector	63
4.8.4	Command 106: Set Purchase Profile.....	64
4.8.5	Command 110: EMM cleanup	65
4.8.6	Command 111: Get History From Call-Collector	66
4.8.7	Command 120: Enable Callback Rules	67
4.8.8	Command 121: Disable Callback Rules.....	68
	FEEDBACK commands (2nn)	69
4.8.9	Command 200: Low credit alarm.....	70
4.8.10	Command 201: Current Debit and Credit	71
4.8.11	Command 202: PPV Purchase List	72
4.8.12	Command 203: PPV Purchase List Report	73
4.8.13	Command 204: PPV Purchase List With Payment Status	74
4.8.14	Command 205: Phone Discrepancies.....	75
4.8.15	Command 206: STU Responding Status.....	76
4.8.16	Command 207: ICC Memory Full Alarm.....	77
4.8.17	Command 211: Start of Report	78
4.8.18	Command 212: End of Report	79
4.8.19	Command 215: Products List.....	80
4.8.20	Command 216: PPV Purchase List Extended	81
4.9	OPERATION commands (10nn).....	82
4.9.1	Command 1000: Acknowledge Command.....	82
4.9.2	Command 1001: Non-acknowledged Command.....	82
4.9.3	Command 1002: No Command.....	83
5.	Error codes	85
5.1	Table of Error codes	85
5.2	Table of Error code extensions	87
5.3	Error codes and extensions for Command Root Header section.....	90
5.4	Error codes and extensions for Command Header section.....	91
6.	UA and CA-S/N checksum	93
6.1	Definitions	93
6.2	Pairing operation.....	93
6.3	Data files.....	93

CUSTOMER NAME

6.4 CA-S/N – Pairing keys 94

6.5 Checksum algorithm..... 94

7. ASCII Table 95

8. Examples 97

8.1 Device_IO connection establishment 97

8.2 Example of command (52) 101

List of figures

Fig. 2-1 CAS overview	3
Fig. 2-2 SMSgw protocol – communication layers	3
Fig. 2-3: Device_IO connection establishment and data exchange phases	6
Fig. 3-1: SMS – SMSgw connections	9
Fig. 3-2: Multi SMS connections	9




List of tables

Table 0-1 Acronyms, abbreviations and other terms	xii
Table 0-2 Data formats	xiii
Table 4-1 Identifiers	14
Table 5-1 Error codes	86
Table 5-2 Error code extensions	89

Conventions used in this guide

Pull-quotes

Pull-quotes are used in this document to clearly draw your attention to some parts of the text. See below for the pull-quotes used in this document. The name of the pull-quote or symbol is on the left (For clarity, these appear in the margins, clear of the main body text) and its purpose is detailed to the right.

CAUTION	Provides information to avoid undesirable effects or indicates that an operation or action could give unexpected results or is irreversible (e.g., data loss etc...).
Important	Information that must not be ignored when carrying out some task or tasks.
Note	Further information, advice or exceptions etc...
NOTICE	Information that Nagravision S.A. respectfully requires its customers and/or partners to observe.
	Technical points that only need to be read by technical staff.
	Provides information by way of a 'TIP' to carry out a task more effectively or efficiently.
	Indicates advice, which if not observed may result in injury and/or equipment damage.

Convention for Windows

Item	Description
Menu commands	In bold type: e.g., Select Save .
Field names, radio buttons and check boxes	In bold type: e.g., Select the Needs publishing check box.
Items selected in a list box	Items selected are shown inverted
Unselected items appear normal	Items unselected are shown without any treatment.

Acknowledgements

Trademarks: Any company's or product name(s) found herein may be the trademarks or registered trademarks of their respective companies.

Printing or viewing online

NOTICE It is strictly prohibited to print this document if it is marked "for online use only" or to disseminate this document with a screen dump/capture utility or similar tool or to view it on a machine that is not part of the System that it was supplied for use on.

Also, refer to the notice at the bottom of this document's front cover.

This document is supplied in Portable Document Format (PDF) format and it requires Adobe Acrobat Reader 3.0 (or later) to be printed (or viewed online). It is recommended to print this document double-sided on A4 paper (Also see note) using a laser printer. If your printer does not have double-sided mode—See your printer's documentation.

Note To print on letter size paper, in Acrobat reader, click **Print...** from the **file** menu (Print dialogue box appears), and then select **shrink to fit** check box.

Your comments

We at Nagravision make every endeavor to produce quality and accurate documentation to meet the needs of our customers. Therefore, if you as our valued customer have any suggestion or note a discrepancy in this document we would be pleased if you let us know for inclusion in a subsequent issue.

Important Please send comments to "publications@nagra.com" with subject "Feedback"

Acronyms and abbreviations

Term	Definition	Description
ANI	Automatic Number Identification	Also known as caller id
CAS	Conditional Access System	A generic term for a system used in pay television.
CC	Call-Collector	This is the sub-system of the CAS managing the callback coming from the STB
DVB	Digital Video Broadcasting	DVB is a family of international standards for all program delivery media: satellite, cable, terrestrial, microwave, MDS, CATV, and SMATV.
EBNF	Extended Backus-Naur Form	A formal mathematical way to describe a language
EMM	Entitlement Management Message	Carries data from the system to one or many smart cards.
ICC	Integrated Circuit Card	Smartcard
IPPV	Impulse Pay Per View	A PPV event product, which can be impulsively purchased through the STB and results in the acquisition of the related entitlement needed in the SMARTCARD.
ITM	Interactive Transaction Manager	This is the new name of the call-collector
MOP	Management Operator	The operator who manages end-users and entitlements.
NVOD	Near Video On Demand.	
PPV	Pay Per View	
Product		A single or group of services or events that may be purchased as a single entity.
SAS	Subscriber Authorization System	This is the sub-system of the CAS that converts the SMS command to EMM
SMS	Subscriber Management System	
SMSgw	SMS gateway	SMSgw is an application and an interface described in this document.
STB	Set-Top Box	The decoder installed at the end-user home
STU	See STB	
UA	Unique Address	This is the id or number of the smartcard
UTC	Coordinated Universal Time	Formerly known as GMT (Greenwich mean time).

Table 0-1 Acronyms, abbreviations and other terms

Data formats

Format	Description	Samples	Data
hex	Raw hexadecimal value	19 or 13(hex) in 2 bytes 88564006 or 05476126(hex) in 4 bytes	0x00:0x13 0x05:0x47:0x61:0x26
HHMMSS	time hour-min-sec represented in ASCII	102500 (10h25 and 00 sec)	0x31:0x30:0x32:0x35:0x30:0x30
ip_num	human representation of an IP address. The length is fixed to 15 bytes (4 x 3-digit num value separated with dot characters).	As an example the IP address 1.112.25.2 must be formatted as: 001.112.025.002	0x30:0x30:0x31:0x2E:0x31:0x31:0x32:0x2E 0x30:0x32:0x35:0x2E:0x30:0x30:0x30:0x32
num	numerical value represented in ASCII.	206	0x32:0x30:0x36
num_x	hexadecimal numerical value represented in ASCII.	6A10F9	0x36:0x3A:0x31:0x30:0x3F:0x39
r_num	numerical value represented in ASCII. The range is restricted.	see num samples	
r_num_x	hexadecimal value represented in ASCII. The range is restricted.	see num_x samples	
r_p_num	numerical value represented in ASCII and padded with "space" char. The range is restricted.	see p_num samples	
p_num	numerical value represented in ASCII and padded with "space" char	206__ (the value is a string of 5 digit, 3 significant digit and padded with 2 space)	0x32:0x30:0x36:0x20:0x20
r_text	any text represented in ASCII characters. The range is restricted.	N as No Y as Yes	0x4E 0x59
text	any text represented in ASCII characters.	SMS_GWY	0x53:0x4D:0x53:0x5F:0x47:0x57:0x59
YYYYMMDD	date year-month-day represented in ASCII.	20030518 (18 May 2003)	0x32:0x30:0x30:0x33:0x30:0x35:0x31:0x38

Table 0-2 Data formats

1. Introduction

This document presents the interface that links the SMS and the Nagravision conditional access system. It provides the specifications of the connection, the commands and the communication protocol used in the interface.

1.1 Audience

This guide is directed at the following people:

1. The customer's personnel involved in the management of the SMS – CAS interface
2. SMS vendor's personnel involved in the development of the interface between the SMS and the CAS.
3. Nagravision's personnel involved in the CAS (marketing, customer support, developer, test team)

1.2 How to use this specification

You do not need to read this specification from cover to cover—as some information is reference material only, therefore depending on your needs chose from the following :

- Chapter 2: System overview and communication protocols
- Chapter 3: SMS-SMSgw connection
- Chapter 4: The command format
- Chapter 5: The error codes

1.3 License notice

This document lists and describes all the commands supported by the Nagravision CAS interface. However, the usage of a given command depends on the business licenses acquired by the customer (i.e. the site operator). Therefore, make sure you have the appropriate license before using a command.

1.4 Related documents

- [1] “Information technology – Syntactic meta-language – Extended BNF” (ISO/IEC 14977:1996)

2. System overview

This chapter presents different aspects of the system related with the SMS gateway interface.

2.1 SMSgw purpose

The figure below illustrates the location of the SMS gateway (SMSgw). The SMSgw is an internal component of the Nagravision CAS system. From the outside world, it can be seen as a gate to the CAS. Through this gate, one or several SMS entities send instructions or commands towards the CAS. On the reverse way, the CAS sends information related to IPPV to a given SMS.

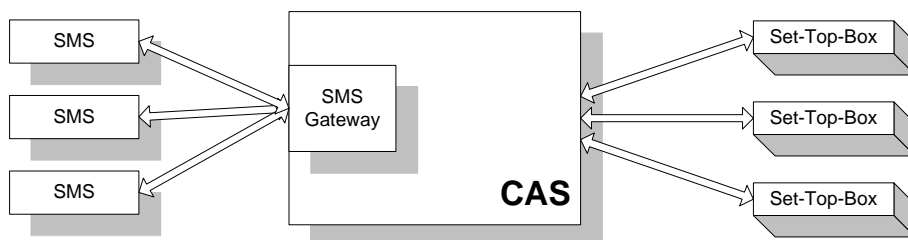


Fig. 2-1 CAS overview

2.2 Communication protocols

2.2.1 Layers

There are three communication layers. The lower level is the TCP/IP and the most abstract level is the SMSgw protocol. Between them, there is the Device IO layer whose role is to gather commands from the TCP/IP stream.

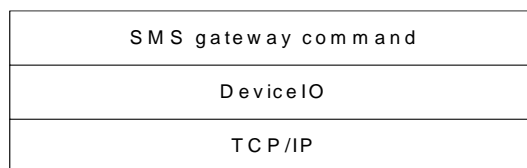


Fig. 2-2 SMSgw protocol – communication layers

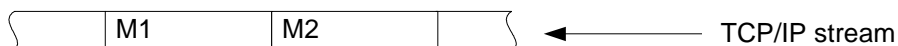
2.2.2 TCP/IP protocol

TCP/IP is a stream-based protocol. The application-oriented messages are joined together and there is no separator between these messages. When reading an application-oriented message from a stream, like TCP/IP, using OS primitives, we may have the three following possibilities:

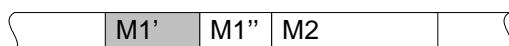
1. The message is incomplete (only n bytes of the data have been read)

2. The message is complete (only the expected message has been read)
3. More than the expected message has been read.

The diagram below shows these three possibilities: Let's assume that the TCP/IP stream contains two application-oriented messages M1 and M2.



Read an incomplete message:



Read one complete message:



Read more than one message:



2.2.3 Device_IO protocol

The Device_IO is a protocol above TCP/IP used to exchange data as a buffer of known size. It consists of a fixed size header followed by the application data. The header size is two bytes, and the application data is a stream of n bytes.

<u>Header section</u>	<u>Payload section</u>
Length (2 bytes)	Application data (n bytes)

The header is a 2-byte hexadecimal value that corresponds to the number of bytes of the payload section. Warning: only the payload section length is indicated in the header. The first transmitted byte is the most significant byte of the "Length value".

The payload section content is described in the following chapters.

2.2.4 SMSgw protocol

The SMSgw protocol defines the application-oriented message. It is a high level protocol for which the user data consists of human readable ASCII characters (from ASCII code 32 to 127) used to describe the value of the command attributes.

2.3 Device_IO communication

2.3.1 Overview

Communications are established through entry points called services. An application establishes a communication with another application by specifying the service name of the target.

For Device_IO communications between applications running on different machines, a communication

must first be established at the TCP/IP transport level.

To manage a communication between two systems the following questions must be addressed:

- How to establish the communication with a Device_IO server and how to indicate for which internal client the communication is intended.
- How to transmit and receive data to/from a Device_IO server.

In the following description, the commands used (open, send, receive, listen, close) are the commands of the underlying transport protocol used (TCP/IP). Device_IO does not redefine these commands. The names used below (open, send, receive, listen and close) are generic names representing the corresponding available system calls. Consequently, the calls described below only show their Device_IO parameters. Transport protocol parameters (like socket pointers for instance) are not shown in the generic description of the calls.

2.3.2 Establishing a connection with a Device_IO server

To open a communication with a Device_IO server, an application must call the Device_IO server. Two parameters must be provided:

1. The machine name on which the server is running
2. The port number (TCP-IP) corresponding to the Device_IO server.

Once the TCP communication with the server has been established, the name of the target service shall be communicated (message_1) to the Device_IO server in order to establish a link between the calling client and the requested service.

As a reply, the server shall send one or two messages. The first message (message_2) contains a connection status and, if the communication attempt has been successful, a second message (message_3) specifies whether the call has been accepted or rejected.

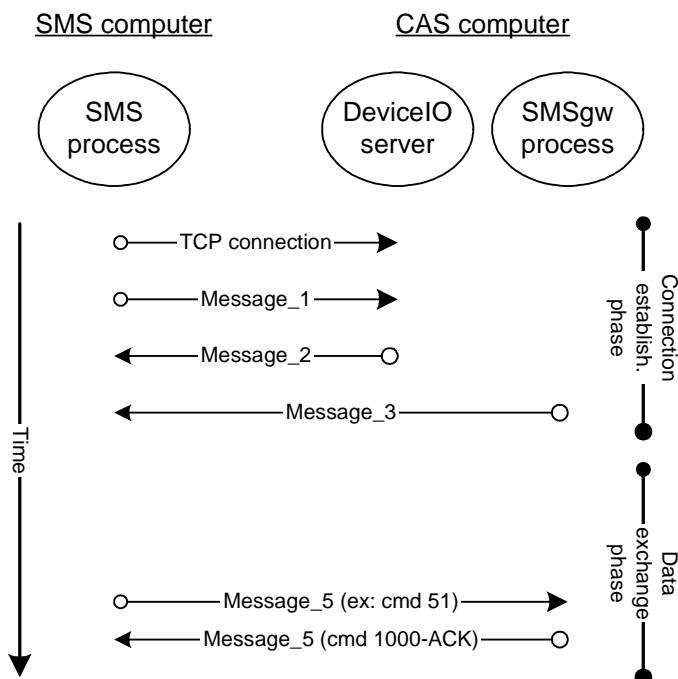


Fig. 2-3: Device_IO connection establishment and data exchange phases

An example of a Device_IO connection can be found in the chapter 6: Examples (on page 97).

2.3.3 Data exchange between client and server

After the communication link is established, messages can be exchanged (message_5) between the client (here, the SMS) and the Device_IO server.

2.3.4 Closing a connection with a Device_IO server

The connection is closed when the communication at the TCP level is closed.

2.3.5 Rules of use

If, for any reason, either the message_2 or the message_3 are not received within a specified time out (e.g. 30 sec) by the SMS at the establishment of the communication, the SMS should close the communication and retry later. The same rule applies if the connection status returned by message_2 is different from "SUCCESS" or if the answer code returned by message_3 indicates that the call is rejected.

2.3.6 Messages format

There are four different message types in the Device_IO protocol. The next sections present the different messages and how they should be utilized.

2.3.7 Message_1 (connect to DeviceIO server)

This is the first message that is sent by the client (the SMS) to the Device_IO server.

Syntax	Size (byte)	Format	Description
len	2	hex	Length in byte of the message (see 2.2.3).
op_mode	1	r_hex	Data transfer operation mode 0 = Normal data transfer
ob_name_len	1	hex	Length of the object name attribute (in bytes).
ob_name	ob_name_len	text	Name of the applicative service to which the connection should be established. Ob_name is a string of bytes at least one byte long and at most 32 bytes long: 1 <= ob_name_len <= 32. This Name is compulsory but its content is up to the client (ex: "SMS_GWY")

2.3.8 Message_2 (answer from DeviceIO server)

This message is a response from the Device_IO server to the client (in this case the SMS); this validates the connection.

MESSAGE_2			
Syntax	Size (byte)	Format	Description
Len	2	hex	Length in byte of the message. In this case, the length is always 1 byte.
connect_status	1	r_hex	Connection status. Refers to the table below for applicable value.

CONNECT_STATUS		
Value	Identifier	Description
0	CONNECT_FAILURE	The connection has failed for any unexpected reason
6	SUCCESS	The operation has been successfully completed.

2.3.9 Message_3 (answer from SMSgw)

This message is a response from the Device_IO server, which validates the connection with the other process (in our case: the SMSgw).

MESSAGE_3			
Syntax	Size (byte)	Format	Description
len	2	hex	Length in bytes of the message. In this case, the length is always 1 byte.
answer_code	1	r_num	0: call accepted 1: call rejected

2.3.10 Message_5 (message from SMS or message from SMSgw)

This message shall contain only one SMSgw command.

MESSAGE_5			
Syntax	Size (byte)	Format	Description
len	2	hex	Length in bytes of the message (see 2.2.3).
data	len	see note	SMSgw command data

The data section of the message_5 follows the structure shown below:

root header	common to all commands - see chapter 0
address header	depends on command type (EMM, CTRL, Feedback, Operation) - see chapters 4.5.2 to 4.5.5
command body	depends on command type - see chapters 4.6 to 4.9

Note

The content of this section is command specific, refers to the format described in chapters 4.7 to 4.11.

3. SMS-SMSGw connections

3.1 Overview

The figure below illustrates the connections utilized by the SMS and the Nagravision SMSgw. As depicted in the figure a connection can be seen as a data channel. On the same channel, the commands are on one way and on the opposite way there are the associated responses (“ack” or “Nack”). An “ack” response means that the incoming command is correctly formatted and it has been successfully processed. A “Nack” response means that either the format or the data structure of the command is not appropriate or the command has encountered some problems during the processing in the system. The SMS is responsible for opening both channels. The commands flow from the SMS to the CAS in the channel “EMM&Control” whereas the commands flow from the CAS to the SMS in the channel “Feedback command”.

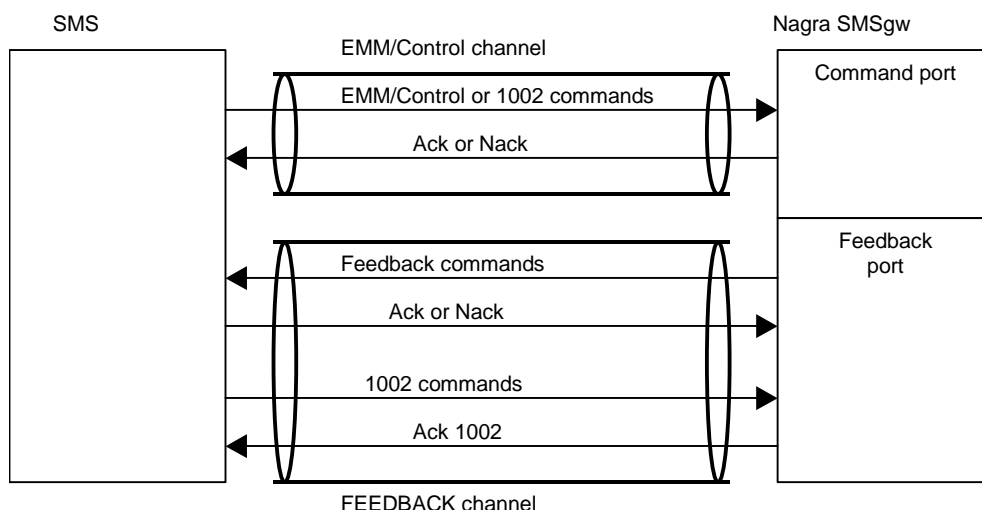


Fig. 3-1: SMS – SMSgw connections

Several SMS may be connected to the same port as shown in the following figure.

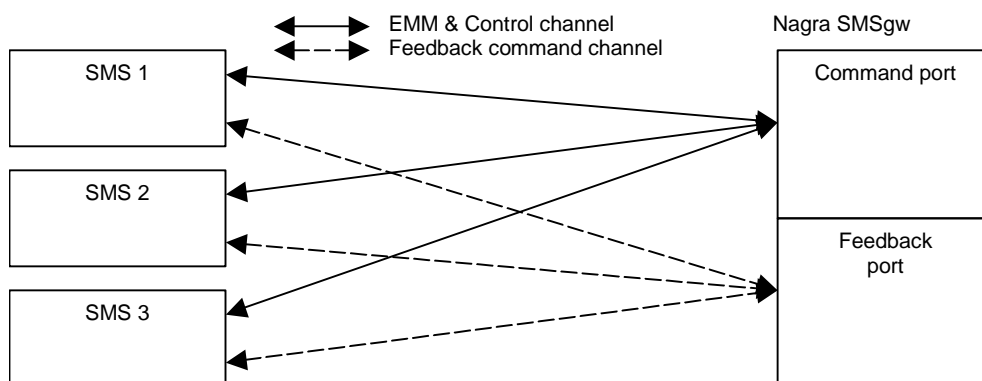


Fig. 3-2: Multi SMS connections

3.2 EMM and Control command flow

The *EMM and Control* commands are the commands sent by the SMS to the CAS. Some examples of such commands are “add product to a smartcard”, “create credit”, “set authorized phone number”.

3.3 Feedback commands flow

The *feedback* commands are the commands sent by the CAS to the SMS. The kind of information issued by the CAS is one of the following:

- List of impulsive purchases that have been made by the end-users
- Error condition in the smartcard (memory full, low credit)
- List of subscription products (reply of command 71: Get product)

3.4 Feedback commands routing

The routing of a *feedback* command means to which SMS the CAS should send a message. The type of message is described in the previous section.

The routing of a *feedback* command depends on the SMS *source_id* associated for a given SMARTCARD, i.e. for a given end-user. The *source_id*, available in the header of any command, is extracted by the CAS during the initialization phase of a end-user. Please refer to the **Important** note below for more details.

Any time an SMS opens a connection on a feedback channel and after the initial Device_IO handshake has been successful, the SMS must also issue a command 1002 in order to associate an *SMS_id* to the connection. The CAS uses that *SMS_id* to enable the routing of feedback commands only if appropriate SMS is up and running.

Important

- The *source_id* identifies an SMS; its value shall not change over time, for the system's lifetime. The *source_id* and *dest_id* MUST be distributed by Nagravision.
 - ITM-DNASP2 system: command 52 is used to catch SMS *source_id*
 - ITM-Aladin system: command 13 is used to catch SMS *source_id*
-

3.5 Rules of use

3.5.1 Source identifier

A source identifier is associated to each connection established between the SMS and the Nagravision CAS. Each connection must have its own source identifier and two connections may not share the same source identifier at any given time. The source identifier associated to a connection is given by the field "source_id" present in the header of each message sent from the SMS toward the CAS.

3.5.2 Transaction number

Each transaction number used must be unique during the life of a given connection. The transaction number is a field located in the header of each message sent from the SMS to the CAS. This is necessary to determine which commands have been processed successfully (ACK, command 1000) or not (NACK, command 1001).

3.5.3 SMS connection is alive

During periods of SMS-CAS interface inactivity, the SMS must send periodically a command 1002 on both ports (EMM/Control and feedback) to the CAS. The main reason for such a recommendation is that idle TCP connections may not be kept open forever by intelligent network devices (such as switches or firewalls). Nagravision recommends that the command 1002 be sent every 5 minutes on both ports.

3.5.4 SMS connection establishment

Any time the SMS opens a connection on either the EMM/Control port or on the Feedback port, it must first generate a command 1002.

4. SMSgw commands

4.1 Specification

4.1.1 Command-response

The SMS sends a command to the CAS. As a response, the CAS sends an acknowledgment message back to the SMS. The same occurs for feedback commands: the CAS sends a feedback command to the SMS. As a response, the SMS sends an acknowledgement back to the CAS. The acknowledgment messages are, of course, reported on the same communication channel by which the corresponding commands have been sent.

Note

The system will not behave as expected if the following values are not handled correctly:

- Transaction_id
- Source_id
- Dest_id

4.1.2 Asynchronous by nature

The protocol between the SMS and the CAS is an asynchronous protocol. Several SMS commands may be sent before acknowledgement messages are received. In other words, there is no need to wait for the acknowledgement of a given command before sending the next commands to the CAS.

Important

The acknowledgement messages may be received (by the SMS) in a different order than the order in which the corresponding commands have been sent. However for a given SMARTCARD, the sequence of EMM will follow the sequence of SMS commands.

Please note also that at the end of the transmission chain, i.e. the SMARTCARD, the sequence of command is not guaranteed.

4.1.3 Feedback

If there is no return path, then the SMS needs not open a connection on the feedback port, unless command 71 (Get Products) is sent on the control port to trigger the generations of commands 215 (Product List) on the feedback port.

4.2 Metrics

Metric	Value range (typical)
Connection	The <i>EMM&control</i> port and <i>feedback</i> ports may accept up to 10 connections each.
SMS commands	The CAS can process up to 10 to 20 SMS <i>EMM&control</i> commands per second.
Feedback commands	The CAS can generate and send up to 500 <i>feedback</i> commands per seconds. The flow of data is not necessarily smooth.

Note Those metrics depend on factors such as system architecture and processing power; they are provided here to give an order of magnitude.

4.3 Time and date

All dates and times must be in UTC.

4.4 Identifiers

This section contains descriptions of all identifiers used throughout this document:

ID	Definition
circuit_id	A number uniquely identifying a video/audio/data stream.
command_id	The identifier of an SMS command, this field is part of the command. It is documented by the SMS for all commands except the FEEDBACK commands that are documented by the IMS or the CC.
dest_id	Identifier of the addressed SMS command. It is entered at the time of system configuration.
IMS_event_id	An IMS generated identifier for each event. Entered through the IMS editor or generated when processing the EPG data feed.
IMS_product_id	The identifier of a product in the IMS. It is the only product identifier known to the IMS.
MOP_PPID	The identifier of the management operator. This identifier is provided by Nagravision at system configuration time.
SMS_product_id	A product identifier for the SMS. It is provided and managed by the SMS. It is entered in the system when the SMS Gateway through commands: 300, 303, 305 or 307.
source_id	An identifier for the source of SMS commands. This identifier is entered in the system at the time of system configuration.
STB serial number	The set-top box hardware serial number.
STU_number	This Nagravision STB number is used to identify the set-top box in the CA system for pairing purposes.

Table 4-1 Identifiers

4.5 Headers

The structure of any message is as follows:

root header	common to all commands (chapter 0)			
address header	EMM 4.5.2	CTRL 4.5.3	Feedback 4.5.4	Operation 4.5.5
command body	cmd 0nn 4.6	cmd 1nn 4.7	cmd 2nn 0	cmd 1000 4.9

The chapters below details the internal structure of the 3 modules (root header, address header and command body).

4.5.1 Root header

Field	Size	Format	Description
transaction_number	9	num	Number used to uniquely identify a transaction across the interface for each source. range: 000000000 to 999999999
command_type	2	r_num	01: EMM 02: CONTROL 04: FEEDBACK 05: OPERATION.
source_id	4	num	A number that identifies a source such as the SMS or IMS (this number is provided by Nagravision) range: 0000 to 9999
dest_id	4	num	Identifier of the addressed SMS. This number is defined at system configuration. range: 0000 to 9999
MOP_PPID	5	num	Identifier of the technical management operator. This number is provided at system configuration by: Nagravision. range: 00000 to 99999
creation_date	8	YYYYMMDD	Creation date of the command (UTC).

List of error codes (NACK messages) are listed in section 4.5.6.

4.5.2 Address header – EMM cmd

Field	Size	Format	Description
broadcast_mode	1	r_text	N = Normal: standard broadcasting mode. B = Batch: command with a lower priority
broadcast_start_date	8	YYYYMMDD	Broadcast start date (UTC). The command must be sent to the SMARTCARD from this date. See note.
broadcast_end_date	8	YYYYMMDD	Broadcast end date (UTC). The command must be sent to the smartcard until this date. See note.
address_type	1	r_text	EMM addressing mode for EMM commands U = Unique G = Global (all smartcard of the MOP are addressed).
UA	10 or 0	num	UA is the Unique Address of the smartcard for which the command is intended. If the “address_type” field = G, then this field is void.

Note

The Nagravision CAS system can be configured to set the broadcast period of the EMM by using either the broadcast dates provided by the SMS (mode A) or the CAS system entirely manages the broadcast dates (mode B). In this mode B, the SMS should set the broadcast dates as “today”. Nagravision recommends setting-up the CAS system in mode B.

List of error codes (NACK messages) are listed in section 4.5.6.

4.5.3 Address header – Control cmd

Field	Size	Format	Description
broadcast_mode	1	r_text	N = Normal: standard broadcasting mode. B = Batch: command with a lower priority
broadcast_start_date	8	YYYYMMDD	Current date (today, UTC)
broadcast_end_date	8	YYYYMMDD	Current date (today, UTC)
address_type	1	r_text	EMM addressing mode for EMM commands U = Unique
UA	10	num	UA is the Unique Address of the smartcard for which the command is intended. ex: 2018327794

List of error codes (NACK messages) are listed in section 4.5.6.

4.5.4 Address header – Feedback cmd

Field	Size	Format	Description
UA	10	num	UA is the Unique Address of the smartcard for which the command is intended. ex: 2018327794

List of error codes (NACK messages) are listed in section 4.5.6.

4.5.5 Address header – Operation cmd

Field	Size	Format	Description
none			There is no header for this command type.

List of error codes (NACK messages) are listed in section 4.5.6.

4.5.6 List of errors applicable to the header sections

Field	Error codes	Error codes extension
transaction_number	BAD_HEADER_SYNTAX	BAD_TRANSACTION_NUMBER_FORMAT
transaction_number	BAD_USAGE	TRANS_NR_ALREADY_IN_USE
command_type	BAD_HEADER_SYNTAX	BAD_COMMAND_TYPE
source_ID	BAD_HEADER_SYNTAX	BAD_SOURCE_ID
source_ID	BAD_USAGE	SOURCE_NOT_AUTHORIZED
source_ID	BAD_USAGE	SOURCE_ALREADY_IN_USE
dest_ID	BAD_HEADER_SYNTAX	BAD_DEST_ID
dest_ID	BAD_USAGE	DEST_NOT_AUTHORIZED
MOP_PPID	BAD_HEADER_SYNTAX	BAD_MOP_PPID
MOP_PPID	INVALID_PPID	MOP_NOT_AUTHORIZED
creation_date	BAD_HEADER_SYNTAX	BAD_DATE_FORMAT
creation_date	BAD_USAGE	DATE_IN_THE_FUTURE
broadcast_mode	BAD_HEADER_SYNTAX	BAD_BROADCAST_MODE
broadcast_start_date	BAD_HEADER_SYNTAX	BAD_DATE_FORMAT
broadcast_end_date	BAD_HEADER_SYNTAX	BAD_DATE_FORMAT
broadcast_end_date	BAD_HEADER_SYNTAX	BAD_DATE_SEQUENCE
broadcast_end_date	BAD_HEADER_SYNTAX	DATE_IN_THE_PAST
address_type	BAD_HEADER_SYNTAX	BAD_ADDRESS_TYPE
address_type	BAD_USAGE	ADDRESS_TYPE_NOT_AUTHORIZED
UA	BAD_HEADER_SYNTAX	BAD_UA_FORMAT
command_ID	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_ID	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT

4.6 EMM commands (0nn)

In this section, we present the structure of all commands related to the alteration of the smartcard. The SMS will generate these commands. For each command, there are two tables. The first table illustrates the data structure of the command. The second table provides the different types of error message that could be returned by the CAS.

4.6.1 Command 2: Add Product

	privilege
sop	
mop	x

The SMS uses this command to add, in a smartcard, an entitlement of services associated to a product.

Important

Command 2 shall not be used to authorize a PPV. The command 10 should be used instead.

COMMAND 2: ADD PRODUCT			
Field	byte	Format	Description
command_id	4	r_num	command_id = 0002
IMS_product_id	12	num	IMS product ID range: 000000000000 to 999999999999
begin_date	8	YYYYMMDD	Subscription begin date (UTC). Subscription is not valid before this date.
end_date	8	YYYYMMDD	Subscription end date (UTC). Subscription is not valid after this date.

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
IMS_product_id	BAD_COMMAND_SYNTAX	BAD_IMS_PRODUCT_ID_FORMAT
IMS_product_id	PRODUCT_NOT_FOUND	NO_EXTENDED_ERROR_CODE
IMS_product_id	BAD_PRODUCT_TYPE	PPV_PRODUCT
IMS_product_id	BAD_PRODUCT_STATUS	CANCELLED_PRODUCT
IMS_product_id	BAD_PRODUCT_STATUS	SUSPENDED_PRODUCT
IMS_product_id	BAD_PRODUCT_STATUS	INVALID_PURCHASE_DATE
IMS_product_id	BAD_PRODUCT_TYPE	DRAFT_PRODUCT
begin_date	BAD_COMMAND_SYNTAX	BAD_DATE_FORMAT
begin_date	BAD_COMMAND_SYNTAX	BAD_DATE_SEQUENCE
end_date	BAD_COMMAND_SYNTAX	BAD_DATE_FORMAT
end_date	BAD_COMMAND_SYNTAX	BAD_DATE_SEQUENCE
UA	UA_NOT_FOUND	NO_EXTENDED_ERROR_CODE
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE
MOP_PPID	INVALID_PPID	ISD_MOP_NOT_FOUND
MOP_PPID	DATABASE_ERROR	DATA_ERROR

4.6.2 Command 4: Product Suspension

	privilege
sop	
mop	x

The SMS uses this command to suspend, in a smartcard, an entitlement of services or PPVs associated to a product.

The end-user will not be able to watch the corresponding services or PPVs until a *product reactivation* command is sent by the SMS. This command may be used when there is a payment problem with the end-user. This command does not impact callbacks.

COMMAND 4: PRODUCT SUSPENSION			
Field	byte	format	Description
command_id	4	r_num	command_id = 0004
IMS_product_id	12	num	IMS product ID range: 000000000000 to 999999999999

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
IMS_product_id	BAD_COMMAND_SYNTAX	BAD_IMS_PRODUCT_ID_FORMAT
IMS_product_id	PRODUCT_NOT_FOUND	NO_EXTENDED_ERROR_CODE
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE

4.6.3 Command 5: Product Reactivation

	privilege
sop	
mop	x

The SMS uses this command to reactivate, in a smartcard, an entitlement of services or PPVs that have been previously suspended (with a command 4: *Product Suspension*). The end-user will be able to watch the corresponding services or PPVs again.

COMMAND 5: PRODUCT REACTIVATION			
Field	byte	format	Description
command_id	4	r_num	command_id = 0005
IMS_product_id	12	num	IMS Product ID range: 000000000000 to 999999999999

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
IMS_product_id	BAD_COMMAND_SYNTAX	BAD_IMS_PRODUCT_ID_FORMAT
IMS_product_id	PRODUCT_NOT_FOUND	NO_EXTENDED_ERROR_CODE
IMS_product_id	RIGHT_NOT_FOUND	NO_EXTENDED_ERROR_CODE
UA	UA_NOT_FOUND	NO_EXTENDED_ERROR_CODE
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE

4.6.4 Command 6: Product Cancellation

	privilege
sop	
mop	x

The SMS uses this command to remove, from a smartcard, an entitlement of services or PPVs associated to a product.

This command is applicable for PPV entitlements that have been purchased through the SMS only. In other words, any IPPV entitlements cannot be cancelled with this command.

The cancellation of PPV entitlements has no impact on the end-user's credit in the smartcard.

COMMAND 6: PRODUCT CANCELLATION			
Field	byte	format	Description
command_id	4	r_num	command_id = 0006
IMS_product_id	12	num	IMS Product ID range: 000000000000 to 999999999999

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
IMS_product_id	BAD_COMMAND_SYNTAX	BAD_IMS_PRODUCT_ID_FORMAT
IMS_product_id	PRODUCT_NOT_FOUND	NO_EXTENDED_ERROR_CODE
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE

4.6.5 Command 7: All products cancellation

	privilege
sop	
mop	x

The SMS uses this command to remove, from a smartcard, all types of entitlements (services and PPVs). This command will also remove all non-watched IPPVs. However, it does not affect non-call collected IPPVs that have been watched. Such IPPVs will be call-collected when the next callback occurs.

Important

This command will also **suspend the IPPV purchases** (equivalent to command 14, "Suspend Impulse Purchase")

COMMAND 7: ALL PRODUCTS CANCELLATION			
Field	byte	format	Description
command_id	4	r_num	command_id = 0007

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE
address_type	BAD_USAGE	ADDRESS_TYPE_NOT_AUTHORIZED
MOP_PPID	INVALID_PPID	ISD_MOP_NOT_FOUND

4.6.6 Command 8: Credit management

	privilege
sop	
mop	x

The SMS uses this command to modify, in a smartcard, the data of an existing credit record. The credit record in a smartcard is mandatory to allow end-users purchasing impulse-pay-per-view (IPPs).

Important

- The credit information in a smartcard can be updated every 4 seconds.
- Only the mode SET CREDIT is authorized.
- The initial credit value in the smartcard is set with the *command 13: Create credit for impulse purchase*.

The *credit* data field in the command shall not exceed the maximum value of 65535.99. This is taken care of by the Call-Collector.

COMMAND 8: CREDIT MANAGEMENT			
Field	byte	format	Description
command_id	4	r_num	command_id = 0008
credit_mode	2	r_num	03 = SET CREDIT. Set the new credit value.
credit	7	r_num	Credit amount (in the local currency) representing the Range: 00000.00 to 65535.99

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
credit_mode	BAD_COMMAND_SYNTAX	BAD_CREDIT_MODE
credit	BAD_COMMAND_SYNTAX	BAD_CREDIT_FORMAT
UA	UA_NOT_FOUND	NO_EXTENDED_ERROR_CODE
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE
MOP_PPID	ACCOUNT_NOT_FOUND	NO_EXTENDED_ERROR_CODE

4.6.7 Command 9: Update Credit Threshold

	privilege
sop	
mop	x

The SMS uses this command to update, in a smartcard, the impulse *credit threshold* value of the end-user smartcard. The *credit threshold* value allows the smartcard to trigger callbacks depending on the credit status.

This command **should not be used** to initialize the Credit Threshold value. The initialization is done implicitly with *command 13: Create Credit For Impulse Purchase*.

If the *credit threshold* value is set to 0 (zero), then no callback will be issued by the smartcard.

COMMAND 9: UPDATE CREDIT THRESHOLD			
Field	byte	format	Description
command_id	4	r_num	command_id = 0009
threshold_credit	7	r_num	Lower limit under which the smartcard must do a low credit call back representing the range 00000.00 to 65535.99

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
threshold_credit	BAD_COMMAND_SYNTAX	BAD_THRESHOLD_CREDIT_FORMAT
threshold_credit	BAD_COMMAND_SYNTAX	CREDIT_THRESHOLD_TOO_HIGH
UA	UA_NOT_FOUND	NO_EXTENDED_ERROR_CODE
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE
MOP_PPID	ACCOUNT_NOT_FOUND	NO_EXTENDED_ERROR_CODE

4.6.8 Command 10: Add Event Product

	privilege
sop	
mop	x

The SMS uses this command to add, in a smartcard, an entitlement to an *event product*. An *event product* is typically a movie showing on a given channel at a given date/time. This command is typically used to add PPV entitlements for end-users who cannot impulsively purchase PPV products.

When an event product is purchased through the SMS (by calling customer service), the smartcard credit and debit values are not impacted.

Important

- The event product is pre-flagged as “call collected” in the smartcard. Consequently, the purge mechanism will delete an event product once the purge date condition matches the event product date.
- For systems without return path, the deletion of PPV entitlements recorded in the end-user’s smartcards should be done either by background EMM cleanup command or by SMS *command 96: Purge PPV and IPPV Records*. For systems with return path, the deletion of PPV entitlements should be done either by background EMM cleanup command or by CAS CCM command generated during callbacks.
- The data in the *event_name* field of the command are not altered by the CAS. The data is transmitted as it is in the smartcard. Then the STB will display the data as it does with the EIT (Event Information Table) data. Consequently, the SMS must format this string in accordance with the specification of the charset used by the STB.
- The *event_name* string must also include control character. This means that the number of displayable characters is reduced consequently.

COMMAND 10: ADD EVENT PRODUCT			
Field	byte	format	Description
command_id	4	r_num	command_id = 0010
IMS_product_id	12	num	IMS Product ID range: 000000000000 to 999999999999
length_event_name	2	r_num	Length of valid data in <i>event_name</i> field. The number of bytes should not exceed 30. This is due to the smartcard storage limitation.
event_name	32	text	Event name as displayed in the PPV purchase history in the STB user interface. The number of characters must match the length <i>length_event_name</i> . The remaining bytes should be filled-up with 0x00.
price	5	num	Price of the product, representing 000.00 to 999.99

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
IMS_product_id	BAD_COMMAND_SYNTAX	BAD_IMS_PRODUCT_ID_FORMAT
IMS_product_id	PRODUCT_NOT_FOUND	NO_EXTENDED_ERROR_CODE
IMS_product_id	PPV_IN_THE_PAST	NO_EXTENDED_ERROR_CODE
IMS_product_id	BAD_PRODUCT_TYPE	REGULAR_PRODUCT
IMS_product_id	BAD_PRODUCT_STATUS	CANCELLED_PRODUCT
IMS_product_id	BAD_PRODUCT_STATUS	SUSPENDED_PRODUCT
IMS_product_id	BAD_PRODUCT_STATUS	INVALID_PURCHASE_DATE
IMS_product_id	BAD_PRODUCT_TYPE	DRAFT_PRODUCT
length_event_name	BAD_COMMAND_SYNTAX	LENGTH_TOO_LONG
length_event_name	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
price	BAD_COMMAND_SYNTAX	BAD_PRICE_FORMAT
UA	UA_NOT_FOUND	NO_EXTENDED_ERROR_CODE
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE
MOP_PPID	INVALID_PPID	ISD_MOP_NOT_FOUND
MOP_PPID	DATABASE_ERROR	DATA_ERROR

4.6.9 Command 13: Create Credit for Impulse Purchase

	privilege
sop	
mop	x

The SMS uses this command to add a credit data structure in a smartcard. This will allow the end-user to perform impulse PPV purchases.

The SMS should send this command only once at the initialization phase of the impulsive purchase capability of the smartcard. Any other credit management should be done with command 8: *credit management*.

Important

- There is no command to remove the credit data structure in a given smartcard.
- If threshold_credit is equal to 0 (zero), then no callback will be issued by the smartcard.

COMMAND 13: CREATE CREDIT FOR IMPULSE PURCHASE			
Field	byte	format	Description
command_id	4	r_num	command_id = 0013
credit	7	r_num	Credit amount set in the smartcard. Warning: the range is 00000.00 to 65535.99.
threshold_credit	7	r_num	Lower limit under which the smartcard must initiate a low credit call back. It represents 00000.00 to 65535.99. Note: The smartcard will truncate this value, i.e. it will only consider the integer part.

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
credit	BAD_COMMAND_SYNTAX	BAD_CREDIT_FORMAT
threshold_credit	BAD_COMMAND_SYNTAX	BAD_THRESHOLD_CREDIT_FORMAT
threshold_credit	BAD_COMMAND_SYNTAX	CREDIT_THRESHOLD_TOO_HIGH
UA	UA_NOT_FOUND	NO_EXTENDED_ERROR_CODE
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE
MOP_PPID	INVALID_PPID	ISD_MOP_NOT_FOUND
MOP_PPID	DATABASE_ERROR	DATA_ERROR

4.6.10 Command 14: Suspend impulse purchase

	privilege
sop	
mop	x

The SMS uses this command to suspend, in a smartcard, the ability of an end-user to perform impulse purchases.

The reactivation of impulse purchases may be completed using *command 15: Reactivate impulse purchase*. This command has no impact on callback operations.

COMMAND 14: SUSPEND IMPULSE PURCHASE			
Field	byte	format	Description
command_id	4	r_num	command_id = 0014

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE

4.6.11 Command 15: Reactivate impulse purchase

	privilege
sop	
mop	x

The SMS uses this command to reactivate, in a smartcard, the impulse purchase capability of an end-user. This command is used to activate the impulse purchase capability of an end-user smartcard that has previously received a command 14: *Suspend impulse purchase* or a command 20: *Suspend end-user smartcard*.

COMMAND 15: REACTIVATE IMPULSE PURCHASE			
Field	byte	format	Description
command_id	4	r_num	command_id = 0015

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE
UA	UA_NOT_FOUND	NO_EXTENDED_ERROR_CODE
MOP_PPID	ACCOUNT_NOT_FOUND	NO_EXTENDED_ERROR_CODE

4.6.12 Command 20: Suspend Subscriber ICC

	privilege
sop	
mop	x

The SMS uses this command to suspend, in a smartcard, all entitlements of services and PPVs as well as the impulse purchase capability. However, the “*free access*” services can still be watched.

The entitlements of impulsive PPVs already purchased are also suspended.

Important

In order to reactivate the end-user smartcard, the SMS should use *command 21* and *command 15*.

The SMS can still continue to send EMM command to a deactivated smartcard (no error returned). This command does not impact callback operations.

COMMAND 20: SUSPEND SUBSCRIBER ICC			
Field	byte	Format	Description
command_id	4	r_num	command_id = 0020

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE
MOP_PPID	INVALID_PPID	ISD_MOP_NOT_FOUND

4.6.13 Command 21: Reactivate Subscriber ICC

	privilege
sop	
mop	x

The SMS uses this command to reactivate all entitlements of services and PPVs. in a smartcard

Important

In order to reactivate the impulsive purchase feature of the end-user smartcard, the SMS should send *command 15* also.

COMMAND 21: REACTIVATE SUBSCRIBER ICC			
Field	byte	Format	Description
command_id	4	r_num	command_id = 0021

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
UA	UA_NOT_FOUND	NO_EXTENDED_ERROR_CODE
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE
MOP_PPID	INVALID_PPID	ISD_MOP_NOT_FOUND

4.6.14 Command 35: Create Operator

	privilege
sop	x
mop	

The SMS uses this command to add a new *Management Operator* (MOP) zone in a smartcard.

This command is used ONLY if there are several operators sharing smartcards to do PayTV business independently.

This command must precede any other command (like *cmd 2: add product*, *cmd 13: create credit*) for a given operator.

The SMS can send this command several times without damaging the entitlements already stored in the smartcard.

Important

- In a single MOP environment, this command should not be used because any new smartcard already contains the pre-defined MOP zone for the unique operator.
- The purpose of this command is not to “virgin” a smartcard, i.e. this command does not perform any cleanup in the smartcard.

COMMAND 35: CREATE OPERATOR			
Field	byte	format	Description
command_id	4	r_num	command_id = 0035
MOP_PPID	5	num	Identifier of the management operator to be created. range: 00000 to 99999

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
MOP_PPID	DATABASE_ERROR	DATA_ERROR
UA	UA_NOT_FOUND	NO_EXTENDED_ERROR_CODE
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE

4.6.15 Command 36: Cancel Operator

	privilege
sop	x
mop	

The SMS uses this command to remove, from a smartcard, a Management Operator (MOP) zone and all its associated data.

Important

This command will indirectly remove all entitlements and credit record currently stored in the smartcard for a given operator.

COMMAND 36: CANCEL OPERATOR			
Field	byte	format	Description
command_id	4	r_num	command_id = 0036
MOP_PPID	5	num	Identifier of the management operator to be created. range: 00000 to 99999

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
MOP_PPID	DATABASE_ERROR	DATA_ERROR
UA	UA_NOT_FOUND	NO_EXTENDED_ERROR_CODE
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE

4.6.16 Command 48: Set Zip code

	privilege
sop	x
mop	

The SMS uses this command to set or update the Zip code of the smartcard. The Zip code information is primarily used for blackout and time zone management.

Important

For countries that use a different zip code scheme (i.e. more digits or alpha/num zip code), the operator should create a lookup map table that translates local zip code format to this zip code format.

COMMAND 48: SET ZIP CODE			
Field	byte	format	Description
command_id	4	r_num	command_id = 0048
zip_code	5	num	end-user's zip code. Range: 00000...99999

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
zip_code	BAD_COMMAND_SYNTAX	BAD_ZIP_CODE_FORMAT
zip_code	ZIP_NOT_FOUND	NO_EXTENDED_ERROR_CODE
UA	UA_NOT_FOUND	NO_EXTENDED_ERROR_CODE
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE

4.6.17 Command 49: Set Callback phone number

	privilege
sop	x
mop	

The SMS uses this command to set or update the parameter “phone-number” in the smartcard. This parameter allows the smartcard/STB to open a connection with the Call-Collector.

The parameter can be a phone number or a MAC address. The STB will determine how to interpret this number. Here are two examples:

- Phone number 1-800-555-1212, cc_number_1 is 18005551212 (with 5 trailing spaces).
- MAC address 00-06-5B-BC-8F-92, cc_number_1 is in decimal as 27308887954 (with 5 trailing spaces).

To reset the phone number in the smartcard, the two following ways are accepted:

- cc_number_1 is 16 space characters (ASCII 0x20)
- cc_number_1 is 16 F characters (ASCII 0x46)

Important

- No character (ex: “-“ or “/”) must be present in the field cc_number_1
- To set an IP address in the smartcard, the SMS must send a command 54 (Set Callback IP address).

COMMAND 49: SET CALLBACK PHONE NUMBER			
Field	byte	format	Description
command_id	4	r_num	command_id = 0049
cc_number_1	16	p_num	Call-Collector phone number. This field must be right padded with trailing ASCII space characters.

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
cc_number_1	BAD_COMMAND_SYNTAX	BAD_PHONE_NUMBER_FORMAT
UA	UA_NOT_FOUND	NO_EXTENDED_ERROR_CODE
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE
MOP_PPID	CB_PROFILE_NOT_FOUND	NO_EXTENDED_ERROR_CODE

4.6.18 Command 50: Cancel ICC

	privilege
sop	x
mop	

The SMS uses this command to permanently disable a smartcard. Such command is performed when a smartcard is removed from operation (for instance the smartcard is declared as stolen, lost, or damaged).

The SMS is not allowed to use a “cancelled” smartcard again because the smartcard *is not recoverable* after such operation.

The smartcard is flagged as cancelled and deactivated in the CAS database. Consequently, any incoming commands addressed to this smartcard will be NON-ACKnowledged by the CAS system.

Important

The effect of this command in the smartcard is NOT reversible. In other words, the smartcard will be permanently out-of-order.

COMMAND 50: CANCEL ICC			
Field	byte	format	Description
command_id	4	r_num	command_id = 0050

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
address_type	BAD_USAGE	ADDRESS_TYPE_NOT_AUTHORIZED

4.6.19 Command 51: Initialize SmartCard

	privilege
sop	x
mop	

The SMS uses this command to initialize the Management Operator (MOP) zone of a new smartcard.

The SMS must issue this command prior to any other command. The CAS will reject any other command while this *command 51* is not successfully processed.

The initialization of a smartcard can occur several times and has no effect on the entitlements stored in the smartcard.

Important

- This command must be the first command issued by the SMS for a new smartcard
- The purpose of this command is not to “reset” the smartcard content, i.e. this command does not perform any cleanup of the smartcard.
- This command does not perform any refurbishing process of a smartcard.

COMMAND 51: INITIALIZE SMARTCARD			
Field	byte	format	Description
command_id	4	r_num	command_id = 0051

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
UA	UA_OUT_OF_RANGE	NO_EXTENDED_ERROR_CODE
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE
MOP_PPID	CB_PROFILE_NOT_FOUND	NO_EXTENDED_ERROR_CODE

4.6.20 Command 52: Pair the ICC with the STB

	privilege
sop	x
mop	

The SMS uses this command to pair a smartcard with the STB. This command is mandatory before any services may be authorized. The smartcard must have previously been initialized with *command 51* before pairing can take place.

This command may be issued several times to pair a smartcard with a different STB.

Important

- The STU_number is a 14 digits string representing a decimal value
- The STU_number can be one of the following formats:
 - a) 10 digit string followed by four space characters in the range “0000000000 “ to “4294967295 “
 - b) 14 digit string in the range 00000000000000 to 00004294967295
- The “un-pairing“ action is performed with the same command 52, but with STU_number “0000000000 “ or 00000000000000
- The STB_number provided in the command by the SMS must not include any checksum.

COMMAND 52: PAIR THE ICC WITH THE STB			
Field	byte	format	Description
command_id	4	r_num	command_id = 0052
STU_number	14	r_num	CA STB serial number in decimal. range: see “important” note above

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
STU_number	BAD_COMMAND_SYNTAX	BAD_STU_NUMBER_FORMAT
UA	UA_NOT_FOUND	NO_EXTENDED_ERROR_CODE
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE

4.6.21 Command 54: Set Callback IP address

	privilege
sop	
mop	x

The SMS uses this command to set or update the smartcard field containing the IP-address of the Call-Collector.

COMMAND 54: SET CALLBACK IP ADDRESS			
Field	byte	format	Description
command_id	4	r_num	command_id = 0054
CC_IP_address	15	ip_num	Call-Collector IP address. Field format is 000.000.000.000 to 255.255.255.255.
CC_IP_port	5	num	Call-Collector TCP/IP port.

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
CC_IP_address	BAD_COMMAND_SYNTAX	BAD_IP_ADDRESS_FORMAT
CC_IP_port	BAD_COMMAND_SYNTAX	BAD_CC_PORT_FORMAT
UA	UA_NOT_FOUND	NO_EXTENDED_ERROR_CODE
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE
MOP_PPID	CB_PROFILE_NOT_FOUND	NO_EXTENDED_ERROR_CODE

4.6.22 Command 55: Set Callback Parameters

	privilege
sop	
mop	x

The SMS uses this command to set or update the smartcard field containing the IP-address and phone-number of the Call-Collector.

COMMAND 55: SET CALLBACK PARAMETERS			
Field	byte	format	Description
command_id	4	r_num	command_id = 0055
phone_field	1	r_num	Smartcard Phone field selector (for future use) Must be set to 1 for the moment.
CC_number_1	16	p_num	Call-Collector phone number. This field must be right padded with trailing ASCII space characters.
CC_IP_address	15	ip_num	Call-Collector IP address. Field format is 000.000.000.000 to 255.255.255.255.
CC_IP_port	5	num	Call-Collector TCP/IP port. As an example port 1458 must be formatted as "01458"
callback_type	1	r_num	0: phone & IP. This is used when the return path is a phone line and the connection is PPP. 1: phone only. This is used when the return path is a phone line and the connection is not PPP. 2: IP only. This is used when the return path is a bi-directional cable.

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
CC_number_1	BAD_COMMAND_SYNTAX	BAD_PHONE_NUMBER_FORMAT
CC_IP_address	BAD_COMMAND_SYNTAX	BAD_IP_ADDRESS_FORMAT
CC_IP_port	BAD_COMMAND_SYNTAX	BAD_CC_PORT_FORMAT
callback_type	BAD_COMMAND_SYNTAX	BAD_CALLBACK_FORMAT
UA	UA_NOT_FOUND	NO_EXTENDED_ERROR_CODE
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE
MOP_PPID	CB_PROFILE_NOT_FOUND	NO_EXTENDED_ERROR_CODE

4.6.23 Command 56: Set PIN code

	privilege
sop	x
mop	

The SMS uses this command to set or update one of the 16 PIN code values that can be stored in the smartcard.

The PIN code is a 4 digits numeric value. The CAS performs no checks other than boundaries on the PIN code, i.e. the same PIN code can be updated many times. The CAS does not hold the PIN code in its database.

Important

- The usage of this command is customer specific. Thus, a document illustrating how this feature works end-to-end should be prepared.
- This command is not available for any type of smartcard.

COMMAND 56: SET PIN CODE			
Field	byte	format	Description
command_id	4	r_num	command_id = 0056
Index	2	r_num	PIN code number. Range: 01-16
PIN	4	num	PIN code value

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
Index	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
Index	VALUE_OUT_OF_RANGE	NO_EXTENDED_ERROR_CODE
PIN	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
UA	UA_NOT_FOUND	NO_EXTENDED_ERROR_CODE
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE

4.6.24 Command 60: Immediate Call Back

	privilege
sop	
mop	x

The SMS uses this command to force a smartcard to “immediately” call back the CAS. Due to the architecture of the system, an immediate call means that the CAS expects to receive a call back from a given smartcard in a time frame from a couple of seconds to several hours. The best case depends on the setting of the CAS and on the STB behavior regarding such call back. The worst case is several hours; this is due to the behavior of the STB. Indeed, when a STB performs a call back, and if the call fails for any reason (e.g. the line is busy), the STB will try again within a period of a couple of minutes to several hours.

COMMAND 60: IMMEDIATE CALL BACK			
Field	byte	format	Description
command_id	4	r_num	command_id = 0060
CbDate	8	YYYYMMDD	Warning: the value provided by the SMS is ignored. Thus, this value should be the date of <i>today</i> .
CbTime	6	HHMMSS	Warning: the value provided by the SMS is ignored. Thus, this value should be 000000.

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
UA	UA_NOT_FOUND	NO_EXTENDED_ERROR_CODE
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE
MOP_PPID	CB_PROFILE_NOT_FOUND	NO_EXTENDED_ERROR_CODE

4.6.25 Command 61: Enable Automatic Call Back

	privilege
sop	
mop	x

The SMS uses this command to activate the automatic feedback functionality of a smartcard. Please refer to section 4.10 (Feedback commands) for more details related to the information received by the SMS when a smartcard/STB performs a call back.

COMMAND 61: ENABLE AUTOMATIC CALL BACK			
Field	byte	format	Description
command_id	4	r_num	command_id = 0061
call_freq	2	r_num_x	01 = annual 02 = semi-annual 03 = quarterly 04 = monthly 05 = semimonthly 1m = every m days (1 should be considered as a flag) m range is 1 .. F (F meaning 15 days)
date_first_call	8	YYYYMMDD	First date (UTC) on which the smartcard should call back.
CbTime	6	HHMMSS	Warning: the value provided by the SMS is ignored. Consequently, the value should be 000000.

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
call_freq	BAD_COMMAND_SYNTAX	BAD_FREQUENCY_FORMAT
date first call	BAD_COMMAND_SYNTAX	BAD_DATE_FORMAT
date_first_call	BAD_USAGE	BAD_DATE_SEQUENCE
UA	UA_NOT_FOUND	NO_EXTENDED_ERROR_CODE
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE
MOP_PPID	CB_PROFILE_NOT_FOUND	NO_EXTENDED_ERROR_CODE

4.6.26 Command 62: Disable Automatic Call Back

	privilege
sop	
mop	x

The SMS uses this command to deactivate the automatic call back functionality of a smartcard.

The call back triggered by memory full, credit threshold limit reached, special events, or as a consequence of an immediate callback command will still occur.

COMMAND 62: DISABLE AUTOMATIC CALL BACK			
Field	byte	format	Description
command_id	4	r_num	command_id = 0062

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
UA	UA_NOT_FOUND	NO_EXTENDED_ERROR_CODE
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE
MOP_PPID	CB_PROFILE_NOT_FOUND	NO_EXTENDED_ERROR_CODE

4.6.27 Command 69: Send Generic IRD Command

	privilege
sop	
mop	x

The SMS uses this command to send an “IRD command” to a decoder. The decoder will execute an operation based on the “IRD command” instruction (ex: reset PIN code, set network ID, force tune, etc.). In other words, this command is not intended to modify the data of the smartcard, but it is intended to start an action executed by the decoder. The CAS provides with this command a secure transport mechanism between the head-end and the STB.

The data structure that is sent to the STB is described below. The SMS Gateway calculates the fields “sequence_number” and “checksum” for the convenience of SMS. It means that the SMS does not provide these two parameters.

```
command_body{
    sequence_number    32 uimsbf
    command_id         8  uimsbf
    operation          8  uimsbf
    for (i=0;i<N;i++){
        data           8  uimsbf
    }
    checksum           8  uimsbf
}
```

As explained above, the purpose of an “IRD command” is to provide the decoder with a message that will conduct the decoder to execute an action (ex: reset PIN code). The “command_id” and “operation” parameters provided to the decoder represent two indexes that will indicate which subroutine should be executed by the decoder.

Nagravision provides the STB manufacturers with a list of “command_id” and “operation” that map a list of given actions. This document is referenced as “Conditional Access Kernel – IRD Command Specification”.

COMMAND 69: SEND GENERIC IRD COMMAND			
Field	byte	format	Description
command_id	4	r_num	command_id = 0069
IRD_command_id	3	r_um	command_id field of IRD command_body. Range: 000 to 255
IRD_operation	3	r_num	Operation field of IRD command_body. Range: 000 to 255
IRD_data_length	2	r_num	Length in bytes of useful part of IRD_data field Range: 00 to 48
IRD_data	96	r_text (see note below)	Hexadecimal data coded in ASCII format. The complete string must be transferred (96 chars representing 48 bytes). However, only the first left IRD_data_length bytes will be included in the data field of the IRD command_body.

Important

As indicated in the table above, the format of the IRD_data field is r_text. However, this data is not a common string, i.e. it is not a text field such as “this is a sentence with characters and 1 2 3 4 numerical values“. The data in this field is a sequence of hexadecimal number coded in text format. Consequently, only alpha characters A, B, C, D, E, F representing the number 10 to 15 in hexadecimal are accepted.

As an example the following IRD_data is accepted by the CAS:

“A0BCD12A29327B9F”

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
IRD_command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
IRD_operation	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
IRD_data_length	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
IRD_data	BAD_COMMAND_SYNTAX	BAD_DATA_FORMAT
UA	UA_NOT_FOUND	NO_EXTENDED_ERROR_CODE
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE

4.6.28 Command 71: Get Products

	privilege
sop	
mop	x

The SMS uses this command to retrieve all the non-impulsive products (subscriptions or PPV) known by the CAS for a given smartcard (i.e. a given end-user).

The information comes from the CAS database. The products are returned asynchronously in command 215: Products List on the feedback port.

Impulsively purchased products can be retrieved in the same way using command 111: Get History From Call-Collector.

COMMAND 71: GET PRODUCTS			
Field	byte	format	Description
command_id	4	r_num	command_id = 0071

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
UA	UA_NOT_FOUND	NO_EXTENDED_ERROR_CODE
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE

4.6.29 Command 96: Purge PPV and IPPV Records

	privilege
sop	
mop	x

The SMS uses this command to purge some old PPV and IPPV records in individual smartcards. There are two use cases to illustrate how this command works:

Example 1: to purge some old PPV and IPPV records stored in a smartcard when an end-user complains that its smartcard is full. As a result, the end-user will be able to proceed to future IPPV purchases.

Example 2: to avoid the smartcard to be filled up with useless PPV and IPPV records. This can be seen as a preventive cleanup task. In this example, any time an end-user calls the SMS center to ask for additional credit, the SMS should issue this command 96.

Important

This command will purge only IPPV records set as “call collected”. The command 97 should be use in conjunction with the command 96 to delete old IPPV records.

The SMS provides the two following parameters:

- *cleanup_date* is the date by which PPV or IPPV records older than this date will be deleted.
- *condition_date* is a parameter with a contextual meaning. It allows differentiating STB with or without a return path. For any STB with return path capability, the *condition date* = 19920101. For other STB, the *condition date* = *cleanup_date*

The SMS is responsible for correctly managing these parameters because the CAS system cannot verify the coherence of these parameters.

COMMAND 96: PURGE PPV AND IPPV RECORDS			
Field	byte	format	Description
command_id	4	r_num	command_id = 0096
cleanup_date	8	YYYYMMDD	Delete any PPV-IPPV record stored in a smartcard if the PPV-IPPV expiration date is older than the cleanup_date.
condition_date	8	YYYYMMDD	- return path exists, date = 19920101 - return path does not exist, date = cleanup_date

List of error codes (in NACK messages) applicable for this section of the message.

SMS Gateway Interface Specification

COMPANY CONFIDENTIAL

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
cleanup_date	BAD_COMMAND_SYNTAX	BAD_DATE_FORMAT
cleanup_date	BAD_COMMAND_SYNTAX	DATE_NOT_IN_THE_PAST
condition_date	BAD_COMMAND_SYNTAX	BAD_DATE_FORMAT
UA	UA_NOT_FOUND	NO_EXTENDED_ERROR_CODE
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE
MOP_PPID	INVALID_PPID	ISD_MOP_NOT_FOUND

4.6.30 Command 97: Set IPPV Records as Reported

	privilege
sop	
mop	x

The SMS uses this command to set some IPPV records on a given smartcard, as reported. This command should be used to address ONLY decoders working in *prepaid without return path* mode. A decoder without return path means a decoder that is not connectable to a Nagravision Call-Collector.

Important

This command should only be used in pre-paid mode environment

This command should be used by the SMS to pre-cleanup IPPV in the smartcard. There are two typical examples illustrating how to use this command. The objective is the same; to avoid the smartcard to be filled up with useless IPPV records. This can be seen as a preventive cleanup task. What differentiates the two examples is the condition that triggers the operation:

Example 1: In this scenario, the SMS periodically and automatically sends the command 97 to specific smartcards.

Example 2: In this scenario, any time an end-user asks for additional credit (prepaid mode), the SMS issues this command 97.

COMMAND 97: SET IPPV RECORDS AS REPORTED			
Field	byte	format	Description
command_id	4	r_num	command_id = 0097
collect_date	8	YYYYMMDD	Set any IPPV record stored in a smartcard as reported (flag "call-collected" = 1) if the IPPV expiration date is older than or equal to the parameter "collect_date".

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
collect_date	BAD_COMMAND_SYNTAX	BAD_DATE_FORMAT
collect_date	BAD_COMMAND_SYNTAX	DATE_NOT_IN_THE_PAST
UA	UA_NOT_FOUND	NO_EXTENDED_ERROR_CODE
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE
MOP_PPID	INVALID_PPID	ISD_MOP_NOT_FOUND

4.7 MACRO EMM commands (9nn)

This chapter describes the data of the command section for the 9nn SMS command series. Those commands are the same kind as the 0nn SMS command series (see previous chapter). The aim of this series is to optimize the overall command message bandwidth of the SMS - CAS interface. Indeed, for a large volume of end-users, the amount of transactions (EMM command messages) is very high. Thus the utilization of the interface should be optimized. The solution adopted here is to concatenate several commands of the 0nn series in one *macro* command.

4.7.1 Command 901: Activate SmartCard

	privilege
sop	x
mop	

This command is a 'one-shot' initialization command designed to improve the performances of the CAS when initializing a new smartcard in the CAS database.

Important

- Please read the **Important** note of command 2, 10, 13, 48, 49, 52, 61, 100, 101
- If the CAS must generate a negative acknowledgement (i.e. NACK) upon processing this SMS command, the body of the command will be truncated to 999 characters long before being inserted in the field 'command_section' of the negative acknowledgement.

COMMAND 901: ACTIVATE SMARTCARD			
Field	byte	format	Description
command_ID	4	r_num	command_ID = 901
zip_code	5	num	end-user's zip code. Range: 00000...99999
STU_number	14	r_num	CA STB serial number in decimal. range: see important note in section 4.6.20
credit	7	r_num	Credit amount (in the local currency) representing the range: 00000.00 to 65535.99
threshold_credit	7	r_num	Lower limit under which the smartcard must do a low credit call back representing the range 0.00 to 65535.99
credit_limit	7	r_num	Credit limit within the range 00000.00 to 65535.99

SMS Gateway Interface Specification

COMPANY CONFIDENTIAL

call_freq	2	r_num_x	01 = annual 02 = semi-annual 03 = quarterly 04 = monthly 05 = semimonthly 1m = every m days (1 should be considered as a flag) m range is 1 .. F (F meaning 15 days)
date_first_call	8	YYYYMMDD	First date (UTC) on which the smartcard should call back.
cc_number_1	16	p_num	Call-Collector phone number. This field must be right padded with trailing ASCII space characters.
phone_number_1	16	p_num	Primary phone number for the call-collector. This field must be padded with trailing ASCII space characters for numbers not requiring 16 digits.
phone_number_2	16	p_num	Alternate phone number for the call-collector.
phone_number_3	16	p_num	Alternate phone number for the call-collector.
nb_of_products	2	r_num	Number of products provided in this command. The range is 00 to 99
for(i=0;i<nb_of_products;i++) {			Counter
IMS_product_ID	12	num	IMS product ID Range: 000000000000 to 999999999999
begin_date	8	YYYYMMDD	Subscription begin date (UTC). Subscription is not valid before this date.
end_date	8	YYYYMMDD	Subscription end date (UTC). Subscription is not valid after this date.
}			
nb_of_ppv	2	r_num	Number of PPV product provided in this command. The range is 00 to 99
for(i=0;i<nb_of_ppv;i++) {			Counter
IMS_product_ID	12	num	IMS product ID range: 000000000000 to 999999999999
length_event_name	2	r_num	Length of valid data in <i>event_name</i> field. The number of bytes should not exceed 30. This is due to the smartcard storage limitation.
event_name	32	text	Event name as displayed in the PPV purchase history in the STB user interface. The number of characters must match the length <i>length_event_name</i> . The remaining bytes should be filled-up with 0x00.
price	5	num	Price of the product, representing 0.00 to 999.99
}			

CUSTOMER NAME

List of error codes sent by the CAS (NACK messages).

Field	Error codes	Error codes extension
command_ID	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE
UA	UA_OUT_OF_RANGE	NO_EXTENDED_ERROR_CODE
MOP_PPID	DATABASE_ERROR	DATA_ERROR (invalid parent MOP)
MOP_PPID	CB_PROFILE_NOT_FOUND	NO_EXTENDED_ERROR_CODE
zip_code	BAD_COMMAND_SYNTAX	BAD_ZIP_CODE_FORMAT
zip_code	ZIP_NOT_FOUND	NO_EXTENDED_ERROR_CODE
STU_number	BAD_COMMAND_SYNTAX	BAD_STU_NUMBER_FORMAT
credit	BAD_COMMAND_SYNTAX	BAD_CREDIT_FORMAT
threshold credit	BAD_COMMAND_SYNTAX	BAD_THRESHOLD_CREDIT_FORMAT
threshold credit	BAD_COMMAND_SYNTAX	CREDIT_THRESHOLD_TOO_HIGH
credit_limit	BAD_COMMAND_SYNTAX	BAD_CREDIT_FORMAT
call_freq	BAD_COMMAND_SYNTAX	BAD_FREQUENCY_FORMAT
date_first_call	BAD_COMMAND_SYNTAX	BAD_DATE_FORMAT
date_first_call	BAD_USAGE	BAD_DATE_SEQUENCE
cc_number_1	BAD_COMMAND_SYNTAX	BAD_PHONE_NUMBER_FORMAT
phone_number_1	BAD_COMMAND_SYNTAX	BAD_PHONE_NUMBER_FORMAT
phone_number_2	BAD_COMMAND_SYNTAX	BAD_PHONE_NUMBER_FORMAT
phone_number_3	BAD_COMMAND_SYNTAX	BAD_PHONE_NUMBER_FORMAT
nb_of_products	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
IMS_product_ID	BAD_COMMAND_SYNTAX	BAD_IMS_PRODUCT_ID_FORMAT
IMS_product_ID	PRODUCT_NOT_FOUND	NO_EXTENDED_ERROR_CODE
IMS_product_ID	BAD_PRODUCT_TYPE	PPV_PRODUCT
IMS_product_ID	BAD_PRODUCT_TYPE	REGULAR_PRODUCT
IMS_product_ID	BAD_PRODUCT_TYPE	DRAFT_PRODUCT
IMS_product_ID	BAD_PRODUCT_STATUS	CANCELLED_PRODUCT
IMS_product_ID	BAD_PRODUCT_STATUS	SUSPENDED_PRODUCT
IMS_product_ID	BAD_PRODUCT_STATUS	INVALID_PURCHASE_DATE
IMS_product_ID	PPV_IN_THE_PAST	NO_EXTENDED_ERROR_CODE
begin_date	BAD_COMMAND_SYNTAX	BAD_DATE_FORMAT
begin_date	BAD_COMMAND_SYNTAX	BAD_DATE_SEQUENCE
end_date	BAD_COMMAND_SYNTAX	BAD_DATE_FORMAT
end_date	BAD_COMMAND_SYNTAX	BAD_DATE_SEQUENCE
length_event_name	BAD_COMMAND_SYNTAX	LENGTH_TOO_LONG
length_event_name	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
price	BAD_COMMAND_SYNTAX	BAD_PRICE_FORMAT

4.7.2 Command 902: Activate SmartCard Without Return Path

	privilege
sop	x
mop	

This command is a 'one-shot' initialization command designed to improve the performances of the CAS when initializing a new smartcard in the CAS database.

Important

- Please read the **Important** note of command 2, 10, 13, 48, 49, 52, 61, 100, 101
- If the CAS must generate a negative acknowledgement (i.e. NACK) upon processing this SMS command, the body of the command will be truncated to 999 characters long before being inserted in the field 'command_section' of the negative acknowledgement.

COMMAND 902: ACTIVATE SMARTCARD WITHOUT RETURN PATH			
Field	byte	format	Description
command_ID	4	r_num	command_ID = 902
zip_code	5	num	End-user's zip code. Range: 00000...99999
STU_number	14	r_num	CA STB serial number in decimal. Range: see important note in section 4.6.20
nb_of_products	2	r_num	Number of products provided in this command. The range is 00 to 99
for(i=0;i<nb_of_products;i++) {			Counter
IMS_product_ID	12	num	IMS product ID range: 000000000000 to 999999999999
begin_date	8	YYYYMMDD	Subscription begin date (UTC). Subscription is not valid before this date.
end_date	8	YYYYMMDD	Subscription end date (UTC). Subscription is not valid after this date.
}			
nb_of_ppv	2	num	IMS product ID Range: 000000000000 to 999999999999
for(i=0;i<nb_of_ppv;i++) {			Counter
IMS_product_ID	12	num	IMS product ID range: 000000000000 to 999999999999
length_event_name	2	r_num	Length of valid data in <i>event_name</i> field. The number of bytes should not exceed 30. This is due to the smartcard storage limitation.

SMS Gateway Interface Specification

COMPANY CONFIDENTIAL

event_name	32	text	Event name as displayed in the PPV purchase history in the STB user interface. The number of characters must match the length <i>length_event_name</i> . The remaining bytes should be filled-up with 0x00.
price	5	num	Price of the product, representing 0.00 to 999.99
}			

List of error codes sent by the CAS (NACK messages).

Field	Error codes	Error codes extension
command_ID	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE
UA	UA_OUT_OF_RANGE	NO_EXTENDED_ERROR_CODE
MOP_PPID	DATABASE_ERROR	DATA_ERROR (invalid parent MOP)
MOP_PPID	CB_PROFILE_NOT_FOUND	NO_EXTENDED_ERROR_CODE
zip_code	BAD_COMMAND_SYNTAX	BAD_ZIP_CODE_FORMAT
zip_code	ZIP_NOT_FOUND	NO_EXTENDED_ERROR_CODE
STU_number	BAD_COMMAND_SYNTAX	BAD_STU_NUMBER_FORMAT
nb_of_products	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
IMS_product_ID	BAD_COMMAND_SYNTAX	BAD_IMS_PRODUCT_ID_FORMAT
IMS_product_ID	PRODUCT_NOT_FOUND	NO_EXTENDED_ERROR_CODE
IMS_product_ID	BAD_PRODUCT_TYPE	PPV_PRODUCT
IMS_product_ID	BAD_PRODUCT_TYPE	REGULAR_PRODUCT
IMS_product_ID	BAD_PRODUCT_TYPE	DRAFT_PRODUCT
IMS_product_ID	BAD_PRODUCT_STATUS	CANCELLED_PRODUCT
IMS_product_ID	BAD_PRODUCT_STATUS	SUSPENDED_PRODUCT
IMS_product_ID	BAD_PRODUCT_STATUS	INVALID_PURCHASE_DATE
IMS_product_ID	PPV_IN_THE_PAST	NO_EXTENDED_ERROR_CODE
begin_date	BAD_COMMAND_SYNTAX	BAD_DATE_FORMAT
begin_date	BAD_COMMAND_SYNTAX	BAD_DATE_SEQUENCE
end_date	BAD_COMMAND_SYNTAX	BAD_DATE_FORMAT
end_date	BAD_COMMAND_SYNTAX	BAD_DATE_SEQUENCE
length_event_name	BAD_COMMAND_SYNTAX	LENGTH_TOO_LONG
length_event_name	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
price	BAD_COMMAND_SYNTAX	BAD_PRICE_FORMAT

4.7.3 Command 903: Add List of Products

	privilege
sop	
mop	x

The SMS uses this command to add a list of new products (service products, non-impulsively purchasable event products and package products) into a smartcard.

Important

- Please read the **Important** note of command 2, 10, 13, 48, 49, 52, 61, 100, 101
- If the CAS must generate a negative acknowledgement (i.e. NACK) upon processing this SMS command, the body of the command will be truncated to 999 characters long before being inserted in the field 'command_section' of the negative acknowledgement.

COMMAND 903: ADD LIST OF PRODUCTS			
Field	byte	format	Description
command_ID	4	r_num	command_ID = 903
nb_of_products	2	r_num	Number of products provided in this command. The range is 00 to 99
for(i=0;i<nb_of_products;i++) {			Counter
IMS_product_ID	12	num	IMS product ID Range: 000000000000 to 999999999999
begin_date	8	YYYYMMDD	Subscription begin date (UTC). Subscription is not valid before this date.
end_date	8	YYYYMMDD	Subscription end date (UTC). Subscription is not valid after this date.
}			
nb_of_ppv	2	r_num	Number of PPV product provided in this command. The range is 00 to 99
for(i=0;i<nb_of_ppv;i++) {			Counter
IMS_product_ID	12	num	IMS product ID range: 000000000000 to 999999999999
length_event_name	2	r_num	Length of valid data in <i>event_name</i> field. The number of bytes should not exceed 30. This is due to the smartcard storage limitation.
event_name	32	text	Event name as displayed in the PPV purchase history in the STB user interface. The number of characters must match the length <i>length_event_name</i> . The remaining bytes should be filled-up with 0x00.
price	5	num	Price of the product, representing 0.00 to 999.99
}			

SMS Gateway Interface Specification

COMPANY CONFIDENTIAL

List of error codes sent by the CAS (NACK messages).

Field	Error codes	Error codes extension
command_ID	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
UA	UA_NOT_FOUND	NO_EXTENDED_ERROR_CODE
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE
MOP_PPID	INVALID_PPID	ISD_MOP_NOT_FOUND
MOP_PPID	DATABASE_ERROR	DATA_ERROR (invalid parent MOP)
nb_of_products	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
IMS_product_ID	BAD_COMMAND_SYNTAX	BAD_IMS_PRODUCT_ID_FORMAT
IMS_product_ID	PRODUCT_NOT_FOUND	NO_EXTENDED_ERROR_CODE
IMS_product_ID	BAD_PRODUCT_TYPE	PPV_PRODUCT
IMS_product_ID	BAD_PRODUCT_TYPE	REGULAR_PRODUCT
IMS_product_ID	BAD_PRODUCT_TYPE	DRAFT_PRODUCT
IMS_product_ID	BAD_PRODUCT_STATUS	CANCELLED_PRODUCT
IMS_product_ID	BAD_PRODUCT_STATUS	SUSPENDED_PRODUCT
IMS_product_ID	BAD_PRODUCT_STATUS	INVALID_PURCHASE_DATE
IMS_product_ID	PPV_IN_THE_PAST	NO_EXTENDED_ERROR_CODE
begin_date	BAD_COMMAND_SYNTAX	BAD_DATE_FORMAT
begin_date	BAD_COMMAND_SYNTAX	BAD_DATE_SEQUENCE
end_date	BAD_COMMAND_SYNTAX	BAD_DATE_FORMAT
end_date	BAD_COMMAND_SYNTAX	BAD_DATE_SEQUENCE
length_event_name	BAD_COMMAND_SYNTAX	LENGTH_TOO_LONG
length_event_name	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
price	BAD_COMMAND_SYNTAX	BAD_PRICE_FORMAT

4.8 CONTROL commands (1nn)

The series of commands 1nn are utilized to set in the Call-Collector module data related with callback generated by the STB. The data provided in the command 1nn is stored in the Call-Collector database. This means that the data is persistent.

4.8.1 Command 100: Redefine Credit Limit

	privilege
sop	
mop	x

The SMS uses this command to set or update for a given end-user, the amount of credit that will be added in the smartcard during any callback, in the Call-Collector database.

The Call-Collector will update automatically the new credit in the smartcard ONLY at the next callback of the smartcard-STB.

The new credit in the smartcard will be the addition of the credit_limit and the current smartcard debit.

Important

- The credit_limit field of the command is a 7-digit value. However, the smartcard reads this value as a 5-digit integer unit, followed by 2-digit representing “cents”. Furthermore, the greatest value applicable in the smartcard is 65535.99. As a consequence, for countries where the currency contains several 0 (ex: Yen), the credit value should be considered as a divided currency value, i.e. divided by 100 or 1000.
- In a Pay-TV site with pre-paid mode and with STB with return path, the credit_limit should be set to 00000.00. This means that the smartcard credit will not be updated during any callback.

COMMAND 100: REDEFINE CREDIT LIMIT			
Field	byte	format	Description
command_id	4	r_num	command_id = 0100
credit_limit	7	r_num	credit limit range: 00000.00 to 65535.99

List of error codes (in NACK messages) applicable for this section of the message.

SMS Gateway Interface Specification

COMPANY CONFIDENTIAL

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
credit_limit	BAD_COMMAND_SYNTAX	BAD_CREDIT_FORMAT
UA	UA_NOT_FOUND	NO_EXTENDED_ERROR_CODE
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE
MOP_PPID	ACCOUNT_NOT_FOUND	NO_EXTENDED_ERROR_CODE

4.8.2 Command 101: Set Authorized Phone Number

	privilege
sop	
mop	x

The SMS uses this command to set or update the phone numbers of a given end-user, in the Call-Collector database. Those numbers are the authorized phone numbers from where a smartcard is allowed to perform any callback.

The caller id (i.e. the phone number from where the smartcard-STB is calling) will be checked with those authorized phone numbers upon each callback. The CAS will generate the warning *command 205* if the phone number used during the callback is not authorized.

COMMAND 101: SET AUTHORIZED PHONE NUMBER			
Field	byte	format	Description
command_id	4	r_num	command_id = 0101
phone_number_1	16	p_num	Primary phone number. This field must be padded with trailing ASCII space characters for numbers not requiring 16 digits.
phone_number_2	16	p_num	Alternate phone number. This field must be padded with trailing ASCII space characters for numbers not requiring 16 digits.
phone_number_3	16	p_num	Alternate phone number. This field must be padded with trailing ASCII space characters for numbers not requiring 16 digits.

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
phone_number_1	BAD_COMMAND_SYNTAX	BAD_PHONE_NUMBER_FORMAT
phone_number_2	BAD_COMMAND_SYNTAX	BAD_PHONE_NUMBER_FORMAT
phone_number_3	BAD_COMMAND_SYNTAX	BAD_PHONE_NUMBER_FORMAT
UA	UA_NOT_FOUND	NO_EXTENDED_ERROR_CODE
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE
MOP_PPID	CB_PROFILE_NOT_FOUND	NO_EXTENDED_ERROR_CODE

4.8.3 Command 105: Cancel ICC On Call-Collector

	privilege
sop	
mop	x

The SMS uses this command to cancel an existing smartcard record in the Call-Collector database. This command should be issued when the smartcard is deleted from the SMS database.

Important

The different records for a given end-user in the CAS database are set as cancelled. As a result of this command, the CAS will not accept any callback from STB. Furthermore, a cancelled smartcard on the Call-Collector will not report a command 206 (STB responding status).

COMMAND 105: CANCEL ICC ON CALL-COLLECTOR			
Field	byte	format	Description
command_id	4	r_num	command_id = 0105

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
UA	UA_NOT_FOUND	NO_EXTENDED_ERROR_CODE
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE
MOP_PPID	CB_PROFILE_NOT_FOUND	NO_EXTENDED_ERROR_CODE
MOP_PPID	ACCOUNT_NOT_FOUND	NO_EXTENDED_ERROR_CODE

4.8.4 Command 106: Set Purchase Profile

	privilege
sop	
mop	x

The SMS uses this command to set or update the attributes of the subscriber's purchase profile in the Call-Collector database.

COMMAND 106: SET PURCHASE PROFILE			
Field	byte	format	Description
command_id	4	r_num	command_id = 106
category	3	r_num	Subscriber's category. Range: 000 to 255
enable_online_purchase	1	r_text	Flag specifying whether online purchases (i.e. any transaction performed through the return path) must be accepted or not. Value is either 'Y' (accepted) or 'N' (forbidden).
enable_PPV_purchase	1	r_text	Flag specifying whether PPV purchases must be accepted or not. . This applies to both online and SMS purchase modes. Value is either 'Y' (accepted) or 'N' (forbidden).
enable_automatic_payment	1	r_text	Flag specifying whether online purchases should debited automatically from the subscriber's account or not. Value is either 'Y' (accepted) or 'N' (forbidden).

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
category	BAD_COMMAND_SYNTAX	INVALID_CATEGORY
enable_online_purchase	BAD_COMMAND_SYNTAX	BAD_FLAG_VALUE
enable_PPV_purchase	BAD_COMMAND_SYNTAX	BAD_FLAG_VALUE
enable_automatic_payment	BAD_COMMAND_SYNTAX	BAD_FLAG_VALUE
UA	UA_NOT_FOUND	NO_EXTENDED_ERROR_CODE
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE
MOP_PPID	CB_PROFILE_NOT_FOUND	NO_EXTENDED_ERROR_CODE

4.8.5 Command 110: EMM cleanup

	privilege
sop	
mop	x

The SMS uses this command to “delete” all EMMs in the CAS and EMM broadcaster for a given smartcard. This command should be used if a discrepancy is suspected between SMS customer data and EMM for this customer.

COMMAND 110: EMM CLEANUP			
Field	byte	format	Description
command_id	4	r_num	command_id = 0110

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
UA	UA_NOT_FOUND	NO_EXTENDED_ERROR_CODE
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE

4.8.6 Command 111: Get History From Call-Collector

	privilege
sop	
mop	x

The SMS uses this command to retrieve a two-month history of IPPV information stored in the Call-Collector database. The response from the CAS system will be generated on the *Feedback* channel. Each PPV record will generate a separate Command 202 (PPV Purchase List).

COMMAND 111: GET HISTORY FROM CALL-COLLECTOR			
Field	byte	format	Description
command_id	4	r_num	command_id = 0111

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
UA	UA_NOT_FOUND	NO_EXTENDED_ERROR_CODE
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE

4.8.7 Command 120: Enable Callback Rules

	privilege
sop	
mop	x

The SMS uses this command to activate, in the Call-Collector, the processing of the callback rules for a given smartcard.

During the next callback from the STB, the CALL-COLLECTOR will check the PPV purchase history of the corresponding end-user against the various callback rules defined into the CAS database. The first matching rule will (possibly) define a new credit limit and a new callback interval for the smartcard.

Please refer to the document 'SAS ITM DNASP-3 User Guide' for a complete description of this marketing facility.

COMMAND 120: ENABLE CALLBACK RULES			
Field	byte	format	Description
command_ID	4	0120	command_ID = 120

List of error codes sent by the CAS (NACK messages).

Field	Error codes	Error codes extension
command_ID	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
UA	UA_NOT_FOUND	NO_EXTENDED_ERROR_CODE
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE
MOP_PPID	CB_PROFILE_NOT_FOUND	NO_EXTENDED_ERROR_CODE

4.8.8 Command 121: Disable Callback Rules

	privilege
sop	
mop	x

The SMS uses this command to deactivate, in the Call-Collector, the processing of the callback rules for a given smartcard and to use a new credit limit as well as a new callback interval for the smartcard.

During the next callback from the STB, the CALL-COLLECTOR will no longer check the purchase history of the corresponding end-user against the various callback rules defined into the CAS database. Instead, it will use the new credit limit to renew the credit on the smartcard and will compute the next regular callback date using the new callback interval specified in the command.

Please refer to the document 'SAS ITM DNASP-3 User Guide' for a complete description of this marketing facility.

COMMAND 121: DISABLE CALLBACK RULES			
Field	byte	format	Description
command_ID	4	0121	command_ID = 121
credit_limit	7	0000000 to 6553599	New credit limit within the range 0.00 to 65535.99
call_freq	2	01 02 03 04 05 1m	annual semi-annual quarterly monthly semimonthly every m days (1 should be considered as a flag)

List of error codes sent by the CAS (NACK messages).

Field	Error codes	Error codes extension
command_ID	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
credit_limit	BAD_COMMAND_SYNTAX	BAD_CREDIT_FORMAT
call_freq	BAD_COMMAND_SYNTAX	BAD_FREQUENCY_FORMAT
UA	UA_NOT_FOUND	NO_EXTENDED_ERROR_CODE
UA	CANCELLED_CARD	NO_EXTENDED_ERROR_CODE
MOP_PPID	ACCOUNT_NOT_FOUND	NO_EXTENDED_ERROR_CODE
MOP_PPID	CB_PROFILE_NOT_FOUND	NO_EXTENDED_ERROR_CODE

FEEDBACK commands (2nn)

The commands described in this section are commands sent by the Nagravision CAS system to the SMS. The “FEEDBACK” commands are used to provide to the SMS information and data related to the IPPV functionality available to a customer. For instance, the “FEEDBACK” command will report to the SMS the list of events (movies) that have been impulsively purchased by the customer. In this context, an impulsive purchase means that the customer purchased a movie with its remote control.

Depending on the CAS configuration up to six kind of FEEDBACK commands will be generated and sent to the SMS:

```
Command 211: Start of report
Command 201: Current credit and debit
for (I=0; I<nb_new_PPV; I++)
{
    Command 202:PPV purchase list
}
Command 205: Calling phone discrepancies (if ANI is enabled)
Command 206:STU Responding Status (responding = Y)
Command 212: End of report
```

Important

The different types of command 2nn that are in the report generated by the CAS can be selected. As a result, the typical recommended report structure based on the Nagravision experience should be

```
Command 211: Start of report
for (I=0; I<nb_new_PPV; I++)
{
    Command 202:PPV purchase list
}
Command 212: End of report
```

In case a STB does not callback within the designated time frame defined in the Call-Collector database, the following FEEDBACK command will be generated:

```
Command 206: STU Responding Status (responding = N)
```

Important

In the scenario described above, the CAS will only issue the command 206. The command 211 and 212 will not be sent.

4.8.9 Command 200: Low credit alarm

The CAS uses this command to report to the SMS that the origin of the last callback (generated by the smartcard) was due to a low credit condition.

The low credit condition is reached when the credit and debit in the smartcard is smaller than the threshold credit.

COMMAND 200: LOW CREDIT ALARM			
Field	byte	format	Description
command_id	4	r_num	command_id = 0200
STU_number	14	num	Nagravision STB serial number
credit	7	r_num	Credit amount representing 00000.00 to 65535.99
debit	7	r_num	Debit amount representing 00000.00 to 65535.99

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
STU_number	BAD_COMMAND_SYNTAX	BAD_STU_NUMBER_FORMAT
credit	BAD_COMAMND_SYNTAX	BAD_CREDIT_FORMAT
debit	BAD_COMMAND_SYNTAX	BAD_DEBIT_FORMAT

4.8.10 Command 201: Current Debit and Credit

The CAS uses this command to report to the SMS the credit and debit (of the smartcard) reported by the smartcard during the last callback.

COMMAND 201: CURRENT DEBIT AND CREDIT			
Field	byte	format	Description
command_id	4	r_num	command_id = 0201
STU_number	14	num	STB CA serial number ex: 72664281 will be formatted as "00000072664281"
credit	7	r_num	Credit amount representing 00000.00 to 65535.99
debit	7	r_num	Debit amount representing 00000.00 to 65535.99

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
STU_number	BAD_COMMAND_SYNTAX	BAD_STU_NUMBER_FORMAT
credit	BAD_COMAMND_SYNTAX	BAD_CREDIT_FORMAT
debit	BAD_COMMAND_SYNTAX	BAD_DEBIT_FORMAT

4.8.11 Command 202: PPV Purchase List

The CAS uses this command to report to the SMS each new IPPV item that has been communicated by the smartcard during the last callback. Consequently, this command is sent as many times as there are new IPPV items provided during the last callback.

COMMAND 202: PPV PURCHASE LIST			
Field	byte	format	Description
command_id	4	r_num	command_id = 0202
STU_number	14	num	STB CA serial number ex: 72664281 will be formatted as "00000072664281"
IMS_product_id	12	num	IMS_product_id of the event product purchased impulsively through the EPG
purchase_date	8	YYYYMMDD	Date of IPPV purchase (UTC)
watched_status	1	r_text	Indicates if the IPPV has been watched (Y) or (N). The IPPV is watched if the accumulated viewing time exceeds the limit defined for the event or the limit predefined for the service carrying the event.

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
STU_number	BAD_COMMAND_SYNTAX	BAD_STU_NUMBER_FORMAT
IMS_product_id	BAD_COMMAND_SYNTAX	BAD_IMS_PRODUCT_ID_FORMAT
purchase_date	BAD_COMMAND_SYNTAX	BAD_DATE_FORMAT
watched_status	BAD_COMMAND_SYNTAX	BAD_FLAG_FORMAT

4.8.12 Command 203: PPV Purchase List Report

The CAS uses this command to report to the SMS the full list of new IPPV items that have been communicated by the smartcard during the last callback. Consequently, this command is sent only once with the complete list of new IPPV items provided during the last callback.

COMMAND 203: PPV PURCHASE LIST REPORT			
Field	byte	format	Description
command_id	4	r_num	command_ID = 203
STU_number	14	num	STB CA serial number ex: 72664281 will be formatted as "00000072664281"
nb_of_ppv	2	num	Number of PPV provided in this command
for(i=0;i<nb_of_ppv;i++) {			Counter
IMS_product_id	12	num	IMS_product_ID of the event product purchased impulsively through the EPG
purchase_date	8	YYYYMMDD	Date of IPPV purchase (UTC)
watched_status	1	r_text	Indicates if the IPPV has been watched (Y) or (N). The IPPV is watched if the accumulated viewing time exceeds the limit defined for the event or the limit predefined for the service carrying the event.
}			

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
STU_number	BAD_COMMAND_SYNTAX	BAD_STU_NUMBER_FORMAT
IMS_product_id	BAD_COMAMND_SYNTAX	BAD_IMS_PRODUCT_ID_FORMAT
purchase_date	BAD_COMMAND_SYNTAX	BAD_DATE_FORMAT
watched_status	BAD_COMMAND_SYNTAX	BAD_FLAG_FORMAT

4.8.13 Command 204: PPV Purchase List With Payment Status

The CAS uses this command to report to the SMS the full list of new IPPV items that have been communicated by the smartcard during the last callback. In addition, the CAS provides to the SMS the information about payment already operated between the CAS and a bank for each IPPV item.

COMMAND 204: PPV PURCHASE LIST WITH PAYMENT STATUS			
Field	byte	format	Description
command_id	4	r_num	command_id = 204
STU_number	14	num	STB CA serial number ex: 72664281 will be formatted as "00000072664281"
nb_of_ppv	2	num	Number of PPV provided in this command
for(i=0;i<nb_of_ppv;i++) {			Counter
IMS_product_id	12	num	IMS_product_ID of the event product purchased impulsively through the EPG
purchase_date	8	YYYYMMDD	Date of IPPV purchase (UTC)
purchase_time	6	HHMMSS	Timestamp of IPPV purchase (UTC)
watched_status	1	r_text	Indicates if the IPPV has been watched (Y) or not (N). The IPPV is watched if the accumulated viewing time exceeds the limit defined for the event or the limit predefined for the service carrying the event.
payment_status	1	r_text	Indicates if the IPPV has been already paid (Y) or not (N). The IPPV is possibly paid if: <ul style="list-style-type: none"> 1. The subscriber's profile authorizes automatic payments and the credit card purchase was successful, 2. The purchase was made impulsively using a prepaid credit.
}			

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
STU_number	BAD_COMMAND_SYNTAX	BAD_STU_NUMBER_FORMAT
IMS_product_id	BAD_COMAMND_SYNTAX	BAD_IMS_PRODUCT_ID_FORMAT
purchase_date	BAD_COMMAND_SYNTAX	BAD_DATE_FORMAT
purchase_time	BAD_COMMAND_SYNTAX	BAD_TIME_FORMAT
watched_status	BAD_COMMAND_SYNTAX	BAD_FLAG_FORMAT
payment_status	BAD_COMMAND_SYNTAX	BAD_FLAG_FORMAT

4.8.14 Command 205: Phone Discrepancies

The CAS uses this command to report to the SMS that during the last callback (generated by the smartcard) there was a phone discrepancy.

A phone discrepancy warning will occur if the customer's phone number (caller_ID) does not match any authorized phone numbers, stored in the CAS database, for that given customer.

The number can be a phone number, an IP address or a MAC address. Please refer to command 49, 54 or 55 definition.

COMMAND 205: PHONE DISCREPANCIES			
Field	byte	format	Description
command_id	4	r_num	command_id = 0205
STU_number	14	num	STB CA serial number Ex: 72664281 will be formatted as "00000072664281"
phone_number_1	16	num	First phone number stored in the CC
phone_number_2	16	num	Second phone number stored in the CC
phone_number_3	16	num	Third phone number stored in the CC
abnormal_phone	16	num	Number used by STB to call CC

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
STU_number	BAD_COMMAND_SYNTAX	BAD_STU_NUMBER_FORMAT
Phone_number_1	BAD_COMAMND_SYNTAX	BAD_PHONE_NUMBER_FORMAT
Phone_number_2	BAD_COMMAND_SYNTAX	BAD_PHONE_NUMBER_FORMAT
Phone_number_3	BAD_COMMAND_SYNTAX	BAD_PHONE_NUMBER_FORMAT
abnormal_phone	BAD_COMMAND_SYNTAX	BAD_PHONE_NUMBER_FORMAT

4.8.15 Command 206: STU Responding Status

The CAS uses this command to report to the SMS that a given end-user smartcard did not make a callback as expected. Only *automatic* and *immediate* callback type are concerned.

A non-responding STB is defined by the Call-Collector's callback expiration parameter.

COMMAND 206: STU RESPONDING STATUS			
Field	byte	format	Description
command_id	4	r_num	command_id = 0206
STU_number	14	num	STB CA serial number Ex: 72664281 will be formatted as "00000072664281"
responding	1	r_text	A STB may be responding: Y or N

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
STU_number	BAD_COMMAND_SYNTAX	BAD_STU_NUMBER_FORMAT
responding	BAD_COMAMND_SYNTAX	BAD_FLAG_FORMAT

4.8.16 Command 207: ICC Memory Full Alarm

The CAS uses this command to report to the SMS that the origin of the last callback (generated by the smartcard) was due to a smartcard memory full condition.

This information is retrieved by the CallCollector during the last callback. The obsolete data (i.e. expired subscriptions or IPPV) in the smartcard will be deleted. Consequently, the smartcard memory is released for subsequent usage.

Important

If the smartcard is loaded with IPPV items that are entitlements for movies broadcast in the future, then those items won't be deleted.

COMMAND 207: ICC MEMORY FULL ALARM			
Field	byte	format	Description
command_id	4	r_num	command_id = 0207
STU_number	14	num	STB CA serial number Ex: 72664281 will be formatted as "00000072664281"

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
STU_number	BAD_COMMAND_SYNTAX	BAD_STU_NUMBER_FORMAT

4.8.17 Command 211: Start of Report

The CAS uses this command to prevent the SMS that a set of *FEEDBACK* messages (i.e. 2nn commands) for a given end-user will follow. The set of messages are bounded or terminated with command 212.

COMMAND 211: START OF REPORT			
Field	byte	format	Description
command_id	4	r_num	command_id = 0211
STU_callback_date	8	YYYYMMDD	Date (UTC) of the callback being reported
STU_callback_time	6	HHMMSS	Time (UTC) of the callback being reported

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
STU_callback_date	BAD_COMMAND_SYNTAX	BAD_DATE_FORMAT
STU_callback_time	BAD_COMMAND_SYNTAX	BAD_TIME_FORMAT

4.8.18 Command 212: End of Report

The CAS uses this command to prevent the SMS that a set of *FEEDBACK* messages (i.e. 2nn commands) for a given end-user is terminated.

COMMAND 212: END OF REPORT			
Field	byte	Format	Description
command_id	4	r_num	command_id = 0212
number_of_IPPV	2	num	The number of IPPV reports (occurrences of command 202) that should have been sent in this report.

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
number_of_IPPV	BAD_COMMAND_SYNTAX	BAD_NUMBER_OF_IPPV_FORMAT

4.8.19 Command 215: Products List

The CAS uses this command to provide to the SMS the list of *subscription type* products known in the CAS database for a given end-user.

The SMS sends first, in the EMM/CTRL channel, a *command 71: Get Products*. Then, the CAS answers with *command 215: Product List* in the FEEDBACK channel.

COMMAND 215: PRODUCTS LIST			
Field	byte	Format	Description
command_id	4	r_num	command_id = 0215
original_transaction_number	9	num	Transaction number of the command 71 that triggered the generation of this command
STU_number	14	r_num	STB CA serial number ex: 72664281 will be formatted as "00000072664281"
ICC_suspended	1	r_text	Smartcard suspend state: Y or N
nb_of_products	2	num	Number of products returned in this command
for(i=0;i<nb_of_products; i++)			
{			
IMS_product_id	12	num	IMS product ID Range: 000000000000 to 999999999999
product_suspended	1	r_text	Product suspend state: Y or N
}			

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
original_transaction_number	BAD_COMMAND_SYNTAX	BAD_TRANSACTION_NUMBER_FORMAT
STU_number	BAD_COMMAND_SYNTAX	BAD_STU_NUMBER_FORMAT
ICC_suspended	BAD_COMMAND_SYNTAX	BAD_FLAG_FORMAT
nb_of_products	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
IMS_product_id	BAD_COMMAND_SYNTAX	BAD_IMS_PRODUCT_ID_FORMAT
product_suspended	BAD_COMMAND_SYNTAX	BAD_FLAG_FORMAT

4.8.20 Command 216: PPV Purchase List Extended

The CAS uses this command to report to the SMS each new IPPV item that has been communicated by the smartcard during the last callback.

This command is sent as many times as there are new IPPV items provided during the last callback.

Important

This command is similar to *command 202: PPV Purchase List*. It provides an additional parameter: the *purchase_time*.

COMMAND 216: PPV PURCHASE LIST EXTENDED			
Field	byte	Format	Description
command_id	4	r_num	command_id = 0216
STU_number	14	num	STB CA serial number Ex: 72664281 will be formatted as "00000072664281"
IMS_product_id	12	num	IMS_product_id of the event product purchased impulsively through the EPG
purchase_date	8	YYYYMMDD	Date of IPPV purchase (UTC)
purchase_time	6	HHMMSS	Timestamp of IPPV purchase (UTC)
watched_status	1	r_text	Indicates if the IPPV has been watched (Y) or (N). The IPPV is watched if the accumulated viewing time exceeds the limit defined for the event or the limit predefined for the service carrying the event.

List of error codes (in NACK messages) applicable for this section of the message.

Field	Error codes	Error codes extension
command_id	BAD_COMMAND_SYNTAX	BAD_COMMAND_ID
command_id	BAD_COMMAND_SYNTAX	BAD_NUMBER_FORMAT
STU_number	BAD_COMMAND_SYNTAX	BAD_STU_NUMBER_FORMAT
IMS_product_id	BAD_COMAMND_SYNTAX	BAD_IMS_PRODUCT_ID_FORMAT
purchase_date	BAD_COMMAND_SYNTAX	BAD_DATE_FORMAT
purchase_time	BAD_COMMAND_SYNTAX	BAD_TIME_FORMAT
watched_status	BAD_COMMAND_SYNTAX	BAD_FLAG_FORMAT

4.9 OPERATION commands (10nn)

4.9.1 Command 1000: Acknowledge Command

The CAS and the SMS use this command to acknowledge a command issued by the other side of the interface. The field “transaction_number” identifies the corresponding command that has been processed successfully.

Important

The fields IMS_product_id and SMS_product_id are no longer applicable since the series of cmd 3nn is not supported by the interface. Consequently, these fields will be set to 000000000000 and 000000000000 respectively.

COMMAND 1000: ACKNOWLEDGE COMMAND			
Field	byte	format	Description
command_id	4	r_num	command_id = 1000
transaction_number	9	num	Transaction number acknowledged Range: 000000000 to 999999999
IMS_product_id	12	num	IMS product ID is always 000000000000
SMS_product_id	12	num	SMS product ID is always 000000000000

4.9.2 Command 1001: Non-acknowledged Command

The command specified could not be completed because it is either REJECTED or POSTPONED. A rejected command is a consequence of a mistake in the command format or in the command processing. A postponed command is a consequence of a CAS system that is busy.

Important

When a command is POSTPONED, the SMS should resubmit the command after a delay of 60 min. Note that the resubmitted command should be identified with a new transaction id.

COMMAND 1001: NON-ACKNOWLEDGE COMMAND			
Field	byte	format	Description
command_id	4	r_num	Command ID = 1001
transaction_number	9	num	transaction_number acknowledged Range: 000000000 to 999999999
nack_status	1	r_num	1 = REJECTED means that the command has been rejected because an error has been detected. 2 = POSTPONED means that the command could not be completed because the system is busy.
error_code	4	r_num	Main error code, refers to chapter 5.1
error_code_ext	4	r_num	Extension error code, refers to chapter 5.2
length_of_command_body	3	num	Length of the following section
command_section	n		Command section of the command that caused the error. n = length_of_command_body

4.9.3 Command 1002: No Command

The SMS should send this command in the two following conditions:

1. While the SMS opens a channel.
2. While there is no activity, i.e. no command to send or receive. The period should be 5 min.

The SMS should separately manage the EMM/CONTROL channel and the FEEDBACK channel:

COMMAND 1002: NO COMMAND			
Field	byte	format	Description
command_id	4	r_num	command_id = 1002

5. Error codes

If an error occurs when executing a SMS command, it is reported by the use of an error code and an error code extension.” The error code indicates an error category and the error code extension gives more details regarding the error source (e.g., an incorrectly specified field or value exceeded).

An error code is always followed by an error code extension (possibly extension 0000: NO_EXTENDED_ERROR_CODE). The error codes and extensions available are indicated in the following tables.

5.1 Table of Error codes

ERROR CODE NAME	CODE	DESCRIPTION
FATAL_ERROR	0000	The Oracle database interface returns an exception as database error or protocol error due to a non-understood field in the SMS command. All SMS EMM commands might generate such FATAL_ERROR.
BAD_ROOT_HEADER_SYNTAX	0001	The syntax of the command root header is not correct. See error code extension
BAD_HEADER_SYNTAX	0002	The syntax of the command header is not correct. See error code extension
BAD_COMMAND_SYNTAX	0003	The syntax of the command is not correct. See error code extension.
DATABASE_ERROR	0004	An error occurred during an IMS database query processing.
MESSAGE_NOT_FOUND	0005	The message referenced in the command does not exist.
PRODUCT_NOT_FOUND	0006	The product_id used in the command does not exist in the IMS database.
CANCELED_CARD	0007	The smartcard referenced in the command has been canceled.
UA_NOT_FOUND	0008	The smartcard UA referenced in the command does not exist
PPV_IN_THE_PAST	0009	The command attempts to access a PPV whose validity is expired.
STU_ALREADY_EXISTS	0010	The command attempts to create a smartcard in the CC database, but the smartcard is bound to an already existing STB.
SERVICE_NOT_FOUND	0011	The service referenced in the command does not exist in the IMS database.
PRODUCT_ALREADY_EXISTS	0013	The command attempts to create an already existing product.
UA_ALREADY_EXISTS	0014	The command attempts to create in the CC database a smartcard that already exists.

ERROR CODE NAME	CODE	DESCRIPTION
BAD_EPG_FORMAT	0015	The format of the EPG data feed is not correct.
DB_INCONSISTENT_TOO_MANY_ROWS	0021	Inconsistency in the IMS database
DB_INCONSISTENT_INVALID_PRODUCT	0022	Inconsistency in the IMS database
PRODUCT_INCONSISTENT	0024	There is an inconsistency between the product definition received and the internal SMS Gateway product database.
TOO_MANY_ITEMS	0025	Too many items are given in the list.
VALUE_OUT_OF_RANGE	0026	The value in the incoming command is out of the authorized range.
BAD_USAGE	0027	The usage of this parameter is not correct. See error code extension.
INVALID_PPID	0028	The MOP PPID used in this command is invalid.
SYSTEM_ERROR	0029	An error not related to the business or to the field value occurs within the system. See error code extension.
BAD_PRODUCT_TYPE	0030	The type of the product is wrong (i.e. PPV instead of SUBSCRIPTION).
BAD_PRODUCT_STATUS	0031	The status of the product does not allow the execution of this command.
ACCOUNT_NOT_FOUND	0032	The account that corresponds to this UA and to this MOP does not exist in the database.
CB_PROFILE_NOT_FOUND	0033	The callback profile that corresponds to this UA and to this MOP does not exist in the database.
ZIP_NOT_FOUND	0034	The Zip code does not exist in the database.
RIGHT_NOT_FOUND	0035	The Right does not exist in the database

Table 5-1 Error codes

5.2 Table of Error code extensions

ERROR CODE EXTENSION NAME	CODE	DESCRIPTION
NO_EXTENDED_ERROR_CODE	0000	No error code extension is available for the error code specified.
BAD_DEBIT_FORMAT	0001	The command contains a debit field whose format is incorrect.
BAD_CREDIT_FORMAT	0002	The command contains a credit field whose format is incorrect.
BAD_CREDIT_MODE	0003	The command contained a credit_mode field whose value is not one of the authorized ones.
BAD_DATE_FORMAT	0004	The command contains a date whose format is incorrect.
BAD_DATE_SEQUENCE	0005	The command contains a begin date and an end date that are out of sequence.
BAD_FREQUENCY_FORMAT	0006	The field call_freq of the command contains a value whose format is incorrect.
BAD_STU_NUMBER_FORMAT	0007	The format of the STB number specified in the command is incorrect.
BAD_IMS_PRODUCT_ID_FORMAT	0008	The format of the IMS_product_id of the command is incorrect.
BAD_MESSAGE_NUMBER_FORMAT	0010	The value of the message_number field of the command is incorrect.
BAD_PHONE_NUMBER_FORMAT	0011	The format of the value of a phone_number field of the command is incorrect.
BAD_PRICE_FORMAT	0013	The value of the price field of the command is incorrect.
BAD_THRESHOLD_CREDIT_FORMAT	0014	The value of the threshold_credit field of the command is incorrect.
BAD_UA_FORMAT	0015	The value of the UA field of the command is incorrect.
BAD_ZIP_CODE_FORMAT	0016	The value of the zip code field of the command is incorrect.
DIFFERENT_PRODUCTS	0017	The command attempts to define a product with a product_id already attributed to a different product.
BAD_BROADCAST_MODE	0019	The broadcast mode is incorrect.
BAD_ADDRESS_TYPE	0020	The format of the value of the address_type field of the command is incorrect.
BAD_MOP_PPID	0021	The MOP_PPID indicated in the command is not valid.
BAD_DEST_ID	0022	The dest_id indicated in the command is not valid.
BAD_SOURCE_ID	0023	The source_id indicated in the command is not valid.

SMS Gateway Interface Specification

COMPANY CONFIDENTIAL

ERROR CODE EXTENSION NAME	CODE	DESCRIPTION
BAD_COMMAND_TYPE	0024	This type of SMS command does not exist.
BAD_COMMAND_ID	0025	The command ID given in the SMS command does not exist.
BAD_NUMBER_FORMAT	0027	A non-numerical character was found in the content of a Num field.
BAD_ERROR_CODE	0032	The error code specified in the command does not exist (section 8).
BAD_ERROR_CODE_EXT	0033	The error code extension specified in the command does not exist (section 8).
CREDIT_THRESHOLD_TOO_HIGH	0034	The value of the threshold_credit field of the command is too high.
BAD_SERVICE_UID_FORMAT	0040	The format of the value of a service_UID field of the command is incorrect.
BAD_SERVICE_NUMBER_FORMAT	0041	The format of the value of the service_number field of the command is incorrect.
BAD_NUMBER_OF_IPPV_FORMAT	0044	The format of the number_of_IPPV field in the command is incorrect.
BAD_IP_ADDRESS_FORMAT	0045	The format of the IP address in the command is incorrect.
EXTERNAL_SYSTEM_NOT_RESPONDING	0048	The other components of the CAS system do not respond to the gateway process.
EXTERNAL_SYSTEM_ERROR	0049	The other components of the CAS system have not successfully processed the command.
BAD_SERVICE_ID_FORMAT	0052	The format of the service ID is incorrect.
BAD_TRANSPORT_ID_FORMAT	0053	The format of the transport ID is incorrect.
BAD_NETWORK_ID_FORMAT	0054	The format of the network ID is incorrect.
BAD_LID_FORMAT	0055	The format of the lid is incorrect.
BAD_PRIORITY_FORMAT	0056	The format of the priority is incorrect.
BAD_MODE_FORMAT	0057	The format of the mode is incorrect.
LENGTH_TOO_LONG	0058	The length is out of range.
BAD_FLAG_VALUE	0059	The flag value is not recognized.
BAD_CC_PORT_FORMAT	0060	The format of the CC port is incorrect.
BAD_TRANSACTION_NUMBER_FORMAT	0061	The format of the transaction number is incorrect.
BAD_PURGE_MODE_FORMAT	0062	The format of the purge mode is incorrect.
BAD_CALLBACK_FORMAT	0063	The format of the callback type is incorrect
BAD_TIME_FORMAT	0064	The time format is incorrect
DATE_NOT_IN_THE_PAST	0065	This date must be set in the past
ACCESS_ERROR	0066	Error when trying to access database
TRANSACTION_ERROR	0067	An error occurs during a database transaction (i.e. limitation reached, overflow...)

CUSTOMER NAME

SMS Gateway Interface Specification

COMPANY CONFIDENTIAL

ERROR CODE EXTENSION NAME	CODE	DESCRIPTION
DATA_ERROR	0068	An error related to the data contained in database occurs (i.e. expected record is missing).
TRANS_NR_ALREADY_IN_USE	0069	This transaction number is already in use within the system.
COMMUNICATION_ERROR	0070	Communication error between components within the system.
INTERNAL_ERROR	0071	System internal error.
SOURCE_NOT_AUTHORIZED	0072	Use of this source identifier is not allowed.
SOURCE_ALREADY_IN_USE	0073	This source identifier is already used by another source.
DEST_NOT_AUTHORIZED	0074	Use of this destination identifier is not allowed.
MOP_NOT_AUTHORIZED	0075	Use of this MOP PPID is not allowed.
DATE_IN_THE_FUTURE	0076	This date must be set in the present or in the past.
CANCELLED_PRODUCT	0077	The product has been cancelled.
SUSPENDED_PRODUCT	0078	The product has been suspended.
INVALID_PURCHASE_DATE	0079	The purchase date is invalid.
DRAFT_PRODUCT	0080	The product is a draft.
PPV_PRODUCT	0081	The product is a PPV.
DATE_IN_THE_PAST	0082	This date must be set in the present or in the future.
ADDRESS_TYPE_NOT_AUTHORIZED	0083	This type of address is not allowed for this command.
ISD_MOP_NOT_FOUND	0084	The MOP record does not exist for this UA.
BAD_DATA_FORMAT	0085	The format of the data is invalid.
REGULAR_PRODUCT	0086	This product is a regular (subscription) product.
INVALID_CATEGORY	0087	The category requested is not defined in the CAS.

Table 5-2 Error code extensions

CUSTOMER NAME

5.3 Error codes and extensions for Command Root Header section

Among the error codes and extensions defined above, those that are applicable to report errors in the root header fields are:

ERROR CODE	ERROR CODE EXTENSION
BAD_ROOT_HEADER_SYNTAX	BAD_COMMAND_TYPE BAD_SOURCE_ID BAD_DEST_ID BAD_MOP_PPID BAD_DATE_FORMAT

The error code and error code extension are used in the following way:

FIELD	ERROR CODE	ERROR CODE EXTENSION
transaction_number	BAD_ROOT_HEADER_SYNTAX	NO_EXTENDED_ERROR
command_type	BAD_ROOT_HEADER_SYNTAX	BAD_COMMAND_TYPE
source_id	BAD_ROOT_HEADER_SYNTAX	BAD_SOURCE_ID
dest_id	BAD_ROOT_HEADER_SYNTAX	BAD_DEST_ID
MOP_PPID	BAD_ROOT_HEADER_SYNTAX	BAD_MOP_PPID
creation_date	BAD_ROOT_HEADER_SYNTAX	BAD_DATE_FORMAT

5.4 Error codes and extensions for Command Header section

ERROR CODE	ERROR CODE EXTENSION
BAD_HEADER_SYNTAX	BAD_BROADCAST_MODE BAD_ADDRESS_TYPE BAD_DATE_SEQUENCE BAD_DATE_FORMAT
UA_NOT_FOUND	NO_EXTENDED_ERROR_CODE
CANCELED_CARD	NO_EXTENDED_ERROR_CODE

The error codes and error code extensions are used in the following way to report errors in command header fields.

FIELD	ERROR CODE	ERROR CODE EXTENSION
broadcast_mode	BAD_HEADER_SYNTAX	BAD_BROADCAST_MODE
broadcast_start_date	BAD_HEADER_SYNTAX	BAD_DATE_FORMAT BAD_DATE_SEQUENCE
broadcast_end_date	BAD_HEADER_SYNTAX	BAD_DATE_FORMAT BAD_DATE_SEQUENCE
address_type	BAD_HEADER_SYNTAX	BAD_ADDRESS_TYPE
UA	UA_NOT_FOUND CANCELED_CARD BAD_HEADER_SYNTAX	NO_EXTENDED_ERROR_CODE NO_EXTENDED_ERROR_CODE BAD_UA_FORMAT

6. UA and CA-S/N checksum

This chapter describes the procedure to manage UA (unique address) and CA-S/N (conditional access serial number) at the customer site.

6.1 Definitions

The mandatory information to allow a CAS to fully recognize a given customer is:

UA Unique address of the smartcard. This is a 12-digits long number that uniquely identifies the smartcard. The first 10 digits represent the address itself; the last 2 digits are a checksum allowing the SMS to verify the number given by the customer. The preferred (but not mandatory) format is:

nn nnnn nnnn cc

This number is printed and bar-coded on the smartcard and is normally accessible through a set-top box menu on the TV screen.

CA-S/N Conditional access serial number. This is a 12-digits long number that uniquely identifies the set-top box for the conditional access system. The first 10 digits represent the serial number; the last 2 digits are a checksum allowing the SMS to verify the number given by the customer. The preferred (but not mandatory) format is:

nn nnnn nnnn cc

This number may be printed (and possibly bar-coded) on a sticker on the set-top box and is normally accessible through a set-top box menu on the TV screen.

This number may differ (and usually differs) from the manufacturing S/N, which identifies the set-top box for the manufacturer himself. We had to use a common format throughout the system and different numbers for all set-top boxes whatever the format chosen by the manufacturer.

6.2 Pairing operation

The pairing operation links a smartcard with a set-top box. The pairing operation is mandatory. Without the pairing operation, the end-user's STB does not work.

The end-user will communicate his UA and his CA-S/N to the SMS during the installation process. Those numbers may be filled in the contract or in any place the customer finds suitable for this purpose.

The pairing key, which is used to effectively prevent the use of a smartcard in another set-top box, is provided in the Nagravision system.

6.3 Data files

Nagravision usually provides the customer with files containing the list of the smartcards produced. Those files contain a list of UA. They are given by production box (250 smartcards) and/or by production batch (indeterminate number of boxes in one file).

On the other hand, the set-top box manufacturer has the responsibility to provide files containing the list of set-top boxes effectively produced. Those files must contain a list or range of CAS/N. If needed by the customer, those files may provide a link between the CAS/N and the manufacturing S/N for each box, but this is not requested by the Nagravision system.

Nagravision may possibly provide the range of CAS/N allocated for each manufacturer but has no view on what is effectively produced.

6.4 CA-S/N – Pairing keys

The CA-S/N is chosen by Nagravision and provided to the set-top box manufacturer, each of them associated with a pairing key. This pairing key is an element of security of the system and shall not be known by the end-user, by the customer, or by any unauthorized people at the manufacturing site. Thus this pairing key shall not appear on any support (label, screen, paper, accessible memory, file...). A violation of this rule is a severe security breach and may induce Nagravision into litigation actions.

6.5 Checksum algorithm

The following C routine describes the formula used for the computation of the checksum:

```
unsigned char calcChecksum (unsigned long SN)
{
    return ((6*(SN/100000000L)+19*(SN/10000000L%10)+
            8*(SN/10000L%1000)+(SN/100L%100))%23+
            (SN%100))%100;
}
```

7. ASCII Table

Decimal	Hex	Binary	Value
00	00	00000000	NUL (Null char.)
32	20	00100000	SP (Space)
33	21	00100001	! (exclamation mark)
34	22	00100010	" (double quote)
35	23	00100011	# (number sign)
36	24	00100100	\$ (dollar sign)
37	25	00100101	% (percent)
38	26	00100110	& (ampersand)
39	27	00100111	' (single quote)
40	28	00101000	((left/opening parenthesis)
41	29	00101001) (right/closing parenthesis)
42	2A	00101010	* (asterisk)
43	2B	00101011	+ (plus)
44	2C	00101100	, (comma)
45	2D	00101101	- (minus or dash)
46	2E	00101110	. (dot)
47	2F	00101111	/ (forward slash)
48	30	00110000	0
49	31	00110001	1
50	32	00110010	2
51	33	00110011	3
52	34	00110100	4
53	35	00110101	5
54	36	00110110	6
55	37	00110111	7
56	38	00111000	8
57	39	00111001	9
58	3A	00111010	: (colon)
59	3B	00111011	; (semi-colon)
60	3C	00111100	< (less than)
61	3D	00111101	= (equal sign)
62	3E	00111110	> (greater than)
63	3F	00111111	? (question mark)
64	40	01000000	@ (AT symbol)
65	41	01000001	A
66	42	01000010	B
67	43	01000011	C
68	44	01000100	D
69	45	01000101	E
70	46	01000110	F
71	47	01000111	G
72	48	01001000	H
73	49	01001001	I
74	4A	01001010	J
75	4B	01001011	K

SMS Gateway Interface Specification

COMPANY CONFIDENTIAL

76	4C	01001100	L
77	4D	01001101	M
78	4E	01001110	N
79	4F	01001111	O
80	50	01010000	P
81	51	01010001	Q
82	52	01010010	R
83	53	01010011	S
84	54	01010100	T
85	55	01010101	U
86	56	01010110	V
87	57	01010111	W
88	58	01011000	X
89	59	01011001	Y
90	5A	01011010	Z
91	5B	01011011	[(left/opening bracket)
92	5C	01011100	\ (back slash)
93	5D	01011101] (right/closing bracket)
94	5E	01011110	^ (caret/circumflex)
95	5F	01011111	_ (underscore)
96	60	01100000	'
97	61	01100001	a
98	62	01100010	b
99	63	01100011	c
100	64	01100100	d
101	65	01100101	e
102	66	01100110	f
103	67	01100111	g
104	68	01101000	h
105	69	01101001	i
106	6A	01101010	j
107	6B	01101011	k
108	6C	01101100	l
109	6D	01101101	m
110	6E	01101110	n
111	6F	01101111	o
112	70	01110000	p
113	71	01110001	q
114	72	01110010	r
115	73	01110011	s
116	74	01110100	t
117	75	01110101	u
118	76	01110110	v
119	77	01110111	w
120	78	01111000	x
121	79	01111001	y
122	7A	01111010	z

CUSTOMER NAME

8. Examples

8.1 Device_IO connection establishment

The following example is a network capture of one whole Device_IO connection process.

```

Packet #1
  Flags:      0x00
  Status:     0x01
  Packet Length: 64
  Timestamp:  18:04:12.473073 10/09/2001
Ethernet Header
  Destination: 08:00:2B:C5:7E:2A
  Source:      00:C0:F0:3D:7F:9D
  Protocol Type: 0x0800 IP
IP Header - Internet Protocol Datagram
  Version:      4
  Header Length: 5 (20 bytes)
  Type of Service: %00000000
  Precedence: Routine, Normal Delay, Normal Throughput, Normal Reliability
  Total Length: 44
  Identifier:    63087
  Fragmentation Flags: %010 Do Not Fragment Last Fragment
  Fragment Offset: 0 (0 bytes)
  Time To Live: 128
  Protocol:      6 TCP - Transmission Control Protocol
  Header Checksum: 0x82E9
  Source IP Address: 192.168.0.31
  Dest. IP Address: 192.168.0.3
  No IP Options
TCP - Transport Control Protocol
  Source Port: 1090 ff-fms
  Destination Port: 20000 sms_gateway
  Sequence Number: 55594
  Ack Number: 0
  Offset: 6 (24 bytes)
  Reserved: %000000
  Code: %000010 Synch
  Window: 8192
  Checksum: 0xCB26
  Urgent Pointer: 0
  TCP Options:
    Option Type: 2 Maximum Segment Size
    Length: 4
    MSS: 1460
  TCP Data Area: No more data.
  Extra bytes (Padding):
    .. 00 00
FCS - Frame Check Sequence
  FCS (Calculated): 0x5FAFB276

```

```

Packet #2
  Flags:      0x00
  Status:     0x01
  Packet Length: 64
  Timestamp:  18:04:12.473409 10/09/2001
Ethernet Header
  Destination: 00:C0:F0:3D:7F:9D
  Source:      08:00:2B:C5:7E:2A
  Protocol Type: 0x0800 IP
IP Header - Internet Protocol Datagram
  Version:      4
  Header Length: 5 (20 bytes)
  Type of Service: %00000000
  Precedence: Routine, Normal Delay, Normal Throughput, Normal Reliability
  Total Length: 44
  Identifier:    56682
  Fragmentation Flags: %010 Do Not Fragment Last Fragment
  Fragment Offset: 0 (0 bytes)
  Time To Live: 60
  Protocol:      6 TCP - Transmission Control Protocol
  Header Checksum: 0xDFEE
  Source IP Address: 192.168.0.3
  Dest. IP Address: 192.168.0.31
  No IP Options
TCP - Transport Control Protocol
  Source Port: 20000 sms_gateway
  Destination Port: 1090 ff-fms
  Sequence Number: 1524200406
  Ack Number: 55595
  Offset: 6 (24 bytes)
  Reserved: %000000
  Code: %010010 Ack Synch

```

SMS Gateway Interface Specification

COMPANY CONFIDENTIAL

```

Window:          33580
Checksum:        0x9939
Urgent Pointer:  0
TCP Options:
  Option Type:   2  Maximum Segment Size
  Length:       4
  MSS:          1460

```

```

TCP Data Area:    No more data.
Extra bytes (Padding):
..               00 00
FCS - Frame Check Sequence
FCS (Calculated): 0x36B97026

```

```

Packet #3
Flags:           0x00
Status:          0x01
Packet Length: 64
Timestamp:       18:04:12.473450 10/09/2001

```

Ethernet Header

```

Destination:     08:00:2B:C5:7E:2A
Source:          00:C0:F0:3D:7F:9D
Protocol Type:   0x0800  IP

```

IP Header - Internet Protocol Datagram

```

Version:         4
Header Length:   5  (20 bytes)
Type of Service: %00000000
Precedence: Routine, Normal Delay, Normal Throughput, Normal Reliability
Total Length:    40
Identifier:      63343
Fragmentation Flags: %010 Do Not Fragment Last Fragment
Fragment Offset: 0  (0 bytes)
Time To Live:    128
Protocol:        6  TCP - Transmission Control Protocol
Header Checksum: 0x81ED
Source IP Address: 192.168.0.31
Dest. IP Address: 192.168.0.3
No IP Options

```

TCP - Transport Control Protocol

```

Source Port:     1090  ff-fms
Destination Port: 20000  sms_gateway
Sequence Number: 55595
Ack Number:      1524200407
Offset:          5  (20 bytes)
Reserved:        %000000
Code:            %010000  Ack
Window:          8760
Checksum:        0x11EB
Urgent Pointer:  0
No TCP Options
TCP Data Area:   No more data.

```

```

Extra bytes (Padding):
.....          00 00 00 00 00 00
FCS - Frame Check Sequence
FCS (Calculated): 0x62217FDC

```

```

Packet #4
Flags:           0x00
Status:          0x01
Packet Length: 69
Timestamp:       18:04:12.485934 10/09/2001

```

Ethernet Header

```

Destination:     08:00:2B:C5:7E:2A
Source:          00:C0:F0:3D:7F:9D
Protocol Type:   0x0800  IP

```

IP Header - Internet Protocol Datagram

```

Version:         4
Header Length:   5  (20 bytes)
Type of Service: %00000000
Precedence: Routine, Normal Delay, Normal Throughput, Normal Reliability
Total Length:    51
Identifier:      63599
Fragmentation Flags: %010 Do Not Fragment Last Fragment
Fragment Offset: 0  (0 bytes)
Time To Live:    128
Protocol:        6  TCP - Transmission Control Protocol
Header Checksum: 0x80E2
Source IP Address: 192.168.0.31
Dest. IP Address: 192.168.0.3
No IP Options

```

TCP - Transport Control Protocol

```

Source Port:     1090  ff-fms
Destination Port: 20000  sms_gateway
Sequence Number: 55595
Ack Number:      1524200407
Offset:          5  (20 bytes)
Reserved:        %000000
Code:            %011000  Ack  Push
Window:          8760
Checksum:        0xC9C3
Urgent Pointer:  0
No TCP Options

```

```

SMSgateway spec. 2.6.2 - decoder v.0.8
DeviceIO: message_1
len:          9

```

CUSTOMER NAME

SMS Gateway Interface Specification

COMPANY CONFIDENTIAL

```

op_mode:          1  Fast data transfer (do not allow tracing)
ob_name_len:      7
ob_name:          SMS_GWY
FCS - Frame Check Sequence
FCS (Calculated): 0xC6762DDA

```

```

Packet #5
Flags:            0x00
Status:           0x01
Packet Length: 64
Timestamp:        18:04:12.486303 10/09/2001

```

Ethernet Header

```

Destination:      00:C0:F0:3D:7F:9D
Source:           08:00:2B:C5:7E:2A
Protocol Type:    0x0800  IP

```

IP Header - Internet Protocol Datagram

```

Version:          4
Header Length:    5  (20 bytes)
Type of Service:  %00000000
Precedence: Routine, Normal Delay, Normal Throughput, Normal Reliability
Total Length:     43
Identifier:       56683
Fragmentation Flags: %010 Do Not Fragment Last Fragment
Fragment Offset:   0  (0 bytes)
Time To Live:     60
Protocol:         6  TCP - Transmission Control Protocol
Header Checksum:  0xDFEE
Source IP Address: 192.168.0.3
Dest. IP Address: 192.168.0.31
No IP Options

```

TCP - Transport Control Protocol

```

Source Port:      20000  sms_gateway
Destination Port: 1090  ff-fms
Sequence Number:  1524200407
Ack Number:       55606
Offset:           5  (20 bytes)
Reserved:         %00000000
Code:             %011000 Ack Push
Window:           33580
Checksum:         0xAADF
Urgent Pointer:   0
No TCP Options

```

SMSGateway spec. 2.6.2 - decoder v.0.8

```

DeviceIO: message_2
len:        1
value:      6
Extra bytes (Padding):
...         00 00 00

```

FCS - Frame Check Sequence

```

FCS (Calculated): 0xBDB648A1

```

```

Packet #6
Flags:            0x00
Status:           0x01
Packet Length: 64
Timestamp:        18:04:12.640961 10/09/2001

```

Ethernet Header

```

Destination:      08:00:2B:C5:7E:2A
Source:           00:C0:F0:3D:7F:9D
Protocol Type:    0x0800  IP

```

IP Header - Internet Protocol Datagram

```

Version:          4
Header Length:    5  (20 bytes)
Type of Service:  %00000000
Precedence: Routine, Normal Delay, Normal Throughput, Normal Reliability
Total Length:     40
Identifier:       63855
Fragmentation Flags: %010 Do Not Fragment Last Fragment
Fragment Offset:   0  (0 bytes)
Time To Live:     128
Protocol:         6  TCP - Transmission Control Protocol
Header Checksum:  0x7FED
Source IP Address: 192.168.0.31
Dest. IP Address: 192.168.0.3
No IP Options

```

TCP - Transport Control Protocol

```

Source Port:      1090  ff-fms
Destination Port: 20000  sms_gateway
Sequence Number:  55606
Ack Number:       1524200410
Offset:           5  (20 bytes)
Reserved:         %00000000
Code:             %010000 Ack
Window:           8757
Checksum:         0x11E0
Urgent Pointer:   0
No TCP Options
TCP Data Area:    No more data.

```

```

Extra bytes (Padding):
..... 00 00 00 00 00 00

```

FCS - Frame Check Sequence

```

FCS (Calculated): 0xB161CF21

```

CUSTOMER NAME

SMS Gateway Interface Specification

COMPANY CONFIDENTIAL

```

Packet #7
  Flags:      0x00
  Status:     0x01
  Packet Length: 64
  Timestamp:  18:04:12.641152 10/09/2001
Ethernet Header
  Destination: 00:C0:F0:3D:7F:9D
  Source:      08:00:2B:C5:7E:2A
  Protocol Type: 0x0800 IP
IP Header - Internet Protocol Datagram
  Version:      4
  Header Length: 5 (20 bytes)
  Type of Service: %00000000
  Precedence: Routine, Normal Delay, Normal Throughput, Normal Reliability
  Total Length: 43
  Identifier:    56686
  Fragmentation Flags: %010 Do Not Fragment Last Fragment
  Fragment Offset: 0 (0 bytes)
  Time To Live: 60
  Protocol:      6 TCP - Transmission Control Protocol
  Header Checksum: 0xDFEB
  Source IP Address: 192.168.0.3
  Dest. IP Address: 192.168.0.31
  No IP Options
TCP - Transport Control Protocol
  Source Port: 20000 sms_gateway
  Destination Port: 1090 ff-fms
  Sequence Number: 1524200410
  Ack Number: 55606
  Offset: 5 (20 bytes)
  Reserved: %000000
  Code: %011000 Ack Push
  Window: 33580
  Checksum: 0xB0DC
  Urgent Pointer: 0
  No TCP Options
SMSgateway spec. 2.6.2 - decoder v.0.8
DeviceIO: message_3
  len: 1
  value: 0
Extra bytes (Padding):
  ... 00 00 00
FCS - Frame Check Sequence
  FCS (Calculated): 0x77CEA603

```

```

Packet #14
  Flags:      0x00
  Status:     0x01
  Packet Length: 64
  Timestamp:  18:04:12.841621 10/09/2001
Ethernet Header
  Destination: 08:00:2B:C5:7E:2A
  Source:      00:C0:F0:3D:7F:9D
  Protocol Type: 0x0800 IP
IP Header - Internet Protocol Datagram
  Version:      4
  Header Length: 5 (20 bytes)
  Type of Service: %00000000
  Precedence: Routine, Normal Delay, Normal Throughput, Normal Reliability
  Total Length: 40
  Identifier:    65135
  Fragmentation Flags: %010 Do Not Fragment Last Fragment
  Fragment Offset: 0 (0 bytes)
  Time To Live: 128
  Protocol:      6 TCP - Transmission Control Protocol
  Header Checksum: 0x7AED
  Source IP Address: 192.168.0.31
  Dest. IP Address: 192.168.0.3
  No IP Options
TCP - Transport Control Protocol
  Source Port: 1090 ff-fms
  Destination Port: 20000 sms_gateway
  Sequence Number: 55606
  Ack Number: 1524200413
  Offset: 5 (20 bytes)
  Reserved: %000000
  Code: %010000 Ack
  Window: 8754
  Checksum: 0x11E0
  Urgent Pointer: 0
  No TCP Options
  TCP Data Area: No more data.
Extra bytes (Padding):
  ..... 00 00 00 00 00 00
FCS - Frame Check Sequence
  FCS (Calculated): 0x22659BEF

```

8.2 Example of command (52)

Raw data:

```

0000: 08 00 2B C5 7E 2A 00 C0 F0 3D 7F 9D 08 00 45 00  ..+.*...8=...E.
0016: 00 78 29 70 40 00 80 06 4F 9D C0 A8 00 1F C0 A8  .x)p@...0.....
0032: 00 03 04 42 4E 20 00 00 D9 78 5A D9 74 24 50 18  ...BN...xZ.t$P.
0048: 21 EB 7B 6B 00 00 00 4E 30 30 30 30 30 30 30 30  !,{k...00000000
0064: 32 30 31 30 30 30 31 30 30 30 32 30 30 32 35 37  2010001000200257
0080: 32 30 30 31 31 30 30 39 4E 32 30 30 31 31 30 30  20011009N2001100
0096: 39 32 30 30 31 31 30 30 39 55 30 30 30 30 30 30  920011009U0000000
0112: 30 30 30 31 30 30 35 32 31 32 33 34 35 36 37 38  0001005212345678
0128: 39 30 20 20 20 20 00 00 00 00 90  ....

```

Whole network packet, interpreted:

```

Flags: 0x00
Status: 0x00
Packet Length: 138
Timestamp: 18:05:08.580530 10/09/2001

Ethernet Header
  Destination: 08:00:2B:C5:7E:2A
  Source: 00:C0:F0:3D:7F:9D
  Protocol Type: 0x0800 IP

IP Header - Internet Protocol Datagram
  Version: 4
  Header Length: 5 (20 bytes)
  Type of Service: 00000000
    000. .... Precedence: Routine,
    ...0 .... Normal Delay,
    .... 0... Normal Throughput,
    .... 0... Normal Reliability
    .... ..0. ECT bit - transport protocol will ignore the CE bit
    .... ..0 CE bit - no congestion
  Total Length: 1200x00
  Identifier: 10608
  Fragmentation Flags: 010
    0.. Reserved
    .1. Do Not Fragment
    ..0 Last Fragment
  Fragment Offset: 0 (0 bytes)
  Time To Live: 128
  Protocol: 6 TCP - Transmission Control Protocol
  Header Checksum: 0x4F9D
  Source IP Address: 192.168.0.31
  Dest. IP Address: 192.168.0.3
  No IP Options

TCP - Transport Control Protocol
  Source Port: 1030 ff-fms
  Destination Port: 20000 sms_gateway
  Sequence Number: 55672
  Ack Number: 1524200484
  Offset: 5 (20 bytes)
  Reserved: 00000000
  Code: 011000 Ack Push
  Window: 8683
  Checksum: 0x7B6B
  Urgent Pointer: 0
  No TCP Options

SMSGateway spec. 2.6.2 - decoder v.0.8
  Len: 78
  SMS command root header
    transaction_number: 000000002
    command_type: 01 ENM
    source_id: 0001
    dest_id: 0002
    MOP_PPID: 00257
    creation_date: 20011009
  ENM
    broadcast_mode: N
    broadcast_start_date: 20011009
    broadcast_end_date: 20011009
    address_type: U
    UA: 0000000001
  Pair the ICC with the STB:
    command_ID: 0052
    STU_number: 1234567890
  FCS - Frame Check Sequence
    FCS (Calculated): 0x7CA87CFC

```

— END OF DOCUMENT —