

# MorphoAccess™ Host System Interface Specification

Revision 3.0 April 2002  
Ref : 3000005996

Produced by **SAGEM SA**  
Etablissement de Saint-Christophe  
2, rue du Petit Albi  
BP 8250 Cergy Saint-Christophe  
95801 CERGY PONTOISE CEDEX,  
FRANCE  
Tel.: 33 (0)1 30 73 70 70  
Fax: 33 (0)1 30 73 16 60  
Copyright ©2001 **SAGEM SA**  
[www.sagem.com](http://www.sagem.com)

## WARNINGS

Copyrights 2001-2002, SAGEM SA, All rights reserved.

Information in this document is subject to change without notice and does not represent a commitment on the part of SAGEM SA. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying or recording, for any purpose without the express written permission of SAGEM SA.

The software described in this document is supplied under a license agreement or nondisclosure agreement. It is against the law to copy the software on any medium except as specifically allowed in the agreement.

This manual makes reference to names and products that are trademarks of their respective owners.

MORPHO® is a registered trademarks of SAGEM SA.

Printed in France.

## HISTORICAL OF REVISIONS

Revision	Date	Author	Document revision history
A	21/11/00	SAGEM SA	Creation of the document
B	19/12/00	SAGEM SA	Explain Application Protocol
C	19/02/01	SAGEM SA	Modification of Low Layer and Application Protocols.
D	02/03/01	SAGEM SA	Add Application commands for MorphoAccess
E	01/09/01	SAGEM SA	Update Version 1.0
F	01/10/01	SAGEM SA	Version 1.1 Add a new commands
G	14/12/01	SAGEM SA	Version 2.01 for MA300
H	30/04/02	SAGEM SA	Version 3.0

Insérer un saut de page après cette ligne de texte caché pour l'historique si nécessaire

# TABLE OF CONTENTS

<b>1</b>	<b>CONVENTIONS</b>	<b>1-6</b>
1.1	ACRONYMS AND ABBREVIATION	1-6
<b>2</b>	<b>SCOPE</b>	<b>2-7</b>
2.1	IDENTIFICATION	2-7
2.2	SYSTEM ARCHITECTURE	2-7
<b>3</b>	<b>PHYSICAL LAYER</b>	<b>3-8</b>
<b>4</b>	<b>LOW LAYER PROTOCOL</b>	<b>4-9</b>
4.1	RS232 PROTOCOL	4-9
4.1.1	Definition	4-9
4.1.2	Frames sequence	4-11
4.1.3	Timing Characteristics	4-11
4.1.4	Communication Error Case	4-11
4.2	RS232 SERVICE PROTOCOL	4-11
4.2.1	Request Counter management	4-11
4.2.2	Timing specification	4-12
4.2.3	Error cases	4-12
4.2.4	Typical Transactions workflow	4-12
4.2.5	Samples	4-14
4.3	TCP/IP PROTOCOL	4-16
4.3.1	Port number	4-16
4.3.2	Byte Order	4-16
4.4	RS422 PROTOCOL	4-16
<b>5</b>	<b>APPLICATION PROTOCOL</b>	<b>5-17</b>
5.1	COMPATIBILITY AND RECOMMENDATION	5-17
5.2	BASE STRUCTURE	5-18
5.3	DATA STRUCTURE	5-18
5.4	INITIALIZATION FUNCTIONS DESCRIPTIONS	5-20
5.4.1	DEFAULT_INIT ID=0x0A	5-20
5.4.2	GET_VERSION ID=0x03	5-22
5.4.3	REBOOT ID=0x04	5-23
5.4.4	PING ID=0x08	5-24
5.5	CONFIGURATION FUNCTIONS DESCRIPTIONS	5-25
5.5.1	SET_CONFIGURATION ID=0x01	5-25
5.5.2	GET_CONFIGURATION ID=0x06	5-31
5.6	SECURITY CONFIGURATION	5-32
5.6.1	SET_PUBLIC_KEY ID=0x53	5-32
5.7	BIOMETRICS FUNCTIONS DESCRIPTION	5-34
5.7.1	AUTHENTICATE ID=0x20	5-34
5.7.2	ENROLL ID=0x21	5-37
5.7.3	IDENTIFY ID=0x22	5-40
5.8	DATABASE FUNCTIONS DESCRIPTION	5-42
5.8.1	CREATE DATABASE ID=0x30	5-42
5.8.2	ERASE BASE ID=0x32	5-44
5.8.3	ERASE ALL BASE ID=0x33	5-45
5.8.4	ADD BASE RECORD ID=0x35	5-46
5.8.5	REMOVE BASE RECORD ID=0x36	5-48

**SAGEM SA**

5.8.6	GARBAGE COLLECTOR ID=0x0F .....	5-49
5.8.7	GET BASE CONFIG ID=0x07 .....	5-50
5.8.8	GET ALL BASE CONFIG ID=0x31 .....	5-52
5.8.9	BASE DOWNLOAD ID=0x37 .....	5-54
5.8.10	RESET DOWNLOAD ID=0x3A .....	5-56
5.9	LOG MANAGEMENT FUNCTIONS DESCRIPTIONS (V1.1) .....	5-57
5.9.1	GET LOG STATUS ID=0x5D .....	5-57
5.9.2	GET LOG ID=0x5B .....	5-59
5.9.3	ERASE LOG ID=0x5C .....	5-60
5.10	TIME MASK FUNCTIONS DESCRIPTION .....	5-61
5.10.1	SET TIME MASK ID=0x42 .....	5-61
5.10.2	GET TIME MASK ID=0x43 .....	5-63
5.11	MULTILINGUAL FUNCTIONS DESCRIPTION .....	5-64
5.11.1	SET USER MESSAGE ID=0x40 .....	5-64
5.11.2	ERASE USER TABLE ID=0x41 .....	5-66
5.12	PROXY FUNCTIONS DESCRIPTION .....	5-67
5.12.1	PRINT SCREEN MESSAGE ID=0x54 .....	5-67
5.12.2	READ FROM KEYBOARD ID=0x55 .....	5-69
5.12.3	ACCESS AUTHORIZATION ID=0x56 .....	5-70
5.12.4	WIEGAND DATACLOCK READ ID=0x52 .....	5-71
5.13	CONTACTLESS FUNCTIONS DESCRIPTION .....	5-72
5.13.1	CONTACTLESS READ ID=0x57 .....	5-72
5.13.2	CONTACTLESS AUTHENT ID=0x59 .....	5-74
5.13.3	CRYPTO WRITE ID=0x58 .....	5-77
5.14	FUNCTIONS TABLE .....	5-78
5.15	ERROR CODES VALUE .....	5-79
5.16	STATUS CODES VALUE .....	5-80
5.17	DATABASE DOWNLOAD ERROR CODES .....	5-81
5.18	USER MESSAGE IDS .....	5-82
5.19	CONSTANTS .....	5-85
5.20	ID VALUE .....	5-85
5.21	CONTACTLESS CODE VALUES .....	5-87
5.22	CONFIGURATION TABLE .....	5-88
5.22.1	Serial Link Configuration .....	5-88
5.22.2	Wiegand/Dataclock Configuration .....	5-89
5.22.3	Setting Up Relay Parameters .....	5-89

## List of Figures

FIGURE 1 :	MORPHOACCESS SYSTEM ARCHITECTURE .....	2-7
FIGURE 2 :	FRAMES SEQUENCE .....	4-11
FIGURE 3 :	PACKET TRANSMISSION OF LESS THAN 1024 BYTES OF EFFECTIVE DATA .....	4-12
FIGURE 4 :	PACKET TRANSMISSION MORE THAN 1024 BYTES .....	4-13
FIGURE 5 :	PACKET TRANSMISSION ERRORS .....	4-13
FIGURE 6 :	TIME MASK STRUCTURE .....	5-61

## List of Tables

TABLE 1 :	ACRONYMS AND ABBREVIATIONS .....	1-6
TABLE 2 :	MORPHOACCESS LOCAL DATABASE STRUCTURE .....	5-18

# 1 Conventions

This section presents acronyms, abbreviations, measurement units, and conventions used in this document.

## 1.1 ACRONYMS AND ABBREVIATION

Acronym/Abbreviation	Definition
MOBI	Biometric Modules
ILV	Identifier Length Value
SPILV	Service Protocol ILV
CRC	Cyclic Redundancy Check
STX	Start Text
ETX	End Text
MSB	Most Significant Bit
LSB	Least Significant Bit
PK	Characteristics points of a finger used by biometrics process

**Table 1 : Acronyms and Abbreviations**

## 2 Scope

### 2.1 IDENTIFICATION

This document defines the interface between a Terminal (MM110 or MorphoAccess) and a host system (typically a PC). The following communication layers are described:

- Physical connection,
- Low level protocol,
- Application protocol.

### 2.2 SYSTEM ARCHITECTURE

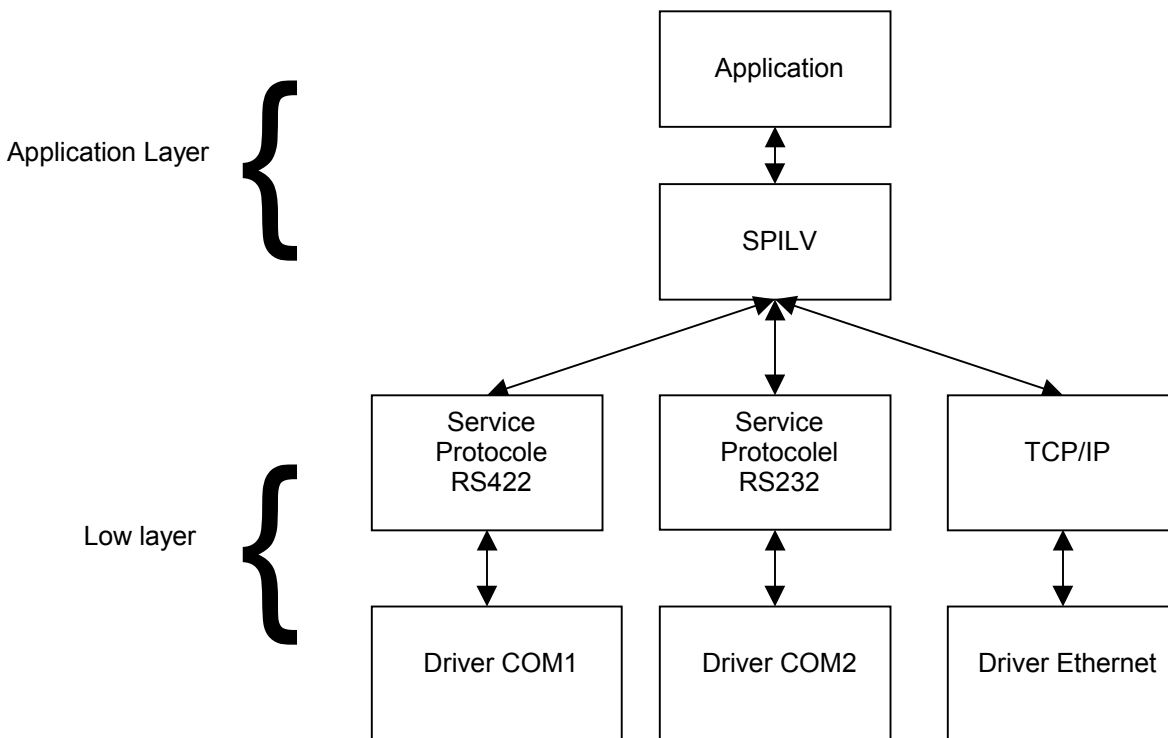


Figure 1 : MorphoAccess System Architecture

## 3 Physical layer

The physical layer between the Host and the terminal will be RS232, TCP/IP or RS422.

- ◆ RS232 Default Configuration :

- Type: 3 wires: TX, RX, GND.
- Baudrate: 9600 Bauds
- Data bits: 7 bits
- Parity: even
- Stop bits: 2 bits
- Flow Control: none

- ◆ RS422 Default Configuration :

- Type: 4 wires: TX+, TX-, RX+, RX-.
- Baudrate: 9600 Bauds
- Data bits: 7 bits
- Parity: even
- Stop bits: 2 bits
- Flow Control: none

- ◆ TCP/IP Default Configuration

- IP Address : 134.1.32.214

- ◆ Wiegand Default Configuration

- Id length : 26 bytes
- Pulse Width : 50  $\mu$ s
- Pulse Interval : 450 $\mu$ s

- ◆ Relay configuration

- enable



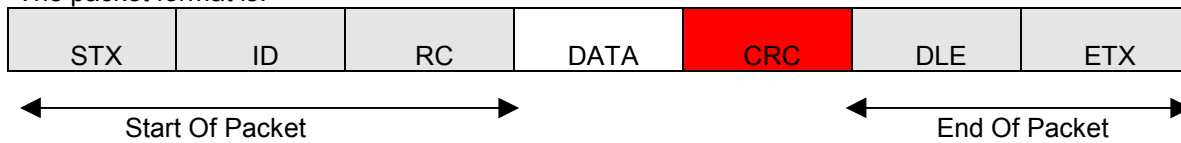
## 4 Low layer protocol

### 4.1 RS232 PROTOCOL

#### 4.1.1 Definition

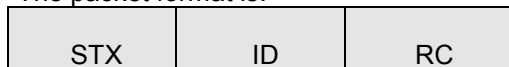
##### 4.1.1.1 DATA PACKET STRUCTURE

The packet format is:



##### 4.1.1.2 ACK PACKET AND NACK PACKET STRUCTURE

The packet format is:



##### 4.1.1.3 ABBREVIATION

Fields name	Definition	Size (Bytes)	Value
<STX> :	Start Text	1	0x02
<ID>	Packet Identifier	1	--
<RC>	Request Counter	1	--
<DATA>	Data value	Up to 1024	--
<CRC>	Transmission error control	2	--
<DLE> :	Data Link Escape	1	0x1B
<ETX>	End Text	1	0x03

The maximum size allowed for a packet is 2058 bytes.(STX+ID+DLE+ETX+(RC+DATA+CRC)\*2 [if stuffed])

##### 4.1.1.4 BYTE ORDER

The packet byte order is Little Endian: multi bytes data are sent least significant byte first (LSB).

##### 4.1.1.5 DATA

Data are formatted as I.L.V. packets.

### 4.1.1.6 STUFFING

- ◆ Software handshake capabilities (XON-XOFF) are preserved by replacing, in the <RC + Data + CRC>, all XON(0x11) / XOFF(0x13) characters by the couple <DLE> <XON+1> (0x12) or <DLE> <XOFF+1> (0x14).
- ◆ To prevent confusion with the frames sequences <STX><ID> and <DLE><ETX>, every <DLE> byte in the <RC + Data + CRC> is preceded by an extra <DLE> byte ('stuffing').  
Stuffing must be processed before sending a packet and removed ('unstuffed') after receiving the packet.  
Notice that a simple <DLE> <ETX> sequence does not necessarily signify the end of the packet, as these can be bytes in the middle of a data string.  
The end of a packet is <ETX> preceded by an odd number of <DLE> bytes.

### 4.1.1.7 CRC CALCULATION

The type of the CRC is CRC16 V41.  
The CRC is computed as a function of the contents of the Data before Stuffing.  
The initial value is 0x0000.

### 4.1.1.8 PACKET IDENTIFIER

The identifier is formatted as follow:

Bit 7 (MSB)	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0 (LSB)
IN/OUT	F (First)	L (Last)	Reserved 0	Packet Type			

The MSB (Bit 7) is reserved for packet direction. Setting this bit set the direction to IN. Clear this bit to set the direction to OUT.

An OUT Packet is a packet sent by the Host to the MorphoAccess™.

An IN packet is a packet received by the Host from the MorphoAccess™.

Bit 6 (F) is reserved for Packet Order information. Set this bit when it is the first packet when transmitting a set of packets.

Bit 5 (L) is reserved for Packet Order information. Set this bit when it is the last packet when transmitting a set of packets.

Bit 4 is a reserved bit and must be clear.

The Bits 3 to 0 are used for packet identification:

The following packet type are implemented:

ID Value	Description
0x1	Data Packet
0x2	ACK Packet
0x4	NACK Packet

## 4.1.2 Frames sequence

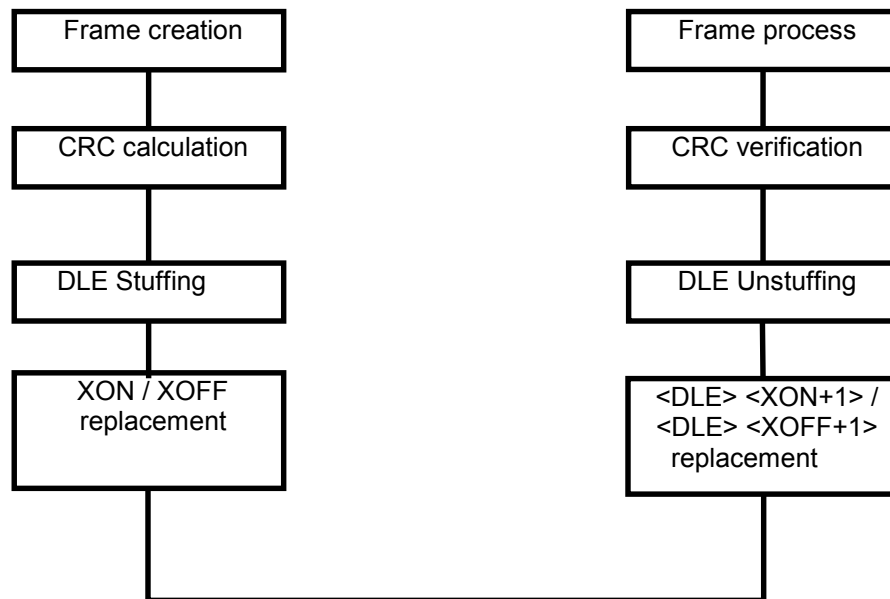


Figure 2 : Frames Sequence

## 4.1.3 Timing Characteristics

The maximum elapsed time between the transmission of two bytes of a frame is 100 ms.

## 4.1.4 Communication Error Case

The following error case must be detected:

- ◆ Timeout between the reception of two bytes (the timeout start after the reception of STX),
- ◆ Bad CRC verification,
- ◆ Unstuffing error (<DLE> is followed by an unexpected character).

## 4.2 RS232 SERVICE PROTOCOL

### 4.2.1 Request Counter management

The following rules have to be implemented:

- The RC of a data packet is filled with the current Counter value,
- The RC of an ACK (NACK) packet is filled with the RC of the data packet to ACK (NACK).
- On the reception of an ACK (NACK) packet, the RC is compare to latest data packet sent. If it is an ACK, the counter is increased on a hit. If several packets are received with the same RC, only one ACK is send.

## 4.2.2 Timing specification

The maximum elapsed time between the emission of a Packet Data and the reception of the ACK is 500 ms.

## 4.2.3 Error cases

- ◆ When a frame is not valid (Bad CRC, Unstuffing error, Rx Timeout), the receptor must send a NACK packet.
- ◆ When the emitter is waiting for an ACK (NACK) packet, all other received packet must be ignored.

## 4.2.4 Typical Transactions workflow

- ◆ Emission of a packet data that contain less than 1024 bytes of effective data.

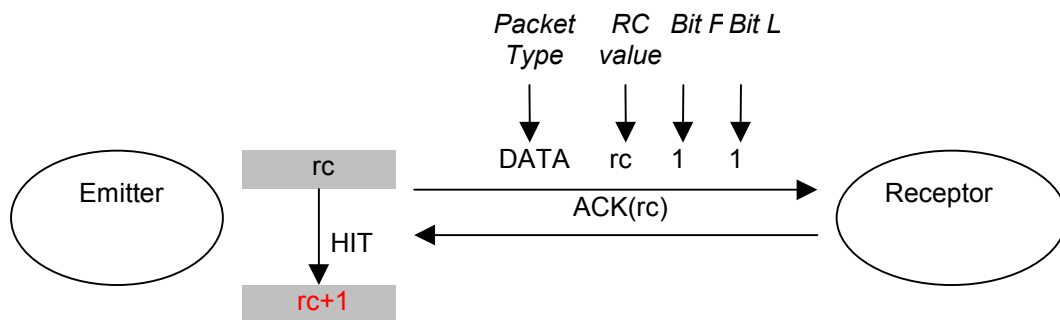
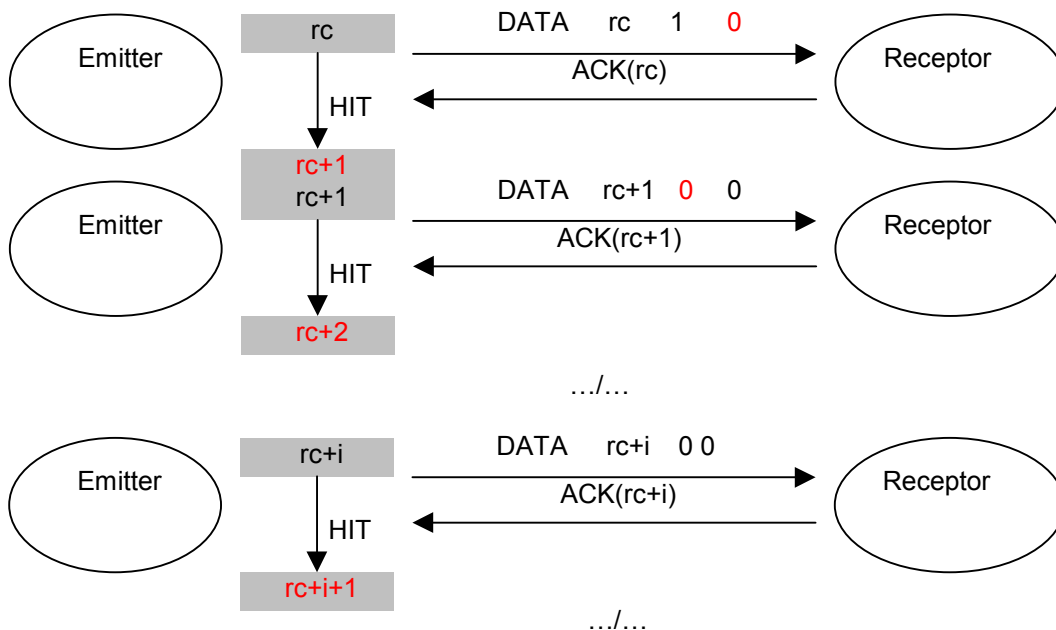


Figure 3 : Packet Transmission of Less Than 1024 Bytes of Effective Data

- ◆ Emission of a packet data that contain more than 1024 bytes of effective data.



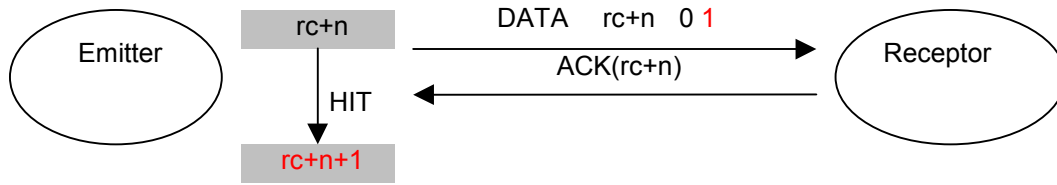
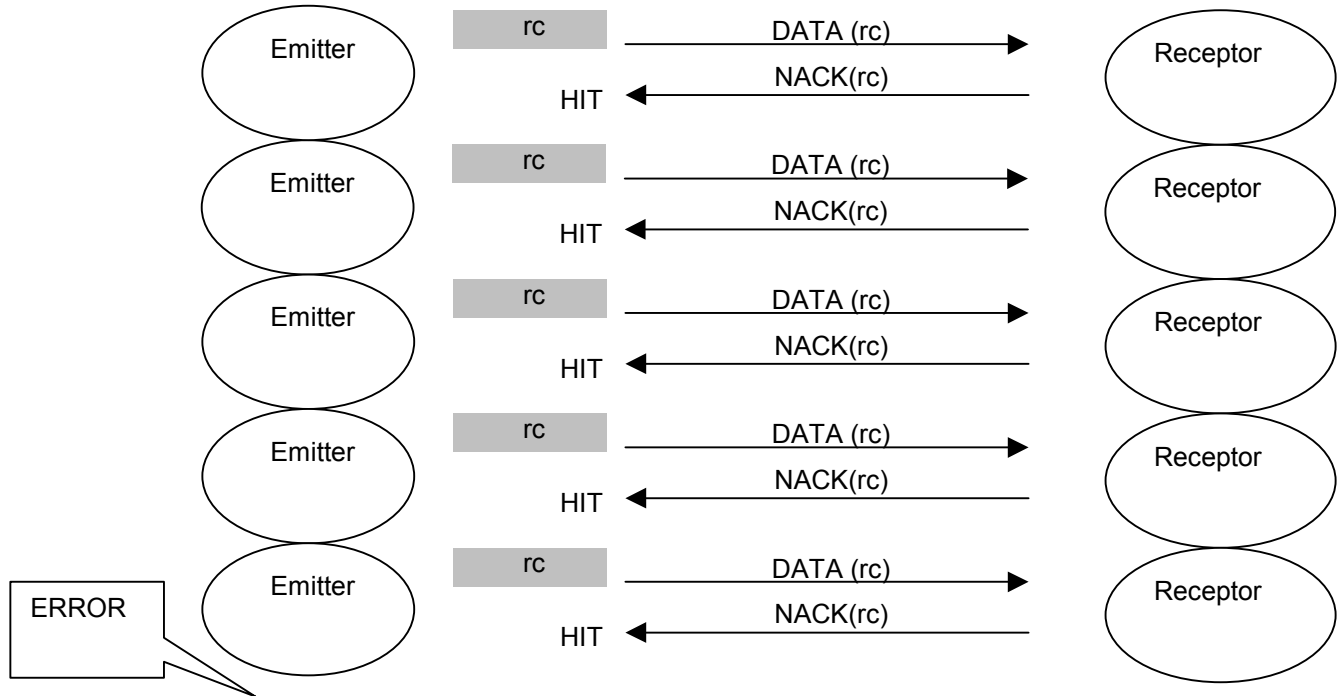


Figure 4 : Packet Transmission more than 1024 bytes

- ◆ An error occurred while transmitting the data packet



- ◆ The data packet is transmitted but the recepteur don't transmit ACK or NACK.

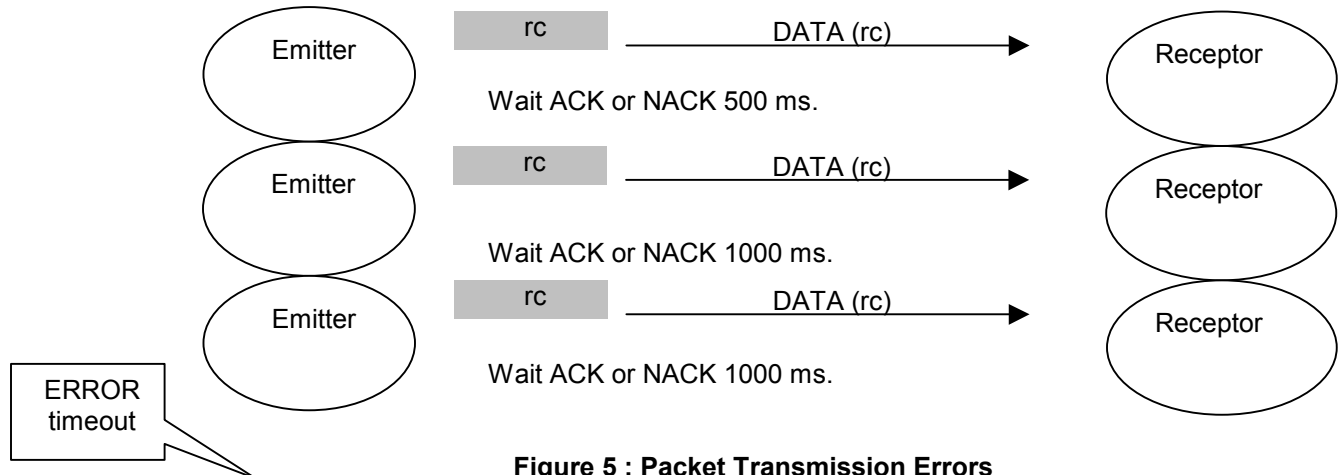


Figure 5 : Packet Transmission Errors



SAGEM SA

```

BB BB BB BB BB BB BB BB BB BB AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA FF FF FF FF FF FF FF
FF FF FF FF FF FF FF FF FF EE EE EE EE EE EE EE EE EE EE EE EE EE EE DD DD DD DD DD DD DD DD DD
DD DD DD DD DD DD DD CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC BB BB BB BB BB BB BB BB BB
BB BB BB BB BB AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA FF FF FF FF FF FF FF FF FF
FF FF FF EE EE EE EE EE EE EE EE EE EE EE EE EE EE EE DD DD DD DD DD DD DD DD DD DD DD
DD CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC BB BB BB BB BB BB BB BB
AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA
EE EE EE EE EE EE EE EE EE EE EE EE EE EE EE DD DD DD DD DD DD DD DD DD DD DD DD DD
CC CC CC CC CC CC CC CC CC CC CC BB BB BB BB BB BB BB BB BB BB BB BB AA AA AA AA AA AA
AA AA AA AA AA AA AA AA AA FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
EE EE EE EE EE EE EE EE DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD CC CC
CC CC CC CC 7E 47 10 03
11:18:19.477 : Packet Sent
11:18:19.497 : Waiting SOP...
  10 E2 01
11:18:19.517 : Received SOP      ID=0x02 F=1      L=1      RC=1
11:18:19.527 : Sending SOP      ID=0x01 F=0      L=0      RC=2
  10 01 02
11:18:19.567 : SOP Sent
11:18:19.577 : Sending Packet CRC=0x71 71
  BB BB BB BB BB BB BB BB BB BB BB BB BB BB BB AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA FF
FF FF FF FF FF FF FF FF FF FF EE EE EE EE EE EE EE EE EE EE EE EE EE EE EE EE EE EE EE
DD DD DD DD DD DD DD DD DD DD CC CC CC CC CC CC CC CC CC CC CC CC CC BB BB BB BB BB BB
BB BB BB BB BB BB BB BB BB BB BB AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA FF FF FF
FF FF FF FF FF FF FF EE EE EE EE EE EE EE EE EE EE EE EE EE EE EE EE DD DD DD DD DD DD
DD DD DD DD DD DD CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC BB BB BB BB BB BB
BB BB BB BB AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA FF FF
FF FF FF EE EE EE EE EE EE EE EE EE EE EE EE EE EE EE DD DD DD DD DD DD DD DD DD DD DD
CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC
AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA
EE EE EE EE EE EE EE EE EE EE EE EE EE EE EE DD DD DD DD DD DD DD DD DD DD DD DD DD
CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC BB BB BB BB BB BB BB BB BB BB BB BB AA AA
AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA
EE EE EE EE EE EE EE EE EE EE EE EE EE EE EE DD DD DD DD DD DD DD DD DD DD DD DD DD
CC CC CC CC CC CC CC CC CC CC CC BB BB BB BB BB BB BB BB BB BB BB BB AA AA AA AA AA AA
AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA
EE EE EE EE EE EE 71 71 10 03
11:18:29.431 : Packet Sent
11:18:29.441 : Waiting SOP...
  10 E2 02
11:18:29.471 : Received SOP      ID=0x02 F=1      L=1      RC=2
11:18:29.481 : Sending SOP      ID=0x01 F=0      L=0      RC=3
  10 01 03
11:18:29.521 : SOP Sent
11:18:29.531 : Sending Packet CRC=0x1E 1E
  DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC BB
BB BB BB BB BB BB BB BB BB BB BB BB BB BB AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA FF FF
FF FF FF FF FF FF FF FF FF FF FF FF FF FF EE EE EE EE EE EE EE EE EE EE EE EE EE EE EE DD DD DD
DD DD DD DD DD DD DD DD DD DD CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC BB BB BB BB
BB BB AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA
EE EE EE EE EE EE EE EE EE EE EE EE EE EE EE DD DD DD DD DD DD DD DD DD DD DD DD DD
CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC BB BB BB BB BB BB BB BB BB BB BB BB AA AA
AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA
EE EE EE EE EE EE EE EE EE EE EE EE EE EE EE DD DD DD DD DD DD DD DD DD DD DD DD DD
CC CC CC CC CC CC CC CC CC CC CC BB BB BB BB BB BB BB BB BB BB BB BB AA AA AA AA AA AA
AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA
EE EE EE EE EE EE DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD CC CC CC CC
CC CC CC BB BB BB BB BB BB BB BB BB BB BB BB BB BB AA AA AA AA AA AA AA AA AA AA AA AA
AA FF FF FF FF FF FF FF FF FF FF FF FF FF FF EE EE EE EE EE EE EE EE EE EE EE EE EE DD
DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD CC CC CC CC CC CC CC
BB BB BB BB BB BB BB BB BB BB BB BB BB BB BB BB AA AA AA AA AA AA AA AA AA AA AA AA
AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA
FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF EE EE EE EE EE EE EE EE EE
BB BB BB BB BB BB BB BB BB BB BB BB BB BB AA AA AA AA AA AA AA AA AA AA AA AA AA AA
FF FF FF FF FF FF FF FF FF FF EE EE EE EE EE EE EE EE EE EE EE EE EE EE EE EE EE EE

```



**SAGEM sa**

```
DD DD DD DD DD DD DD DD CC CC CC CC CC CC CC CC CC CC CC CC CC CC BB BB BB BB BB BB BB BB BB BB
BB BB BB BB BB BB BB AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA FF FF FF FF FF FF FF FF FF FF
FF FF FF FF FF EE EE EE EE EE EE EE EE EE EE EE EE EE EE EE EE DD DD DD DD DD DD DD DD DD DD DD
DD DD DD CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC BB BB BB BB BB BB BB BB BB BB BB BB BB BB
BB AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA
EE EE EE EE EE EE EE EE EE EE EE EE EE EE EE EE DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD
CC CC CC CC CC CC CC CC CC CC CC CC CC CC BB BB BB BB BB BB BB BB BB BB BB BB BB BB BB BB BB BB
AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA
EE EE EE EE EE EE EE EE EE EE DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD CC CC CC
CC CC CC CC CC CC CC CC CC CC CC CC CC CC BB BB BB BB BB BB BB BB BB BB BB BB BB BB BB BB BB BB
AA AA AA AA AA 1E 1E 10 03
11:18:39.696 : Packet Sent
11:18:39.706 : Waiting SOP...
  10 E2 03
11:18:39.736 : Received SOP   ID=0x02 F=1     L=1     RC=3
11:18:39.736 : Sending SOP    ID=0x01 F=0     L=1     RC=4
  10 21 04
11:18:39.776 : SOP Sent
11:18:39.796 : Sending Packet CRC=0x28 ED
  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF EE EE EE EE EE EE EE EE EE EE EE EE EE EE EE
DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC BB BB
BB BB BB BB BB BB BB BB BB BB BB BB BB BB AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA
FF FF FF FF FF FF FF FF FF FF FF FF EE EE EE EE EE EE EE EE EE EE EE EE EE EE EE EE EE EE DD DD DD
DD DD DD DD DD DD DD DD CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC BB BB BB BB BB BB BB BB
BB BB BB BB BB BB BB AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA ED 28 10 03
11:18:41.759 : Packet Sent
11:18:41.769 : Waiting SOP...
  10 E2 04
11:18:41.799 : Received SOP   ID=0x02 F=1     L=1     RC=4
```

### 4.3 TCP/IP PROTOCOL

For using the TCP/IP protocol, the PC is configured as a client socket, whereas the MorphoAccess terminal is a server socket.

#### 4.3.1 Port number

The socket is opened and closed with each command at the port 11010.

#### 4.3.2 Byte Order

The packet byte order is Little Endian: multi bytes data are sent least significant byte first (LSB).

### 4.4 RS422 PROTOCOL

The RS422 protocol is implemented in the same way that the RS232 one, with the same default configuration. For the moment it is not possible to configure it sending an ILV, but directly via the MorphoAccess.



## 5 Application Protocol

### 5.1 COMPATIBILITY AND RECOMMENDATION

This version of the documentation provides new commands allowing the initialisation or the configuration the MorphoAccess™ terminal. These additional commands are:

Initialization functions :

- ◆ **DEFAULT\_INIT**  
This function allows to initialize the module with the default configuration
- ◆ **GET\_VERSION**  
This function returns the serial number of the terminal, the revision of the software application and the Type of MorphoAccess
- ◆ **REBOOT**  
This function allows to reboot the MorphoAccess to initialize the application.
- ◆ **PING**  
This function allows to verify that the terminal respond correctly.

Configuration functions :

- ◆ **SET\_CONFIGURATION**  
This function initializes the module using custom parameters.
- ◆ **GET\_CONFIGURATION**  
This function retrieves the system configuration.

Beware: there is some new return codes in this version. For every return request, it is advised to test the request status in order to check if its value is different from the ILV\_OK.

**Note : Beware: return codes can be changed in futur versions.**

The following table describes modifications of the previous interface.

AUTHENTICATE	Request	It's now possible to indicate the minutes with the value ID_PKCOMP The timeout parameter is activated
	Reply	Changed the status code ILVERR_WAITING_BIO in ILVERR_BIO_INPROGRESS
ENROLL	Request	The timeout parameter is activated
	Reply	Changed the status code ILVERR_WAITING_BIO in ILVERR_BIO_INPROGRESS
IDENTIFY	Request	The timeout parameter is activated
	Reply	Changed the status code ILVERR_WAITING_BIO in ILVERR_BIO_INPROGRESS
ERASE BASE	Reply	Added ILVERR_BADPARAMETER in status code
ERASE_BASE_ALL	Reply	Added ILVERR_BADPARAMETER status code and ILVERR_BIO_INPROGRESS status code
ADD_BASE_RECORD	Reply	Changed the status code ILVERR_WAITING_BIO in ILVERR_BIO_INPROGRESS
GARBAGE_COLLECTOR	Reply	Added ILVERR_BADPARAMETER status Changed the status code ILVERR_WAITING_BIO in ILVERR_BIO_INPROGRESS
GET_BASE_CONFIG	Reply	Changed the status code ILVERR_WAITING_BIO in ILVERR_BIO_INPROGRESS

## 5.2 BASE STRUCTURE

The MorphoAccess Base format is :

- ◆ A Bio formatted base (With 3 Fields)
  - User Data
  - PKs
  - Internal Field
- ◆ Additional Fields (0 to 7) to store User Data.

User ID 25 bytes in ASCII	Pks 217 bytes * Nb Fingers	Additional Field 1 3 bytes	Additional Fields i Length limited to 128 bytes	
				} a record: a person
				} another person
				.....

**Table 2: MorphoAccess Local Database Structure**

### Remarks :

- ◆ A person is represented by a record and can be found in the Database thanks to his User Id (Index in the database)
- ◆ The size of the PK buffer is 256 bytes while it is sent by the PC, 512 bytes during the biometric process and 217 bytes once it is stored in the database.
- ◆ Field one is reserved for administration purpose. When creating additional fields or to use the admin/user status feature, you must create this administrator field. First byte set to 1 enables administrator login. First byte set to 0 disables administrator login.
- ◆ The Additional Fields are defined in the CREATE DATABASE command description. Please refer to 5.8.1.
- ◆ Given that memory size is fixed, as the number of fields increases (increasing the size of each individual record), the number of enrolled people the database can hold decreases. All the data stored in this base are ASCII characters strings.

## 5.3 DATA STRUCTURE

The application data has three fields:

1. Identifier called I ; this is the identifier of the command,
2. Length called L; this is the length of the *Value* field in byte,
3. Value called V; this is the parameters or data.

This data structure is variable. The Value field can contain optional ILV formatted data. Its length is variable.

The I field has 8 bits length and the L field has 16 bits length, as shown below:

	Length	Value
1 byte	2 bytes	<Length> bytes

*SAGEM SA*

For a length value > 64k, the ILV is formatted as follow:

Identifier	Length	Length value	Value
1 byte	0xFFFF	4 bytes	<Length> bytes

**Remarks :** All the parameters are in bytes. Only ASCII or Corresponding table parameters are specified.

If a request doesn't correspond to a code function describes in following chapter, the code reply is  
CMD\_INVALIDREQUEST = 0x50.

## 5.4 INITIALIZATION FUNCTIONS DESCRIPTIONS

### 5.4.1 DEFAULT\_INIT ID=0x0A

This function allows to initialize the module with the default configuraton.

#### 5.4.1.1 REQUEST

Identifier value	0x0A
Length value	0x0000
Value(Parameters)	None

#### 5.4.1.2 REPLY

Identifier value	0x0A
Length value	0x0001
Value(Parameters)	Request Status            1 byte

#### Request Status:

<b>ILV_OK</b>	The execution of the function succeeded
<b>ILVERR_CMD_INPROGRESS</b>	A command is already in progress
<b>ILVERR_BADPARAMETER</b>	Input parameters are not valid
<b>ILVERR_ERROR</b>	An error occurred during the execution of the function.

#### Note:

If the request status is ILV\_OK, the MorphoAccess™ terminal is configured with the default values:

- ◆ ID\_COM1 and ID\_COM2
  - Baudrate                    9600 bauds
  - Data bits                    7 bits
  - Parity                        Even
  - Stop bits                    2 bits
  - Flow Control                no
  - Send ( COM1/COM2)        no
  - Format                        0 (ASCII format)
- ◆ Net Configuration
  - Ip Address                   134.1.32.214
  - SubNet                       255.255.0.0
  - Default Gateway            134.1.6.20
- ◆ Relay
  - Relay                        active
  - Timeout                     3000 ms
- ◆ Wiegand/DataClock out
  - Format                        0 (ASCII format)
  - Send                         no
- ◆ Wiegand/DataClock options
  - facility code                7 (used to discriminate Wiegand ID)
  - dataclock levels            0 (normal data normal strobe)
  - led in                        0 (MorphoAccess™ led is not driven by card reader)
  - led out/card present       0 (Led signal is not forced)
- ◆ Language
  - Default                      0 (english)

*SAGEM SA*

- ◆ Morpho Access application
  - Mode 0 (identification mode)
  - Host Com 3 (ethernet)
  - Log file 0 (log disabled)
  - Deister 0 (Deister is disabled)
  - Admin ident 0 (administrator biometric identification is disabled)
  - Time mask 0 (Time zone feature is disabled)
- ◆ Graphical User Identify
  - Change language 0 (no language change allowed)
  - Display hour 0 (clock is not displayed)
  - Welcome field 1 1 (only User ID is shown on positive identification...)
  - Welcome field 2 0 (...and nothing else)
  - Idle time 0 (no sleep mode)

To validate the new values, it is necessary to reboot the terminal. The REBOOT command can be used.

## 5.4.2 GET\_VERSION ID=0x03

This function returns the serial number of the terminal, the revision of the software application and the MorphoAccess type.

### 5.4.2.1 REQUEST

Identifier value	0x03
Length value	0x0000
Value(Parameters)	None

### 5.4.2.2 REPLY

Identifier value	0x03	
Length value	0x0011	
Value(Parameters)	Request Status	1 byte
	Serial Number	9 bytes
	Software Revision	4 bytes
	Extended Memory	1 byte
	MorphoAccess Type	2 byte

#### Request Status:

<b>ILV_OK</b>	The execution of the function succeeded
<b>ILVERR_CMD_INPROGRESS</b>	A command is already in progress
<b>ILVERR_BADPARAMETER</b>	Input parameters are not valid
<b>ILVERR_ERROR</b>	An error occurred during the execution of the function.

**Serial Number:** Serial Number of the MorphoAccess. The output buffer is a string of ASCII characters.

**Software Revision:** The software revision is formatted as below

b6	b5	b4	b3	b2	b1	b0	b7	1 byte	1 byte	1 byte(LSB)
R	R	R	R	R	R	R	R	R	Major revision	Minor revision

**Storage Memory:** This command gives the size of the flash memory, in megabytes. This size is variable, according to the MorphoAccess type.

**MorphoAccess Type:** To get the MorphoAccess type ( MA200, MA300). The output value is 200 or 300. This value is sent in big endian encoding.

### 5.4.3 REBOOT ID=0x04

This function allows to reboot the MorphoAccess in order to initialize the application. For example, after net parameters changes, it is necessary to reboot the system.

#### 5.4.3.1 REQUEST

Identifier value	0x04
Length value	0x0000
Value(Parameters)	None

#### 5.4.3.2 REPLY

Identifier value	0x04
Length value	0x0001
Value(Parameters)	Request Status            1 byte

**Request Status:**

- ILVERR\_CMD\_INPROGRESS**    A command is already in progress
- ILVERR\_BADPARAMETER**     Input parameters are not valid

**Notes:**

If the reboot command is successful, the system cannot send a reply ILV since it is rebooting.

## 5.4.4 PING ID=0x08

This function allows to check that the terminal responds correctly.

### 5.4.4.1 REQUEST

Identifier value	0x08
Length value	0x0000
Value(Parameters)	None

### 5.4.4.2 REPLY

Identifier value	0x08
Length value	0x0001
Value(Parameters)	Request Status            1 byte

#### Request Status:

<b>ILV_OK</b>	The terminal answered.
<b>ILVERR_CMD_INPROGRESS</b>	A command is already in progress
<b>ILVERR_BADPARAMETER</b>	Input parameters are not valid



## 5.5 CONFIGURATION FUNCTIONS DESCRIPTIONS

### 5.5.1 SET\_CONFIGURATION ID=0x01

This function initializes the module using custom parameters.

#### 5.5.1.1 REQUEST

Identifier value	0x01																				
Length value	0x0001 + <L <sub>s</sub> > + <L <sub>w</sub> > + <L <sub>t</sub> > + <L <sub>r</sub> > + <L <sub>d</sub> > + <L <sub>o</sub> > + <L <sub>M</sub> > + <L <sub>G</sub> > + <L <sub>L</sub> >																				
Value(Parameters)	<table border="0"> <tr> <td>MorphoAccess Mode</td> <td>1 byte</td> </tr> <tr> <td>Serial Link Configuration</td> <td>L<sub>s</sub> bytes</td> </tr> <tr> <td>Wiegand/DataClock Configuration</td> <td>L<sub>w</sub> bytes</td> </tr> <tr> <td>Network Configuration</td> <td>L<sub>t</sub> bytes</td> </tr> <tr> <td>Relay Configuration</td> <td>L<sub>r</sub> bytes</td> </tr> <tr> <td>Date and Time Configuration</td> <td>L<sub>d</sub> bytes</td> </tr> <tr> <td>Wiegand/DataClock Options</td> <td>L<sub>o</sub> bytes</td> </tr> <tr> <td>MorphoAccess™ Application</td> <td>L<sub>M</sub> bytes</td> </tr> <tr> <td>Graphical user interface parameters</td> <td>L<sub>G</sub> bytes</td> </tr> <tr> <td>Language Parameters</td> <td>L<sub>L</sub> bytes</td> </tr> </table>	MorphoAccess Mode	1 byte	Serial Link Configuration	L <sub>s</sub> bytes	Wiegand/DataClock Configuration	L <sub>w</sub> bytes	Network Configuration	L <sub>t</sub> bytes	Relay Configuration	L <sub>r</sub> bytes	Date and Time Configuration	L <sub>d</sub> bytes	Wiegand/DataClock Options	L <sub>o</sub> bytes	MorphoAccess™ Application	L <sub>M</sub> bytes	Graphical user interface parameters	L <sub>G</sub> bytes	Language Parameters	L <sub>L</sub> bytes
MorphoAccess Mode	1 byte																				
Serial Link Configuration	L <sub>s</sub> bytes																				
Wiegand/DataClock Configuration	L <sub>w</sub> bytes																				
Network Configuration	L <sub>t</sub> bytes																				
Relay Configuration	L <sub>r</sub> bytes																				
Date and Time Configuration	L <sub>d</sub> bytes																				
Wiegand/DataClock Options	L <sub>o</sub> bytes																				
MorphoAccess™ Application	L <sub>M</sub> bytes																				
Graphical user interface parameters	L <sub>G</sub> bytes																				
Language Parameters	L <sub>L</sub> bytes																				

#### MorphoAccess Mode:

It is not available, set to 0. This parameter is mandatory.

#### Serial Link Configuration:

This packet configures the serial link.

Identifier value	ID_COM1 or ID_COM2														
Length value	0x0007														
Value	<table border="0"> <tr> <td>Speed</td> <td>1 byte</td> </tr> <tr> <td>Databits</td> <td>1 byte</td> </tr> <tr> <td>Stop bits</td> <td>1 byte</td> </tr> <tr> <td>Parity</td> <td>1 byte</td> </tr> <tr> <td>Flow control</td> <td>1 byte</td> </tr> <tr> <td>Send</td> <td>1 byte</td> </tr> <tr> <td>Format</td> <td>1 byte</td> </tr> </table>	Speed	1 byte	Databits	1 byte	Stop bits	1 byte	Parity	1 byte	Flow control	1 byte	Send	1 byte	Format	1 byte
Speed	1 byte														
Databits	1 byte														
Stop bits	1 byte														
Parity	1 byte														
Flow control	1 byte														
Send	1 byte														
Format	1 byte														

#### Notes:

- See the corresponding table for the serial parameters at the end of the document.
- You can define up to two ILV packets for serial link configuration using ID\_COM1 and ID\_COM2

**SAGEM SA**

**Wiegand/DataClock Configuration:**

This packet enables or disables the output Wiegand/DataClock. You can specify the output format of the ID.

Identifier value	ID_WIEGAND or ID_DATALOCK
Length value	0x0002
Value	Format                    1 byte
	Send                        1 byte

Send :                    To activate/deactivate the output port  
Format :                It is not available, set to 0. This parameter is mandatory.

**Notes:**

- Using Wiegand, ID is sent in binary format.
- Using Dataclock, ID is sent in BCD format.
- For historical reasons, you can define two ILV packets using ID\_WIEGAND and ID\_DATALOCK but configuring one or another as the same effect.

**Network Configuration:**

This packet configures the Ip adress of the terminal.

Identifier value	ID_TCPIP
Length value	0x0000 + <L>
Value	Ip adress                    8..16 bytes
	Subnet                      8..16 bytes
	Ip adress Gateway        8..16 bytes

Ip adress :                the Ip adress of the MorphoAccess ( use for the TCP/IP connect )  
Subnet :                    The mask of the subnet<sup>1</sup>  
Ip adress Gateway :      The Ip adress to the gateway.

**Notes:**

- To validate the new values it is necessary to reboot the terminal. The REBOOT can be used.
- The three parameters are ASCII null terminated strings. The format of the IP address is aaa.bbb.ccc.ddd.
- The default Ip adress is 134.1.32.214

**Relay Configuration:**

This packet activates or deactivates the relay.

Identifier value	ID_RELAY
Length value	0x0002
Value	Relay mode                1 byte
	Timeout activate        1 byte

Relay mode :              Activate/ Deactivate the relay of the morphoAccess<sup>2</sup>.  
Timeout activate :        To configure the release time of the relay.

**Notes:**

- See the coresponding table of the relay parameters at the end of the document.

<sup>1</sup> Contact your network administrator for the values.

<sup>2</sup> See the Terminal block board in Installation guide document.

**SAGEM SA**

**Date and Time Configuration:**

This packet configures the time and the date of the system.

Identifier value	ID_DATE
Length value	0x000D
Value	Time/Date system 13 bytes

Time/Date system : A string of ASCII characters with **Day**, **Month**, **Years**, **Hours**, **minutes** and **Seconds** each encoded with 2 bytes: "DDMMYYHHmmSS". For example the string "010601120000" indicates the 06/01/01 at 12:00:00 clock.

**Notes:**

- The string is null terminated.

**Wiegand/DataClock Options :**

This packet configures additional parameters of the Wiegand/DataClock.

Identifier value	ID_WIEGAND_OPTION
Length value	0x0004
Value	Facility Code 1 byte Dataclock Levels 1 byte Led In 1 byte Led Out / Card Present 1 byte

- Facility Code :           ▪ 0-255. Used to differentiates Wiegand ID
- Dataclock Levels :       ▪ 0 Data and strobe are not inverted.  
                              ▪ 1 Strobe is inverted.  
                              ▪ 2 Data is inverted.  
                              ▪ 3 Data and strobe are inverted.
- Led In :                    ▪ 0 The led signal is ignored.  
                              ▪ 1 Led is driven by led signal.
- Led Out / Card Present :  ▪ 0 Card Present signal in Dataclock mode.  
                              ▪ 0 Led Out signal in Wiegand mode.  
                              ▪ 1 Signal is forced to led out.

**Notes:**

- To validate the new values it is necessary to reboot the terminal. The REBOOT can be used.

**SAGEM SA**

**Mopho Access Application:**

This packet configures parameters of the application.

Identifier value	ID_APPLI_OPTION	
Length value	0x0006	
Value	Host Com	1 byte
	Mode	1 byte
	Log File	1 byte
	Deister	1 byte
	Admin identification	1 byte
	Time mask	1 byte

- Host Com :
  - 0 for “Stand Alone” mode.
  - 1 selects the first serial link (COM1).
  - 2 selects the second serial link (COM2).
  - 3 selects the ethernet port.
- Mode :
  - 0 for Identification mode.
  - 1 for Authentication mode.
  - 2 for Proxy mode.
- Log File :
  - 0 disables logs file.
  - 1 enables logs file.
- Deister :
  - 0 disables Deister contactless card reader connection.
  - 1 enables Deister contactless card reader connection.
- Admin identification
  - 0 disables biometric administrator identification.
  - 1 enables biometric administrator identification.
- Time mask
  - 0 disables time zone feature.
  - 1 enables time zone feature.

**Notes :**

- To validate the new values it is necessary to reboot the terminal. The REBOOT can be used.

**Graphical user interface configuration:**

This packet configures graphical user interface parameters.

Identifier value	ID_GUI	
Length value	0x0005	
Value	Change language	1 byte
	Display hour	1 byte
	Welcome field 1	1 byte
	Welcome field 2	1 byte
	Idle time	1 byte

- Change language:
  - 0 the user can't change the language.
  - 1 the user is allowed to change the language.
- Display hour:
  - 0 disables clock display.
  - 1 enables clock display.
- Welcome field 1:
  - 0 – 9 user first additional field to be displayed when identified.
- Welcome field 2:
  - 0 – 9 user second additional field to be displayed when identified.
- Idle Time
  - Time in minutes after which MorphoAccess™ enters sleep mode.

**Notes :**

- To validate the new values it is necessary to reboot the terminal. The REBOOT can be used.

*SAGEM SA*

**Language Parameters:**

This packet allows to parameter the language service.

Identifier value	ID_LANGUAGE		
Length value	0x0001		
Value	Default language		1 byte

- Default language:
- 0 for English.
  - 1 for Spanish.
  - 2 for French.
  - 3 for user language.

**Remarks:**

- Serial Link, Wiegand/dataclock, Network, Relay, Date, Time Configuration, Wiegand/DataClock options, Application, GUI and Language are optional.
- All this parameters are described in configuration table see chapter 5.22.

## 5.5.1.2REPLY

Identifier value	0x01	
Length value	0x0001	
Value(Parameters)	Request Status	1 byte

### Request Status:

<b>ILV_OK</b>	The execution of the function succeeded
<b>ILVERR_ERROR_SL</b>	An error occurred during the execution of Serial Link Configuration.
<b>ILVERR_ERROR_WIEGAND</b>	An error occurred during the execution of Wiegand/Dataclock Configuration.
<b>ILVERR_ERROR_NETWORK</b>	An error occurred during the execution of TCP/IP Configuration.
<b>ILVERR_BADPARAMETER</b>	One of the parameter of the request is incorrect.
<b>ILVERR_CMD_INPROGRESS</b>	Another command is already in progress
<b>ILVERR_ERROR</b>	An error occurred during the execution of the function.

## 5.5.2 GET\_CONFIGURATION ID=0x06

This function retrieves the system configuration.

### 5.5.2.1 REQUEST

Identifier value	0x06
Length value	0x0000 + <i>
Value(Parameters)	Parameter Identifier 1 Parameter Identifier 2 ..... Parameter Identifier i

#### Parameter Identifier:

It is used to determine which parameter will be returned. It must be a list identifier like ID\_COM1, ID\_COM2, ID\_WIEGAND, ID\_DATACLOCK, ID\_RELAY, ID\_TCPIP, ID\_DATE, ID\_WIEGAND\_OPTION, ID\_APPLI\_OPTION, ID\_GUI, ID\_LANGUAGE.

### 5.5.2.2 REPLY

Identifier value	0x06
Length value	0x0002 + <L <sub>s</sub> > + <L <sub>w</sub> > + <L <sub>t</sub> > + <L <sub>r</sub> > + <L <sub>d</sub> > + <L <sub>o</sub> > + <L <sub>m</sub> > + <L <sub>G</sub> > + <L <sub>L</sub> >
Value(Parameters)	Request Status 1 byte MorphoAccess Mode 1 byte Serial Link Configuration L <sub>s</sub> bytes Wiegand/DataClock Configuration L <sub>w</sub> bytes Network Configuration L <sub>t</sub> bytes Relay Configuration L <sub>r</sub> bytes Date and Time Configuration L <sub>d</sub> bytes Wiegand/DataClock Options L <sub>o</sub> bytes Morpho Access Application L <sub>m</sub> bytes Graphical user interface parameters L <sub>G</sub> bytes Language Parameters L <sub>L</sub> bytes

#### Request Status:

ILV_OK	The execution of the function succeeded
ILVERR_ERROR	An error occurred during the execution of the function.
ILVERR_CMD_INPROGRESS	Another command is already in progress

#### MorphoAccess Mode :

It is not available. Set to 0.

#### Notes :

- See the request SET\_CONFIGURATION command to get the details of ILV packet, Serial Link configuration, Network configuration, Relay configuration, Date/Time configuration, Application configuration, Graphical user interface configuration and Language configuration.

## 5.6 SECURITY CONFIGURATION

These parameters allow to configure some aspects of security. For the moment it is a way to make a DSA on an authentication command

### 5.6.1 SET\_PUBLIC\_KEY ID=0x53

#### 5.6.1.1 REQUEST

Identifier value	0x53	
Length value	0x0001+0x0194	
Value(Parameters)	Management of a signature for Pk	1 byte
	Public Key	128 bytes
	Parameter p	128 bytes
	Parameter q	20 bytes
	Parameter g	128 bytes

#### Management of the signature:

With this parameter a verification of the signature of Pk send by the host or read in a contactless card is performed.

0x00	The verification of Pk by a signature is not performed
0x01	A verification of Pk is performed using a signature

The value by default is 0x00. If this utility is activated, a signature must be send in the same packet. Every time that the MorphoAccess reboot these parameters have to be initialized.

#### Public Key:

If the management of the signature is on, a public key has to be sent byte per byte from the least significant byte to the most significant one.

#### Parameters p,q,g:

As for the public key these parameters have to be send when a management of signature is performed (from the least significant byte to the most significant one).



## 5.6.1.2 REPLY

Identifier value	0x53	
Length value	0x0001	
Value(Parameters)	Request Status	1 byte

If the function does not succeed, The MorphoAccess is configured with values by default.

### Request Status:

ILV_OK	The function completed successfully
ILVERR_BADPARAMETER	Bad input parameters
ILVERR_ERROR	An error occurred during the execution of the function.

### Notes:

When the Deister contactless card reader option is not activated on the MorphoAccess™ terminal, the command is invalid. Therefore its reply is CMD\_INVALID\_REQUEST (ID = 0x50).

## 5.7 BIOMETRICS FUNCTIONS DESCRIPTION

### 5.7.1 AUTHENTICATE ID=0x20

This fonction captures a finger and check if it matches with the minutiae file sent to the terminal.

#### 5.7.1.1 REQUEST

Identifier value	0x20	
Length value	0x0005 + <L1+3> + <L2+3> + <L+3>	
Value (Parameters)	Timeout	2 bytes
	Matching threshold	1 byte
	Acquisition quality threshold	1 byte
	Biometrics Data 1	L1+3 bytes
	Biometrics Data 2	L2+3 bytes
	User ID	L+3 bytes

#### Timeout:

Finger detection timeout in seconds. Its value must be between 0 and 30.  
 A value of 0 corresponds to an infinite timeout.

#### Matching Threshold:

This parameter can be set to values from 0 to 10. This parameter specifies how tight the matching threshold is. Threshold scoring values are identified below:

0	Low threshold for test purpose only.	There are few rejections, but many recognitions.
1	Very few persons rejected.	Tolerant application.
...		
5	Intermediate threshold.	Intermediate.
...		
9	Very high threshold (few false acceptances).	Secure Application.
10	High threshold for test purpose only.	There are very few recognition, many rejections.

#### Acquisition Quality Threshold:

Not used in MorphoAccess. Set to 0.

**SAGEM SA**

**Biometrics data:**

One (resp. two) ILV formatted data containing the reference minutiae of one (resp two) finger.

Identifier value	ID_PK_COMP or ID_PK_MAT	
Length value	0x0000 + <L>	
Value (Parameters)	Minutiae	L bytes

**Identifier:**

- If the identifier is ID\_PK\_MAT, the minutiae buffer must be 512 chars.
- If the identifier is ID\_PK\_COMP, size of minutiae buffer should be between 170 and 256.

**User ID:**

One ILV formatted data containing personal user data or other informations. This field can be used to create log files or display informations during users authentication.  
The Identifier must be ID\_USER\_ID.

Identifier value	ID_USER_ID	
Length value	0x0000 + <L>	
Value (Parameters)	User data	L bytes

## 5.7.1.2REPLY

Identifier value	0x20	
Length value	0x0002	
Value (Parameters)	Request status	1 byte
	Matching result	1 byte

### Request status:

<b>ILV_OK</b>	The function succeeded.
<b>ILVERR_ERROR</b>	An error occurred during the execution of the function.
<b>ILVERR_BADPARAMETER</b>	The matching threshold value or timeout value are not in the range or there is no input biometrics data.
<b>ILVERR_INVALID_MINUTIAE</b>	The reference ILV minutiae is not valid: bad identifier, corrupted minutiae.
<b>ILVERR_TIMEOUT</b>	The finger detection timeout has expired.
<b>ILVERR_CMD_INPROGRESS</b>	Another command is already in progress.
<b>ILVERR_ADMIN_EXECUTING</b>	In mode administrator.
<b>ILVERR_BIO_INPROGRESS</b>	An identification or an authenticate is processing, retry later.

### Matching result:

<b>ILVSTS_HIT</b>	The comparison succeeded.
<b>ILVSTS_NO_HIT</b>	It is not the same finger.
<b>ILVSTS_LATENT</b>	The system detected a latent fingerprint in the input fingerprint.

## 5.7.2 ENROLL ID=0x21

This function captures and enrolls a live finger. Number of fingers and type of enrollment can be specified. The calculated minutiae can be also exported to the host.

### 5.7.2.1 REQUEST

Identifier value	0x21	
Length value	0x0008 + <L+3> + <L1+3> + ... <Li+3>	
Value (Parameters)	Database identifier	1 byte
	Timeout	2 bytes
	Acquisition quality threshold	1 byte
	Enrollment type	1 byte
	Number of fingers	1 byte
	Save record	1 byte
	Export minutiae	1 byte
	User ID	L+3 bytes
	Additional user data field 1	L1+3 bytes
	...	
	Additional user data field i	Li+3 bytes

#### Database identifier:

It will be used if the Terminal contains several local bases. This is the same identifier that in the *CREATE DATABASE* command. With MA300, this value must be between 0 and 15.

If *Save record* field in the request is set to FALSE this value is ignored.

#### Timeout:

Finger detection timeout in second. Its value must be between 0 and 30.

A value of 0 corresponds to an infinite timeout.

#### Acquisition Quality Threshold:

Not used in MorphoAccess. Set to 0.

#### Enrollment type:

It defines the number of acquisitions per finger.

- Set this value to 0x00 for default enrollment; the minutiae are calculated from two-finger print image acquisition.
- Set this value to 0x01 for simple enrollment; the minutiae are calculated from 1 finger print image acquisition.

#### Number of fingers:

The number of finger to enroll. This function can enroll 1 or 2 fingers.

- Set this value to 0x01 to enroll 1 finger per user.
- Set this value to 0x02 to enroll 2 fingers per user.

#### Save Record:

Set this Boolean to TRUE to store calculated minutiae into the locale database. Otherwise set it to FALSE.

<sup>3</sup> See the base structure.

**SAGEM SA**

**Export Minutiae:**

Defines the format of the exported minutiae.

- Set this value to 0x00 will not include the calculated minutiae in the reply.
- Set this value to 0x01 to export the minutiae in the default format (compressed minutiae). In this case the PK buffer is 256 char compressed PKs. The corresponding Id is PK\_COMP.
- Other values ranged in [0xAA, 0xFF] define the size of the returned compressed minutiae. For example PK\_COMP\_175 (0xAF) corresponds to a 175 bytes returned PK. Authorised value are PK\_COMP\_170 to PK\_COMP\_255.

**User ID:**

One ILV formatted data containing personal user data to be saved into the database if the *Save Record* value is set to TRUE. Those data can be retrieved by the *IDENTIFY* function under Hit condition.

The Identifier must be ID\_USER\_ID.

Identifier value	ID_USER_ID	
Length value	0x0000 + <L>	
Value (Parameters)	User data	L bytes

**Additional User Data field j:**

**Identifier:**

- ID\_PUC\_DATA (If value is a string),
- ID\_C\_DATA (for a character),
- ID\_L\_DATA for a long,
- ID\_S\_DATA for a short.

The content of the data is not interpreted by the system. The content of the buffer must be defined by the user. All the additional user data fields defined in the *CREATE DATABASE* command have to be set in the *ENROLL* command.

Identifier value	ID_PUC_DATA or ID_C_DATA or ID_L_DATA or ID_S_DATA.	
Length value	0x0000 + <Lj>	
Value (Parameters)	User data	Lj bytes

**Note:**

Size and order of additional data fields must be consistent with database structure.

## 5.7.2.2 REPLY

Identifier value	0x21	
Length value	0x0006 + <L1+3> + <L2+3>	
Value (Parameters)	Request status	1 byte
	Enroll status	1 byte
	User database index	4 bytes
	Biometrics Data 1	L1+3 bytes
	Biometrics Data 2	L2+3 bytes

### Request status:

<b>ILV_OK</b>	The execution of the function succeeded.
<b>ILVERR_ERROR</b>	An error occurred during the execution of the function.
<b>ILVERR_BADPARAMETER</b>	One or more input parameters are out of range.
<b>ILVERR_INVALID_USER_DATA</b>	The input ILV user data is not valid: bad identifier or wrong size.
<b>ILVERR_TIMEOUT</b>	The finger detection timeout has expired.
<b>ILVERR_CMD_INPROGRESS</b>	Another command is already in progress.
<b>ILVERR_BIO_INPROGRESS</b>	An identification or an authenticate is processing.
<b>ILVERR_BASE_NOT_FOUND</b>	The specified base does not exist.

### Enroll status:

<b>ILVSTS_OK</b>	The enrollment succeeded.
<b>ILVSTS_DB_FULL</b>	This status can be returned if the <i>Save Record</i> is TRUE. It means that the maximum number of users that can be stored in the local database has been reached.
<b>ILVERR_BADPARAMETER</b>	One or more input parameters are out of range.

If the *request status* differs from **ILV\_OK**, the *enroll status* is not returned.

### User database index:

This is the database index of the record (person).  
 If *Save record* field in the request was set to FALSE the returned value is 0xFFFFFFFF.  
 If the *request status* differs from **ILV\_OK** or the *enroll status* differs from **ILVSTS\_OK** the *User database index* is not returned.

### Biometrics data:

Identifier value	ID_PK_COMP	
Length value	0x0000 + <L>	
Value (Parameters)	Minutiae	L bytes

The length parameter is equal to the *Export minutiae* value in the request. The range is [0x00AA, 0x0100].

### 5.7.3 IDENTIFY ID=0x22

This function identifies a live finger against the local database.

#### 5.7.3.1 REQUEST

Identifier value	0x22	
Length value	0x0006	
Value(Parameters)	Database identifier	1 byte
	Timeout	2 bytes
	Matching threshold	2 bytes
	Acquisition Quality Threshold	1 byte

**Database identifier:**

With MorphoAccess™ 200, it is the same identifier used in the CREATE DATABASE Command.  
 With MorphoAccess™ 300, this value can be between 0 and 15.

**Timeout:**

Finger detection timeout in seconds. Zero value corresponds to an infinite timeout.

**Matching Threshold:**

This parameter can be set to values from 0 to 10. This parameter specifies how tight the matching threshold is. Threshold scoring values are identified below. For example :

0	Low threshold for test purpose only	There are few rejections, but many recognition
1	Very few persons rejected	Tolerant application
...		
5	Intermediate threshold ( default value )	Intermediate
...		
9	Very high threshold (few false acceptances)	Secure Application
10	High threshold for test purpose only	There are very few recognition, and many rejections

**Acquisition Quality Threshold:**

Not yet used, set to 0.



### 5.7.3.2 REPLY

Identifier value	0x22	
Length value	0x0006 + <L <sub>1</sub> >	
Value(Parameters)	Request Status	1 byte
	Matching Result	1 byte
	User Database Index	4 bytes
	User ID	L <sub>1</sub> bytes

#### Status:

<b>ILV_OK</b>	The execution of the function succeeded
<b>ILVERR_ERROR</b>	An error occurred during the execution of the function.
<b>ILVERR_BADPARAMETER</b>	The matching threshold value or timeout value are not in the range
<b>ILVERR_TIMEOUT</b>	The finger detection timeout has expired
<b>ILVERR_CMD_INPROGRESS</b>	Another command is already in progress
<b>ILVERR_BIO_INPROGRESS</b>	An Identification or an authenticate is processing, retry later
<b>ILVERR_BASE_NOT_FOUND</b>	The specified base doesn't exist

#### Matching Result:

This is the result of the identification of the coded fingerprint in the database.

<b>ILVSTS_HIT</b>	The comparison succeeded
<b>ILVSTS_NO_HIT</b>	It is not the same finger
<b>ILVSTS_LATENT</b>	The system detected a latent fingerprint in the input fingerprint.

If the request status differs from ILV\_OK, the matching result is not returned.

#### User Database index:

This is the index database of the record (person). If the request status differs from ILV\_OK, or the matching status differs from ILVSTS\_HIT, the User Database index is not returned.

#### User ID:

Identifier value	ID_UDER_ID	
Length value	0x0000 + <L>	
Value	User Data field	L bytes

One ILV formatted data containing personal user data saved in the database. The maximum size for the user data is 25 bytes. If the request status differs from ILV\_OK, or if the matching status differs from ILVSTS\_HIT, the User Data are not returned.

**Remark:** if the database is empty, the function will stop immediately.

## 5.8 DATABASE FUNCTIONS DESCRIPTION

### 5.8.1 CREATE DATABASE ID=0x30

This function creates a biometric database in flash memory.

#### 5.8.1.1 REQUEST

Identifier value	0x30	
Length value	0x0005 + <L1> + <L2> + ... + <Li>	
Value(Parameters)	Database identifier	1 byte
	Flash type	1 byte
	Person NB in database	2 bytes
	Finger number per person	1 byte
	Additional data Field 1	L1 bytes
	Additional data Field 2	L2 bytes
	...	
Additional data Field i	Li bytes	

#### Database identifier :

Index of the database to create. With MorphoAccess™ 200, it is advised to set this value to 0. With MorphoAccess™ 300, this value can be between 0 and 15.

#### Flash Type :

Must be set to 0.

#### Person NB in database :

Maximum NB of person of the database. With MorphoAccess™ 200, this value cannot exceed 800. With MorphoAccess™ 300, this value must be set to 3000.

#### Finger number per person :

Number of finger per person in the database (1 or 2)

#### Additional data Field :

Add a new field to an existing base.

Identifier value	ID_FIELD	
Length value	0x0002 + <L>	
Value	Field size	2 bytes
	Field name	L bytes

#### Field size :

Define the max size (in bytes) of a record. It cannot exceed 128 bytes.

#### Field name:

String specifying the field name. The size of this string must be less or equal to DDB\_MAX\_FIELD\_NAME\_LEN.

**SAGEM SA**

**Notes :**

- You can create only 8 additional fields.
- With MorphoAccess™ 300, the total length of additional fields must not exceed 216 bytes.
- Only field one is used by the application. See the base structure chapter for more information. Other fields are not used, for futur versions only.

**Notes :**

With MorphoAccess™ 200, in identification mode, the local base used by the MorphoAccess is the base with identifier 0.

### 5.8.1.2REPLY

Identifier value	0x30	
Length value	0x0001	
Value(Parameters)	Request Status	1 byte

**Request Status:**

<b>ILV_OK</b>	The execution of the function succeeded
<b>ILVERR_ERROR</b>	An error occurred during the execution of the function.
<b>ILVERR_BASE_ALREADY_EXISTS</b>	The Database identifier is wrong, this database already exist
<b>ILVERR_NO_SPACE_LEFT</b>	The Database can not be created because there is not enough memory
<b>ILVERR_BADPARAMETER</b>	Wrong number of finger or ILV Format not respected.
<b>ILVERR_CMD_INPROGRESS</b>	Another command is already in progress
<b>ILVERR_OUT_OF_FIELD</b>	The number of the additionnal field is more than 28
<b>ILVERR_FLASH_INVALID</b>	Flash type invalid
<b>ILVERR_ADMIN_EXECUTING</b>	Administrator Mode
<b>ILVERR_BIO_INPROGRESS</b>	An Identification or an authenticate is processing, retry later

## 5.8.2 ERASE BASE ID=0x32

This function erases all records in a local base of the MorphoAccess™.

### 5.8.2.1 REQUEST

Identifier value	0x32	
Length value	0x0001	
Value(Parameters)	Database Identifier	1 byte

#### Database identifier:

With MorphoAccess™ 200, it is the same identifier used in the CREATE DATABASE Command.  
With MorphoAccess™ 300, this value can be between 0 and 15.

### 5.8.2.2 REPLY

Identifier value	0x32	
Length value	0x0001	
Value(Parameters)	Request Status	1 byte

#### Status:

<b>ILV_OK</b>	The execution of the function succeeded
<b>ILVERR_ERROR</b>	An error occurred during the execution of the function.
<b>ILVERR_BASE_NOT_FOUND</b>	The specified base doesn't exist
<b>ILVERR_BADPARAMETER</b>	Wrong number of finger or ILV Format not respected.
<b>ILVERR_CMD_INPROGRESS</b>	Another command is already in progress
<b>ILVERR_BIO_INPROGRESS</b>	Bio Function is processing, retry later

### 5.8.3 ERASE ALL BASE ID=0x33

This function deletes all the local bases in flash support.

#### 5.8.3.1 REQUEST

Identifier value	0x33	
Length value	0x0001	
Value( parameters)	Flash type	1 byte

#### Flash Type :

Must be set to 0.

#### Notes :

For the moment, only the boot sector flash is available, value 0x00 has to be used.

#### 5.8.3.2 REPLY

Identifier value	0x33	
Length value	0x0001	
Value(Parameters)	Request Status	1 byte

#### Status:

<b>ILV_OK</b>	The execution of the function succeeded
<b>ILVERR_ERROR</b>	An error occurred during the execution of the function.
<b>ILVERR_FLASH_INVALID</b>	Flash type invalid
<b>ILVERR_BADPARAMETER</b>	Wrong number of finger or ILV Format not respected.
<b>ILVERR_CMD_INPROGRESS</b>	Another command is already in progress

## 5.8.4 ADD BASE RECORD ID=0x35

This function adds a record to the local database.

### 5.8.4.1 REQUEST

Identifier value	0x35	
Length value	0x0001 + <L <sub>1</sub> > + .. + <L <sub>i</sub> > + <L <sub>T</sub> >	
Value(Parameters)	Database identifier	1 byte
	Biometrics Data 1	L <sub>1</sub> bytes
	...	
	Biometrics Data i	L <sub>i</sub> bytes
	User ID	L bytes
	Additional User Data 1	N <sub>1</sub> bytes
	...	
	Additional User Data k	N <sub>k</sub> bytes
	Time Stamp	L <sub>T</sub> bytes

#### Database identifier:

The database Identifier must be the same as the one used with CREATE DATABASE Command.

#### Biometrics Data:

Data per finger (1 or 2) containing the reference minutiae

Identifier value	ID_PKCOMP	
Length value	0x0000 + <L>	
Value	Minutiae	L bytes

**Minutiae:** The PKs buffer must be 256 char compressed PKs.

#### User ID:

Identifier value	ID_USER_ID	
Length value	0x0000 + <L>	
Value	ID user	L bytes

This field contains personal user data saved in the user ID field.

#### Additional User Data :

Identifier value	ID_PUC_DATA or ID_C_DATA or ID_L_DATA or ID_S_DATA	
Length value	0x0000 + <L>	
Value	Value	L bytes

**Value:** Value of Additional User Data k.

#### Remarks:

- ◆ All the 'Additional User Data field' defined in CREATE DATABASE have to be set in the ADD BASE RECORD command.
- ◆ The number of fingers per person must be the same as defined by CREATE DATABASE.
- ◆ 'User ID' can be retrieved by the *Identify* function under Hit condition.
- ◆ Size and order of additional data fields must be consistent with database structure.

**SAGEM SA**

**Time Stamp:**

Identifier value	ID_TIMESTAMP	
Length value	0x0000 + <L>	
Value	Value	L bytes

One ILV containing a time stamp that will be stored in the database and used to check whether it needs to be updated or not.  
**Value:** it could be a string with **Day, Month, Years, Hours, minutes and Seconds** each encoded with 2 bytes: "DDMMYYHHmmSS".  
 This field is necessary.

**Notes :**

In identification mode, the local base used by the MorphoAccess is the base with identifier 0.

**5.8.4.2REPLY**

Identifier value	0x35	
Length value	0x0006	
Value(Parameters)	Request Status	1 byte
	Base Status	1 byte
	User Database Index	4 bytes

**Request Status:**

<b>ILV_OK</b>	The execution of the function succeeded
<b>ILVERR_ERROR</b>	An error occurred during the execution of the function.
<b>ILVERR_BADPARAMETER</b>	At least one biometric data is required.
<b>ILVERR_INVALID_USER_DATA</b>	The input ILV user data is not valid: bad identifier, wrong size.
<b>ILVERR_INVALID_MINUTIAE</b>	The reference ILV minutiae is not valid: bad identifier, corrupted minutiae.
<b>ILVERR_ALREADY_ENROLLED</b>	The person is already in this database
<b>ILVERR_BIO_INPROGRESS</b>	An Identification or an authenticate is processing, retry later
<b>ILVERR_CMD_INPROGRESS</b>	Another command is already in progress
<b>ILVERR_BASE_NOT_FOUND</b>	The specified base doesn't exist

**Base Status:**

<b>ILVSTS_OK</b>	The enrollment succeed
<b>ILVSTS_DB_FULL</b>	The maximum number of users that can be stored in the database has been reached.
<b>ILVSTS_LATENT</b>	The system detected a latent fingerprint in the input fingerprint.

**User Database Index:**

This is the record index of the person.  
 If the request status differs from ILV\_OK, the User Database Index is not returned.

## 5.8.5 REMOVE BASE RECORD ID=0x36

This function removes a record from the local database.

### 5.8.5.1 REQUEST

Identifier value	0x36	
Length value	0x0001 + <L <sub>1</sub> > + <L <sub>T</sub> >	
Value(Parameters)	Database identifier	1 byte
	User ID	L <sub>1</sub>
	Time Stamp	L <sub>T</sub>

#### Database identifier:

With MorphoAccess™ 200, the index of the base used (See CREATE DATABASE Command).  
 With MorphoAccess™ 300, this value can be between 0 and 15.

#### User ID:

An ILV formatted data that contains the USER ID for the deletion of the record.

Identifier value	ID_USER_ID	
Length value	0x0000 + <L>	
Value	Value	L bytes

#### Time Stamp:

Identifier value	ID_TIMESTAMP	
Length value	0x0000 + <L>	
Value	Value	L bytes

One ILV containing a time stamp that will be stored in the database and used to check whether it needs to be updated or not.

Value: it could be a string with **Day**, **Month**, **Years**, **Hours**, **minutes** and **Seconds** each encoded with 2 bytes: "DDMMYYHHmmSS".

This field is necessary.

#### Notes :

With MorphoAccess™ 200, in identification mode, the local base used by the MorphoAccess is the base with identifier 0.

### 5.8.5.2 REPLY

Identifier value	0x36	
Length value	0x0001	
Value(Parameters)	Request Status	1 byte

#### Request Status:

ILV_OK	The execution of the function succeeded
ILVERR_ERROR	An error occurred during the execution of the function.
ILVERR_BADPARAMETER	The input ILV Identifier Data is not valid: bad Id, bad length.
ILVERR_CMD_INPROGRESS	Another command is already in progress
ILVERR_INVALID_USER_ID	The User ID does not exist in the database



### 5.8.6 GARBAGE COLLECTOR ID=0x0F

With MorphoAccess™ 200, this function executes a garbage collector in the local Database.  
With MorphoAccess™ 300, this function does not exist.

#### 5.8.6.1 REQUEST

Identifier value	0x0F	
Length value	0x0001	
Value(Parameters)	Database Identifier	1 byte

#### 5.8.6.2 REPLY

Identifier value	0x0F	
Length value	0x0001	
Value(Parameters)	Request Status	1 byte

#### Request Status:

<b>ILV_OK</b>	The execution of the function succeeded
<b>ILVERR_ERROR</b>	An error occurred during the execution of the function.
<b>ILVERR_BASE_NOT_FOUND</b>	The specified base doesn't exist
<b>ILVERR_CMD_INPROGRESS</b>	Another command is already in progress
<b>ILVERR_BADPARAMETER</b>	Wrong number of finger or ILV Format not respected.
<b>ILVERR_BIO_INPROGRESS</b>	An Identification or an authenticate is processing, retry later

#### Remarks:

- With MorphoAccess™ 200, this function can only be used on the first local base
- Execution time is significant. It has to be executed while local identification is not used.

### 5.8.7 GET BASE CONFIG ID=0x07

With MorphoAccess™ 200, this function retrieves the configuration of the local database.  
 With MorphoAccess™ 300, this function retrieves the configuration of one of the 16 local bases (0 to 15).

#### 5.8.7.1 REQUEST

Identifier value	0x07	
Length value	0x0001	
Value(Parameters)	Database Identifier	1 byte

#### 5.8.7.2 REPLY

Identifier value	0x07	
Length value	0x0012 + < L <sub>T</sub> >	
Value(Parameters)	Request Status	1 byte
	NB Finger/person	1 byte
	Max record number	4 bytes
	Current Record Number	4 bytes
	Free Record Number	4 bytes
	Fields Number	4 bytes
	ILV Time Stamp	L <sub>T</sub> bytes

#### Request Status:

<b>ILV_OK</b>	The execution of the function succeeded
<b>ILVERR_ERROR</b>	An error occurred during the execution of the function.
<b>ILVERR_BASE_NOT_FOUND</b>	There is no Database corresponding to the Identifier specified in the Request
<b>ILVERR_CMD_INPROGRESS</b>	Another command is already in progress
<b>ILVERR_BADPARAMETER</b>	The input ILV Identifier Data is not valid: bad Id, bad length.
<b>ILVERR_BIO_INPROGRESS</b>	An Identification or an authenticate is processing, retry later

#### NB Finger/person:

Number of fingers saved per person.

#### Max record number:

Maximum number of records in database

#### Current Record Number:

Number of records currently saved in Database

#### Free Record Number:

Number of records currently aviable in Database

#### Fields Number:

Number of supplementary fields used in Database

*SAGEM SA*

**Time Stamp:**

Indicates when the database was modified for the last time. Id must be:

Identifier value	ID_TIMESTAMP
Length value	0x0000 + <L>
Value	Value L bytes

Value: a string with **D**ay, **M**onth, **Y**ears, **H**ours, **m**inutes and **S**econds each encoded with 2 bytes:  
“DDMMYYHHmmSS”

**Notes :**

- Max, Current and Free Record Number are useful to indicate to the host that a GARBAGE\_COLLECTOR command is required.

## 5.8.8 GET ALL BASE CONFIG ID=0x31

This function scans each database and returns the configuration of each created database.

### 5.8.8.1 REQUEST

Identifier value	0x31
Length value	0x0000

### 5.8.8.2 REPLY

Identifier value	0x31	
Length value	0x0002 +< L1>+...+< Li>+ ...+< Ln>	
Value(Parameters)	Request Status	1 byte
	Nb Returned Bases	1 byte
	Base 1 configuration	L1 bytes
	...	
	Base i configuration	Li bytes
	...	
	Base n configuration	Ln bytes

#### Request Status:

<b>ILV_OK</b>	The execution of the function succeeded
<b>ILVERR_BADPARAMETER</b>	Error in ILV request.

If "Request Status" is *ILV\_OK* the following fields are returned:

#### NB Returned Bases:

This number ranged between 0-16 is the number of returned bases.

#### Base i Configuration:

This ILV is returned for each created base.

Identifier value	ID_BASE_CONF	
Length value	0x0016 + L	
Value(Parameters)	Base Nb	1 byte
	Base Valid	1 byte
	NB Finger/person	1 byte
	Max Record Number	4 bytes
	Current Record Number	4 bytes
	Free Record Number	4 bytes
	Fields Number	4 bytes
	ILV Time Stamp	3+L bytes

#### Base Nb:

The ID of the returned base configuration. Ranged in 0-15.

#### Base Valid:

This flag is set to 1 if the base is valid.

**SAGEM SA**

**NB Finger/person:**

Number of fingers saved per person.

**Max record number:**

Maximum number of records in database

**Current Record Number:**

Number of records currently saved in Database

**Free Record Number:**

Number of records currently available in Database

**Fields Number:**

Number of supplementary fields used in Database

**Time Stamp:**

Indicates when the database was modified for the last time. Id must be:

Identifier value	ID_TIMESTAMP
Length value	0x0000 + <L>
Value	Value L bytes

**Value:** a string with **Day**, **Month**, **Years**, **Hours**, **minutes** and **Seconds** each encoded with 2 bytes: "DDMMYYHHmmSS"

**Notes:**

- Max, Current and Free Record Number are useful to indicate to the host that a GARBAGE\_COLLECTOR command is required.

## 5.8.9 BASE DOWNLOAD ID=0x37

This function allows downloading a base to the MorphoAccess™.

*Note: the base must have been previously created with the « Create Database » request. Base parameters must be exactly the same than the base to download (i.e. Database identifier, Number of persons in database, finger number per person, additional fields.)*

### 5.8.9.1 REQUEST

Identifier value	0x37	
Length value	0x0002 + L	
Value(Parameters)	Base Id	1 byte
	Packet Number	1 byte
	Data	L bytes

#### Base ID:

The identifier of the database previously created with the “Create Database” request.

#### Packet Number:

The base can be divided in 127 packets:

0 for packet 1.

...

i for packet i+1

...

The last packet has a specific format:

- The most significant bit must be set to 1.
- Other bits define the packet number.

#### Data:

The packet to download<sup>4</sup>.

### 5.8.9.2 REPLY

Identifier value	0x37	
Length value	0x0001	
Value(Parameters)	Request Status	1 byte
	Error Code	1 byte
	Next Packet Number	1 byte

#### Request Status:

ILV_OK	The execution of the function succeeded
ILVERR_BADPARAMETER	The input ILV Identifier Data is not valid: bad Id, bad length.
ILVERR_CMD_INPROGRESS	Another command is already in progress

<sup>4</sup> For more information about the format of the data to download, please contact your SAGEM SA sales representative.

**SAGEM SA**

**Error Code:**

This value is returned only if Request Status is ILV\_OK.  
Different from SDWN\_NO\_ERROR means that an error occurred during the packet download.  
See chapter 5.17 for detailed description.

**Next Packet Number:**

Number of the next requested packet.  
In case of error the value is the same than "Packet Number" in the request.  
In case of success the value is "Packet Number" + 1, excepted for the last packet where the value is set to 0.

**Notes:**

**Packet number correspondance:**

Request : Packet Number	Reply : Next Packet Number	
	No Error	Error
0	1	0
...	...	...
i	i+1	i
...	...	...
Last + 0x80	0	Last + 0x80

To validate the new database it is necessary to reboot the terminal. The REBOOT can be used.

## 5.8.10 RESET DOWNLOAD ID=0x3A

This function allows resetting the download service.

*When the first « Download Base » request is sent, the system allocates an image of the base in RAM. In case of abnormal download termination resources can be freed with the « Reset Download » command.*

### 5.8.10.1 REQUEST

Identifier value	0x3A
Length value	0x0000

### 5.8.10.2 REPLY

Identifier value	0x3A	
Length value	0x0001	
Value(Parameters)	Request Status	1 byte

#### Request Status:

<b>ILV_OK</b>	The execution of the function succeeded
<b>ILVERR_BADPARAMETER</b>	The input ILV Identifier Data is not valid: bad Id, bad length.
<b>ILVERR_CMD_INPROGRESS</b>	Another command is already in progress



## 5.9 LOG MANAGEMENT FUNCTIONS DESCRIPTIONS (V1.1)

The MorphoAccess™ terminal is able to log all its biometric activities. This function can be used to get information about quality of enrolments, authentications or identifications and also for time and attendance purposes.

Logged Information is:

- Event date and time.
- Type of event (enrolment, identification, authentication).
- ID of the person concerned by the event.
- Biometric information (minutiae quality or matching score).

A log occupies 64 bytes. It is possible to log 3840 events.

### 5.9.1 GET LOG STATUS ID=0x5D

This function returns the state of the log data.

#### 5.9.1.1 REQUEST

Identifier value	0x5D	
Length value	0x0001	
Value(Parameters)	Log Base Id	1 byte

#### Log Base Id:

Defines the number of the base. In V1.1 this parameter is mandatory but ignored.

#### 5.9.1.2 REPLY

Identifier value	0x5D	
Length value	0x000A	
Value (Parameters)	Request status	1 byte
	Log Enabled	1 byte
	Current Line	2 bytes
	Number of Lines	2 bytes
	Line Size	2 bytes
	Base State	1 byte
	Version	1 byte

#### Request status:

ILV_OK	The execution of the function succeeded
ILVERR_ERROR	An error occurred during the execution of the function.
ILVERR_BADPARAMETER	Input parameters error.

#### Log Enabled :

Reflects the configuration. 1 if log are enabled, 0 if they are disabled.

---

*SAGEM SA*

**Current Line :**

The current free line in the base.

**Number of Lines :**

The total number of line in the base.

**Line Size :**

The size of a line.

**Base State :**

0x01 if base is valid. 0xFF if it's full.

**Version :**

The current version of the log service.

Notes: The 4 most significant bit are coding the version number. The 4 least significant bit are coding the revision number. Version 1.1 is encoded as 0x11.

## 5.9.2 GET LOG ID=0x5B

This function allows to download the base.

### 5.9.2.1 REQUEST

Identifier value	0x5B		
Length value	0x0005		
Value (Parameters)	Log Base Id		1 byte
	Offset from current line		2 bytes
	Number of lines		2 bytes

#### Log Base Id :

Defines the number of the base. In V1.1 this parameter is mandatory but ignored.

#### Offset from current line :

Offset from the last written line. Represents the line in the base from where the upload starts.

#### Number of lines:

This parameter is the number of line to upload from « Offset from current line. » This parameter must be ranged between 1 and 512 (V1.1 specification).

### 5.9.2.2 REPLY

Identifier value	0x5B		
Length value	0x0003 + L		
Value (Parameters)	Request status		1 byte
	Number of lines		2 bytes
	Data		L bytes

#### Request status:

ILV_OK	The execution of the function succeeded
ILVERR_ERROR	An error occurred during the execution of the function.
ILVERR_BADPARAMETER	Input parameters error.

#### Number of lines :

Number of line really uploaded. This number is equal to 0 in case of error.

#### Data:

Lines in binary format<sup>5</sup>.

<sup>5</sup> For more information about the format of the log data, please contact your SAGEM SA sales representative.

### 5.9.3 ERASE LOG ID=0x5C

This function erases the log base.

Warning : the function execution does not depend on the "Log Enable" flag.

#### 5.9.3.1 REQUEST

Identifier value	0x5B	
Length value	0x0001	
Value (Parameters)	Log Base Id	1 byte

#### Log Base Id:

Defines the number of the base. In V1.1 this parameter is mandatory but ignored.

#### 5.9.3.2 REPLY

Identifier value	0x5B	
Length value	0x0002	
Value (Parameters)	Request status	1 byte
	Erase status	1 bytes

#### Request status:

ILVSTS_OK	The execution of the function succeeded
ILVERR_ERROR	An error occurred during the execution of the function.
ILVERR_BADPARAMETER	Input parameters error.

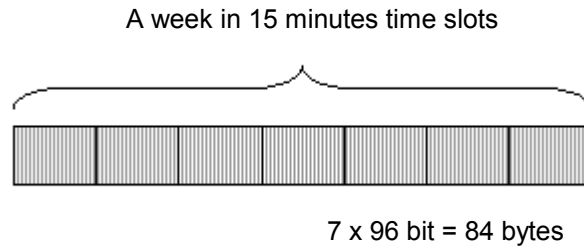
#### Erase status :

ILVSTS_OK	The execution of the function succeeded
ILVSTS_FALSH_ERROR	An error occurred during the access to the base.

## 5.10 TIME MASK FUNCTIONS DESCRIPTION

The MorphoAccess™ terminal is able to grant access to a user according to its time mask.

A user time mask is a bit array architected as:



**Figure 6: time mask structure**

### 5.10.1 SET TIME MASK ID=0x42

This function allows to set a user time mask.

#### 5.10.1.1 REQUEST

Identifier value	0x42	
Length value	0x0058 + <L>	
Value(Parameters)	Base Id	1 byte
	User ID	L bytes
	Time mask	87 bytes

**Base Id:**

ID of the database in which the user is registered.

**User ID:**

An ILV formatted data containing the user ID whose time mask is set.

Identifier value	ID_USER_ID	
Length value	0x0000 + <L>	
Value	Value	L bytes (<25)

**Time mask:**

An ILV formatted data that containing the time mask to set.

Identifier value	ID_TMSK	
Length value	0x0054	
Value	Time mask	84 bytes

## 5.10.1.2 REPLY

Identifier value	0x42		
Length value	0x0002		
Value (Parameters)	Request status		1 byte
	Result		1 byte

### Request status:

<b>ILV_OK</b>	The execution of the function succeeded.
<b>ILVERR_ERROR</b>	An error occurred during the execution of the function.
<b>ILVERR_BADPARAMETER</b>	Input parameters error.
<b>ILVERR_BASE_NOT_FOUND</b>	Used database does not exist.
<b>ILVERR_FIELD_NOT_FOUND</b>	Database cannot handle time masks.
<b>ILVERR_FIELD_INVALID</b>	Database cannot handle time masks.
<b>ILVERR_INVALID_USER_ID</b>	User was not found.

### Result:

<b>ILVSTS_OK</b>	The execution of the function succeeded
<b>ILVSTS_LATENT</b>	An error occurred during the execution of the function.
<b>ILVSTS_DB_FULL</b>	Input parameters error.

## 5.10.2 GET TIME MASK ID=0x43

This function allows to get a user time mask.

### 5.10.2.1 REQUEST

Identifier value	0x43	
Length value	0x0001 + <L>	
Value(Parameters)	Base Id	1 byte
	User ID	L bytes

**Base Id:**

ID of the database in which the user is registered.

**User ID:**

An ILV formatted data containing the user ID whose time mask is set.

Identifier value	ID_USER_ID	
Length value	0x0000 + <L>	
Value	Value	L bytes (<25)

### 5.10.2.2 REPLY

Identifier value	0x43	
Length value	0x0059	
Value (Parameters)	Request status	1 byte
	Result	1 byte
	Time mask	87 bytes

**Request status:**

ILV_OK	The execution of the function succeeded.
ILVERR_ERROR	An error occurred during the execution of the function.
ILVERR_BADPARAMETER	Input parameters error.
ILVERR_BASE_NOT_FOUND	Used database does not exist.
ILVERR_FIELD_NOT_FOUND	Database cannot handle time masks.
ILVERR_FIELD_INVALID	Database cannot handle time masks.
ILVERR_INVALID_USER_ID	User was not found.

**Result:**

ILVSTS_OK	The execution of the function succeeded
ILVSTS_LATENT	An error occurred during the execution of the function.

**Time mask:**

An ILV formatted data that containing the time mask.

Identifier value	ID_TMSK	
Length value	0x0054	
Value	Time mask	84 bytes

## 5.11 MULTILINGUAL FUNCTIONS DESCRIPTION

### 5.11.1 SET USER MESSAGE ID=0x40

This function sets a message in the user message table. Each message is defined by a 19 characters string and an ID. Message IDs are defined chapter 5.18.

This function adds a message in the user language zone if it does not exist.

#### 5.11.1.1 REQUEST

Identifier value	0x40	
Length value	L1+3 + <L2+3> + ...+ <Li+3>+ ... + <Ln+3>	
Value(Parameters)	User Message Declaration 1	L1+3 bytes
	User Message Declaration 2	L2+3 bytes
	...	
	User Message Declaration i	Li+3 bytes
	...	
	User Message Declaration n	Ln+3 bytes

#### User Message Declaration:

This ILV defines the value of a given message.

Identifier value	ID_USER_MESSAGE	
Length value	Li	
Value(Parameters)	User Message Id	1 byte
	Message Text	Li-1 bytes

#### User Message Id:

The Id of the message to edit. If the message has already been set, an error code will be returned. To set the message again, the whole message table must be erased with the “Erase User Message Table” function.

User message Ids are listed in chapter 0: User message ids.

#### Message Text:

This Null terminated string defines the new message text.



## 5.11.1.2 REPLY

Identifier value	0x40	
Length value	0x0002 + <5 bytes> + ... + <5 bytes>	
Value (Parameters)	Request status	1 byte
	<User Message Error 1>	5 bytes
	...	
	<User Message Error j>	5 bytes
	...	
	<User Message Error k>	5 bytes

### Request status:

<b>ILVSTS_OK</b>	The execution of the function succeeded
<b>ILVERR_BADPARAMETER</b>	Input parameters error.

### User Message error:

This ILV is returned only if the update of a given message failed. It usually happens when the message has already been defined.

Identifier value	ID_USER_MESSAGE	
Length value	0x0002	
Value(Parameters)	User Message Id	1 byte
	Error status	1 byte

<b>ILVERR_INVALID_ACCESS</b>	The message has already been defined.
<b>ILVERR_BADPARAMETER</b>	“User Message Id” is out of range.
<b>ILVERR_FLASH_INVALID</b>	An undefined error occurred.

## 5.11.2 ERASE USER TABLE ID=0x41

This function erases the whole user message table.

### 5.11.2.1 REQUEST

Identifier value	0x41
Length value	0x0000

### 5.11.2.2 REPLY

Identifier value	0x41	
Length value	0x0001	
Value (Parameters)	Request status	1 byte

#### Request status:

ILVSTS_OK	The execution of the function succeeded
ILVERR_BADPARAMETER	Input parameters error.
ILVERR_FLASH_INVALID	An undefined error occurred.

## 5.12 PROXY FUNCTIONS DESCRIPTION

### 5.12.1 PRINT SCREEN MESSAGE ID=0x54

This function adds a message on a new screen at the position specified by Position X and Position Y. this

#### 5.12.1.1 REQUEST

Identifier value	0x54		
Length value	0x0003+<L>		
Value(Parameters)	Position X	1 byte	
	Position Y	1 byte	
	Timeout	1 byte	
	Message	L bytes	

**Position X:**

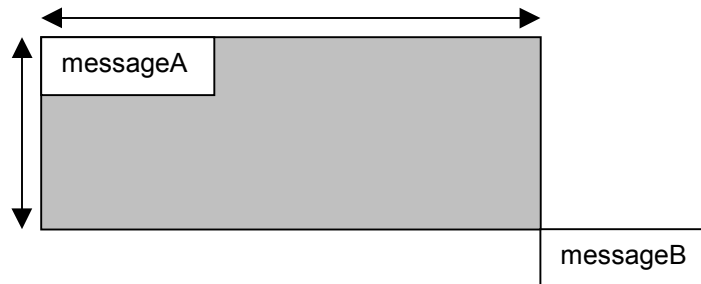
**Position Y:**

These 2 settings allow to configure the position of the string in the screen.

The range of X is 0 to 136

The range of Y is 0 to 34

**Example:**



MessageA is set at (0,0).

MessageB is set at (136,34).

**Timeout:**

Not yet implemented. Message will be displayed until next print screen command call.

**Parameters:**

String to print on the screen (ASCII mode)

The length of the message limited to 90 characters (3 lines of 30 characters).

SAGEM SA

### 5.12.1.2 REPLY

Identifier value	0x54	
Length value	0x0001	
Value(Parameters)	Status	1 byte

**Status:**

ILV_OK	The function completed successfully
ILVERR_ERROR	An error occurred during the execution of the function.

**Notes:**

When a MorphoAccess™ terminal is not in proxy mode, the command is invalid. Therefore its reply is CMD\_INVALID\_REQUEST (ID = 0x50).

## 5.12.2 READ FROM KEYBOARD ID=0x55

This function performs a keyboard read. A new screen is pushed.

### 5.12.2.1 REQUEST

Identifier value	0x55	
Length value	0x0002+<L>	
Value(Parameters)	Secret mode	1 byte
	Time of input	1 byte
	Prompt	L bytes

#### Secret Mode:

Not yet implemented. Must be set to 0x01. By default it is configured in secret mode.

#### Time of input:

Defines the time during which the user is invited to enter data.

#### Prompt:

This string defines the input prompt printed on the screen.

#### Note:

For the moment the maximum length of the input zone is 20 characters.

### 5.12.2.2 REPLY

Identifier value	0x55	
Length value	0x0001+ <L>	
Value(Parameters)	Status	1 byte
	Data	L bytes

#### Status:

ILV_OK	The function completed successfully
ILVERR_ERROR	An error occurred during the execution of the function.

#### Data:

Keyboard data (ASCII mode). This string is not null terminated.

#### Notes:

When a MorphoAccess™ terminal is not in proxy mode, the command is invalid. Therefore its reply is CMD\_INVALID\_REQUEST (ID = 0x50).

### 5.12.3 ACCESS AUTHORIZATION ID=0x56

This function is a way to open access or to refuse it

#### 5.12.3.1 REQUEST

Identifier value	0x56	
Length value	0x0001	
Value(Parameters)	Parameter	1 byte

**Parameter:**

The value 0x01 open the access and lights on green the leds, whereas 0x00 refuse access and lights on red the leds.

#### 5.12.3.2 REPLY

Identifier value	0x56	
Length value	0x0001	
Value(Parameters)	Status	1 byte

**Status:**

ILV_OK	The function completed successfully
ILVERR_ERROR	An error occurred during the execution of the function.

**Notes:**

When a MorphoAccess™ terminal is not in proxy mode, the command is invalid. Therefore its reply is CMD\_INVALID\_REQUEST (ID = 0x50).

## 5.12.4 WIEGAND DATACLOCK READ ID=0x52

This function performs a read on wiegand or dataclock input (according to the MorphoAccess hardware configuration).

### 5.12.4.1 REQUEST

Identifier value	0x52	
Length value	0x0002	
Value(Parameters)	Timeout	2 bytes

After timeout (in seconds), the MorphoAccess replies if no data was read on the Wiegand / Dataclock input. Timeout = 0 means infinite timeout.

### 5.12.4.2 REPLY

Identifier value	0x52	
Length value	0x0002 + <L <sub>1</sub> > + <L <sub>2</sub> >	
Value(Parameters)	Request Status	1 byte
	Matching Result	1 byte
	Facility code	L <sub>1</sub> bytes
	User ID	L <sub>2</sub> bytes

#### Request Status / Matching Result :

ILVERR\_TIMEOUT / ILVSTS\_LATENT : No data has been received during the timeout specified

ILV\_OK / ILVSTS\_WIEGAND\_DATACLOCK\_READ\_ERROR : Invalid data has been received

ILV\_OK / ILVSTS\_OK : Successfull read

#### Facility code :

This ILV is present if the read is successfull and if the MorphoAccess is configured in Wiegand IN mode

Identifier value	ID_FACILITY_CODE	
Length value	0x0001	
Value	Facility code	1 byte

#### User ID :

This ILV is present if the read is successfull

Identifier value	ID_USER_ID	
Length value	0x0000 + <L>	
Value	User Data field	L bytes

#### Notes:

When a MorphoAccess™ terminal is not in proxy mode, the command is invalid. Therefore its reply is CMD\_INVALID\_REQUEST (ID = 0x50).

## 5.13 CONTACTLESS FUNCTIONS DESCRIPTION

It is possible to connect a Deister contactless card reader to port COM1. To enable this feature, the configuration key `/cfg/Maccess/com1/Deister` should be set to 1. In that case, port COM1 is disable for ILV communication and ID transmission. Moreover parameters are fixed to: 9600 bauds, 8 bits of data, no parity, no handshake and 1 stop bit.

### Data organisation according to MIFARE standards.

A card is defined by a unique serial number. The card is divided in 16 sectors. Each sector is divided in 4 blocs. Each bloc contains 16 bytes of data. Data are encoded with two set of key. To be able to read a card the reader should use the same key set. Fourth blocs cannot be read, they are used to store key sets.

Data can be accessed by blocs as follows:

	Bloc 0	Bloc 1	Bloc 2	Bloc 3
Sector 0	<i>Bloc 1</i>	<i>Bloc 2</i>	<i>Bloc 3</i>	
Sector 1	<i>Bloc 4</i>	<i>Bloc 5</i>	<i>Bloc 6</i>	
...				
Sector 15	<i>Bloc 46</i>	<i>Bloc 47</i>	<i>Bloc 48</i>	

### 5.13.1 CONTACTLESS READ ID=0x57

This function is an interface to perform some specific functionalities. If an external product is connected this command allows to read data at the address specified in parameters.

#### 5.13.1.1 REQUEST

Identifier value	0x57	
Length value	0x0004	
Value(Parameters)	ID function	1 byte
	Parameters	3 bytes

#### ID function:

ID of the function to perform.

#### Parameters:

Parameters associated with each function.

An address of card parameter has this structure:

<Address xxx> is composed of 3 bytes: <B><N><C>

<B>: First block number to read (1 to 48)

<N>: Number of blocks to read (1 to 48)

<C>: Key number for the read (1or 2)

Blocks are numbered in a absolute way, 1 for block 0 sector 0, then 3 blocks for each sectors.

#### Notes:

For the moment there are two specific funtions implemented in this way

- Read serial number of card reader ( ID = AC\_READ\_SERIAL\_NUMBER ). Even if it is not necessary to have an address to read the serial number, the field has to be set.
- Read data in a card ( ID = AC\_READ\_DATA\_CARD)



### 5.13.1.2 REPLY

Identifier value	0x57	
Length value	0x0002+<L>	
Value(Parameters)	ID function	1 byte
	Status	1 byte
	Data	L bytes

Each function has a specific data and status associated with.

#### Reply Status:

<b>STATUS_CARD_PRESENT</b>	Card present
<b>STATUS_CARD_NO_PRESENT</b>	Card no present
<b>STATUS_ERROR</b>	Error read
<b>STATUS_SUCCESS</b>	Successful read
<b>STATUS_WRONG_CARD</b>	Card has been changed
<b>STATUS_AUTHENT_ERROR</b>	Wrong keys
<b>STATUS_ERROR</b>	Error read
<b>STATUS_CARD_CLOSED</b>	Card not opened
<b>ILVERR_BADPARAMETER</b>	Bad input parameters

#### Data:

##### ID function:

ID function	Data
AC_READ_SERIAL_NUMBER	Serial number
AC_READ_DATA_CARD	Datas read from the card

#### Notes:

When the Deister contactless card reader option is not activated on the MorphoAccess™ terminal, the command is invalid. Therefore its reply is CMD\_INVALID\_REQUEST (ID = 0x50).

## 5.13.2 CONTACTLESS AUTHENTIC ID=0x59

This function is an interface to perform an authenticate matching Pk on a badge and on fingerprint.

Identifier value	0x59	
Length value	0x0005+< L <sub>1</sub> > + ... + < L <sub>i</sub> > + <K> + <J>	
Value(Parameters)	Timeout	2 bytes
	Matching Threshold	2 bytes
	Acquisition Quality Threshold	1 byte
	Address Biometric Data 1	L <sub>1</sub> bytes
	...	
	Address Biometric Data i	L <sub>i</sub> bytes
	Address of data for hashing	K bytes
	Address of signature	J bytes

### 5.13.2.1 REQUEST

#### Timeout, Matching threshold and Acquisition Quality Threshold :

parameters take the same values that the ones in command AUTHENTICATE (ID = 0x20) .  
 Acquisition Quality Threshold are not yet implemented.

#### Address Biometric Data 1 :

#### Address biometric data i :

This parameters give addresses containing the reference minutiae of fingerprint. The reference minutiae are compressed with algorithm PKCOMP ( with MorphoKit for example )

#### Notes:

It is possible to send PKs with ID\_ADDRDATA, the verification of signature will be OK, but it is very important to send the Pks with ID\_ADDRPK otherwise the authentication will be impossible, same thing with signature. For the moment this ILV must not be superior to 500 bytes

Identifier value	ID_ADDRPK	
Length value	0x0003	
Value(Parameters)	Address	3 bytes

#### Address :

An address of card parameter has this structure:

<Address xxx> is composed of 3 bytes: <B><N><C>

<B>: First block number to read (1 to 48)

<N>: Number of blocks to read (1 to 48)

<C>: Key number for the read (1 or 2)

Blocks are numbered in a absolute way, 1 for block 0 sector 0, then 3 blocks for each sectors.

**SAGEM SA**

**Address of data for hashing :**

This packet give address containing the data used for the buffer hashing.

Identifier value	ID_ADDRDATA		
Length value	0x0004		
Value(Parameters)	Numbers of bytes	1 byte	
	Address	3 bytes	

**number of bytes :**

Is for the number of bytes to take in consideration. It can be set from 1 to a maximum of 64 bytes

**Adress :**

An address of card parameter has this structure:

<Address xxx> is composed of 3 bytes: <B><N><C>

<B>: First block number to read (1 to 48)

<N>: Number of blocks to read (1 to 48)

<C>: Key number for the read (1or 2)

Blocks are numbered in a absolute way, 1 for block 0 sector 0, then 3 blocks for each sectors.

**Address of signature :**

If a verification of signature is performed (for the moment with algorithm DSA) a buffer has to be given to make a hash. This parameter aims at configuring this buffer:

Each ILV packet contains an address which can be a Pk, Data or the signature. ILV packets have to be send in the same order that the buffer signed in order to allow the build of the buffer by the application. The signature has to be send the last one.

**Notes :**

This packet is ignored if the verification of Pk by a signature is not performed.

Identifier value	ID_ADDRSIGN		
Length value	0x0003		
Value(Parameters)	Address	3 bytes	

**Adress :**

An address of card parameter has this structure:

<Address xxx> is composed of 3 bytes: <B><N><C>

<B>: First block number to read (1 to 48)

<N>: Number of blocks to read (1 to 48)

<C>: Key number for the read (1or 2)

Blocks are numbered in a absolute way, 1 for block 0 sector 0, then 3 blocks for each sectors.

**For example :**

If the buffer signed is made with: fingerprint1,data1,data2, fingerprint2, 5 ILV will be send:

First one the ILV with <address fingerprint1>, then ILV with <address data1>,then <address data2>,<address fingerprint2>, and finally <address signature>

## 5.13.2.2 REPLY

Identifier value	0x59	
Length value	0x0002	
Value(Parameters)	Status	1 byte
	Matching result	1 byte

### Status:

<b>ILV_OK</b>	The function completed successfully
<b>ILVERR_ERROR</b>	An error occurred during the execution of the function.
<b>ILVERR_BADPARAMETER</b>	The matching threshold value or timeout value are not in the range; or there is no input biometrics data
<b>ILVERR_INVALID_MINUTIAE</b>	The reference ILV minutiae is not valid: bad identifier, corrupted minutiae.
<b>ILVERR_TIMEOUT</b>	The finger detection timeout has expired
<b>ILVERR_FINGER_REMOVED</b>	The finger has been removed during capture, not used by MorphoAccess
<b>ILVERR_CMD_INPROGRESS</b>	Another command is already in progress
<b>ILVERR_CARD_PROBLEM</b>	A card problem occurs
<b>ILVERR_BAD_SIGNATURE</b>	Error in check of signature
<b>ILVERR_WAITING_BIO</b>	An Identification or an authenticate is processing, retry later

### Matching Result:

This is the result of the matching between the reference minutiae and the calculated one.

<b>ILVSTS_HIT</b>	The comparison succeed
<b>ILVSTS_NO_HIT</b>	It is not the same finger
<b>ILVSTS_LATENT</b>	The system detected a latent fingerprint in the input fingerprint.

### Note :

If the request status differs from ILV\_OK, the matching result is not returned.

### Notes:

When the Deister contactless card reader option is not activated on the MorphoAccess™ terminal, the command is invalid. Therefore its reply is CMD\_INVALID\_REQUEST (ID = 0x50).

### 5.13.3 CRYPTO WRITEID=0x58

This command is a way to send MIFARE keys to the reader card to decrypt data of the badge sector by sector

#### 5.13.3.1 REQUEST

Identifier value	0x58		
Length value	0x000E		
Value(Parameters)	Key number		1 byte
	Sector number		1 byte
	Keys sector		12 bytes

**Key number:**

This parameter takes two values 1 or 2.

**Sector number:**

This value can be set from 0 to 15.

**Keys sector:**

This value is divided in 6 bytes for keys A and 6 bytes for keys B.

#### 5.13.3.2 REPLY

Identifier value	0x58		
Length value	0x0001		
Value(Parameters)	Status		1 byte

**Status:**

ILV_OK	The function completed successfully
ILVERR_BADPARAMETER	Problem with input parameters
ILVERR_ERROR	An error occurred during the execution of the function.

**Notes:**

When the Deister contactless card reader option is not activated on the MorphoAccess™ terminal, the command is invalid. Therefore its reply is CMD\_INVALID\_REQUEST (ID = 0x50).

## 5.14 FUNCTIONS TABLE

ID	Command
0x01	Set configuration
0x02	<i>Reserved</i>
0x03	Get version
0x04	Reboot
0x06	Get configuration
0x07	Get base configuration
0x08	Ping
0x09	<i>Reserved</i>
0x0A	Default init
0x0F	Garbage collector
0x20	Authentication
0x21	Enrol
0x22	Identify
0x30	Create base
0x31	Get all base config.
0x32	Erase base
0x33	Erase flash
0x35	Add base record
0x36	Remove base record
0x37	Download database
0x3A	Reset database download.
0x40	Set user message
0x41	Erase user message table
0x42	Set time mask
0x43	Get time mask
0x50	<b>Invalid Request</b>
0x52	Wiegand/Dataclock read
0x53	Set Public Key
0x54	Print screen message
0x55	Keyboard read
0x56	Access autorisation
0x57	Contactless read
0x58	Cryptowrite
0x59	Contactless authentication
0x5A	<i>Not Yet Implemented</i>
0x5B	Get log
0x5C	Erase log
0x5D	Get log status
0xAA	<i>Reserved</i>
0xFE	<i>Reserved</i>
0xFF	<i>Internal use only</i>

## 5.15 ERROR CODES VALUE

Error codes macro	Definition	Value
ILVERR_WAITING_BIO	A Bio Function is processing, the MorphoAccess™ will answer later	1
ILV_OK	Successful result	0
ILVERR_ERROR	An error occurred	-1
ILVERR_BADPARAMETER	Input parameters are not valid	-2
ILVERR_INVALID_MINUTIAE	The minutiae is not valid	-3
ILVERR_INVALID_USER_ID	The User ID does not exist in the database	-4
ILVERR_INVALID_USER_DATA	The user data is not valid	-5
ILVERR_TIMEOUT	No response after defined time.	-6
ILVERR_INVALID_ID_PROTOCOL	The protocol used is not valid	-7
ILVERR_ALREADY_ENROLLED	The person is already in this base	-8
ILVERR_BASE_NOT_FOUND	The specified base doesn't exists	-9
ILVERR_BASE_ALREADY_EXISTS	The specified base already exists	-10
ILVERR_BIO_INPROGRESS	Bio Function in progress	-11
ILVERR_CMD_INPROGRESS	A command is already in progress	-12
ILVERR_FLASH_INVALID	Flash type invalid	-13
ILVERR_NO_SPACE_LEFT	Not Enough memory for the creation of a database	-14
ILVERR_ADMIN_EXECUTING	Administrator Mode in use.	-15
ILVERR_BAD_SIGNATURE	The signature check failed.	-16
ILVERR_CARD_PROBLEM	An error occured while reading the card.	-17
ILVERR_ERROR_SL	An error occurred during the execution of Serial Link Configuration.	-18
ILVERR_ERROR_WIEGAND	An error occurred during the execution of the output Wiegand/Dataclock Configuration.	-19
ILVERR_ERROR_NETWORK	An error occurred during the execution of Network Configuration.	-20
ILVERR_OUT_OF_FIELD	The number of the additionnal field is more than 8 or cumulated length of additional user data is more than 216.	-21
ILVERR_INVALID_ACCESS	Access cannot be be granted because time does not agree with time zone	-22
ILVERR_FIELD_NOT_FOUND	Field does not exist	-23
ILVERR_FIELD_INVALID	Field size or field name is invalid	-24
ILV_NOT_IMPLEMENTED	The request is not yet implemented	-99

## 5.16 STATUS CODES VALUE

Status codes macro	Definition	Value
ILVSTS_OK	Successful	0
ILVSTS_HIT	Authentication or Identification succeeded	1
ILVSTS_NO_HIT	Authentication or Identification failed	2
ILVSTS_LATENT	Security Protection Triggered	3
ILVSTS_DB_FULL	The database is full.	4
ILVSTS_DB_EMPTY	The database is empty.	5
ILVSTS_BAD_QUALITY	Bad finger and/or enroll quality	6
ILVSTS_DB_OK	The database is right.	7
ILVSTS_ACTIVATED	The Terminal is activated.	8
ILVSTS_NOTACTIVATED	The Terminal is not activated.	9
ILVSTS_DB_KO	The last 'AddRawDatabase' command is unsuccessful	10
ILVSTS_WIEGAND_DATACLOCK_READ_ERROR	Read error on Wiegand / Dataclock input	11



## 5.17 DATABASE DOWNLOAD ERROR CODES

Error Code :	Value	
SDWN_NO_ERROR (any packet)	0x00	The packet has been successfully downloaded.
SDWN_ERR_BAD_PACKET_NUMBER (any packet)	0x11	Packet number does not correspond to the one awaited.
SDWN_ERR_BAD_HEADER (first packet)	0x12	Base header does not correspond with the base created with "Create Database".
SDWN_ERR_NO_BASE (first packet)	0x13	No base has been created.
SDWN_ERR_BASE_NAME (first packet)	0x14	Base name does not correspond with the base created with "Create Database".
SDWN_ERR_NUMBER_PERS (first packet)	0x15	Number of person does not correspond with the base created with "Create Database".
SDWN_ERR_MEMORY (first packet)	0x16	Memory error. No enough resource to download a base.
SDWN_ERR_OVERLOAD (any packet)	0x17	Too many data have been sent. Base size does not correspond with the base created with "Create Database".
SDWN_ERR_TOO_SMALL (last packet)	0x18	No enough data have been sent. Base size does not correspond with the base created with "Create Database".
SDWN_ERR_WRITING (last packet)	0x19	An error occurred during the base inscription in flash.
SDWN_ERR_GENERIC (any packet)	0x20	Generic error.

## 5.18 USER MESSAGE IDS

User message ID	Corresponding message in the English table	Value
SINTL_MESSAGE_IDENT_1	Place your finger	0
SINTL_MESSAGE_IDENT_2	for Identification	1
SINTL_MESSAGE_IDENT_3	Please	2
SINTL_MESSAGE_AUTHENT_1	Place your finger	3
SINTL_MESSAGE_AUTHENT_2	for Authentication	4
SINTL_MESSAGE_AUTHENT_3	Please	5
SINTL_MESSAGE_PASS_YOUR_BADGE_1	Pass your badge	6
SINTL_MESSAGE_PASS_YOUR_BADGE_2	for Authentication	7
SINTL_MESSAGE_PASS_YOUR_BADGE_3	Please	8
SINTL_MESSAGE_AUTHENT_AFTER_BADGE_READ_1	Authentication of	9
SINTL_MESSAGE_AUTHENT_AFTER_BADGE_READ_3	Place your finger	10
SINTL_MESSAGE_PROXY	Proxy mode	11
SINTL_MESSAGE_USER_NOT_FOUND_1	User not found in	12
SINTL_MESSAGE_USER_NOT_FOUND_2	current database	13
SINTL_MESSAGE_USER_NOT_FOUND_3		14
SINTL_MESSAGE_WELCOME	Welcome	15
SINTL_MESSAGE_IDENT_OK	IDENTIFIED	16
SINTL_MESSAGE_AUTHENT_OK	AUTHENTICATED	17
SINTL_MESSAGE_IDENT_NOT_OK_1	NOT IDENTIFIED	18
SINTL_MESSAGE_IDENT_NOT_OK_2		19
SINTL_MESSAGE_AUTHENT_NOT_OK_1	NOT AUTHENTICATED	20
SINTL_MESSAGE_AUTHENT_NOT_OK_2		21
SINTL_MESSAGE_PLEASE_RETRY	Please retry	22
SINTL_MESSAGE_PLEASE_WAIT	Please wait	23
SINTL_MESSAGE_IDENT_TIMEOUT_1	IDENTIFICATION	24
SINTL_MESSAGE_IDENT_TIMEOUT_2	TIMEOUT	25
SINTL_MESSAGE_AUTHENT_TIMEOUT_1	AUTHENTICATION	26
SINTL_MESSAGE_AUTHENT_TIMEOUT_2	TIMEOUT	27
SINTL_MESSAGE_TIMEOUT	TIMEOUT	28
SINTL_MESSAGE_IDENT_ERROR_1	IDENTIFICATION	29
SINTL_MESSAGE_IDENT_ERROR_2	ERROR	30
SINTL_MESSAGE_AUTHENT_ERROR_1	AUTHENTICATION	31
SINTL_MESSAGE_AUTHENT_ERROR_2	ERROR	32
SINTL_MESSAGE_NO_LICENCE	NO LICENCE	33
SINTL_MESSAGE_NO_DATABASE	No Database	34
SINTL_MESSAGE_EMPTY_BASE	Empty Database	35
SINTL_MESSAGE_NO_BASE_MA200_1	Warning	36
SINTL_MESSAGE_NO_BASE_MA200_2	Base does not exist	37
SINTL_MESSAGE_NO_BASE_MA200_3	Waiting for data	38
SINTL_MESSAGE_NO_BASE_MA300_1	Base does not exist	39
SINTL_MESSAGE_NO_BASE_MA300_2	Please select	40
SINTL_MESSAGE_NO_BASE_MA300_3	Another one	41

**SAGEM SA**

SINTL_MESSAGE_CONTACT_ADMINISTRATOR_1	Please contact	42
SINTL_MESSAGE_CONTACT_ADMINISTRATOR_2	Administrator	43
SINTL_MESSAGE_BASE_MANAGEMENT_1	Base management	44
SINTL_MESSAGE_BASE_MANAGEMENT_2		45
SINTL_MESSAGE_LOG_MANAGEMENT_1	Log management	46
SINTL_MESSAGE_LOG_MANAGEMENT_2		47
SINTL_MESSAGE_ANALYSE_1		48
SINTL_MESSAGE_ANALYSE_2	Remove finger	49
SINTL_MESSAGE_ANALYSE_3	Analysing ...	50
SINTL_MESSAGE_MOVE_UP	Move up	51
SINTL_MESSAGE_MOVE_DOWN	Move down	52
SINTL_MESSAGE_MOVE_LEFT	Move left	53
SINTL_MESSAGE_MOVE_RIGHT	Move right	54
SINTL_MESSAGE_PRESS_HARDER	Press harder	55
SINTL_MESSAGE_BAD_FINGER	Bad finger	56
SINTL_MESSAGE_OR_BAD_FINGER	or bad finger	57
SINTL_MESSAGE_ALREADY_ENROLLED	Already enrolled	58
SINTL_MESSAGE_ENROLLMENT	Enrollment	59
SINTL_MESSAGE_PLACE_FIRST_FINGER	Place 1st finger	60
SINTL_MESSAGE_PLACE_SECOND_FINGER	Place 2nd finger	61
SINTL_MESSAGE_AGAIN	again	62
SINTL_MESSAGE_ENROLL_OK_1	Person now	63
SINTL_MESSAGE_ENROLL_OK_2	Enrolled	64
SINTL_MESSAGE_ENROLL_ERROR_1	Enrollment	65
SINTL_MESSAGE_ENROLL_ERROR_2	Error	66
SINTL_MESSAGE_USER_ALREADY_ENROLLED_1	USER	67
SINTL_MESSAGE_USER_ALREADY_ENROLLED_2	ALREADY ENROLLED	68
SINTL_MESSAGE_ADD_RECORD_ERROR_1	ADDRECORD ERROR	69
SINTL_MESSAGE_ADD_RECORD_ERROR_2		70
SINTL_MESSAGE_DEL_RECORD_IN_PROGRESS_1	Del record	71
SINTL_MESSAGE_DEL_RECORD_IN_PROGRESS_2	in	72
SINTL_MESSAGE_DEL_RECORD_IN_PROGRESS_3	Progress	73
SINTL_MESSAGE_DEL_RECORD_OK_1	Person ID :	74
SINTL_MESSAGE_DEL_RECORD_OK_3	is deleted	75
SINTL_MESSAGE_TIME_MASK_ERROR_1	Time mask	76
SINTL_MESSAGE_TIME_MASK_ERROR_2	Error	77
SINTL_MESSAGE_TIME_MASK_ERROR_3		78
SINTL_MESSAGE_ACCESS_DENIED_1	Access Denied	79
SINTL_MESSAGE_ACCESS_DENIED_2		81
SINTL_MESSAGE_ACCESS_DENIED_3		82
SINTL_MESSAGE_TIME_MASK_FIELD_ABSENT_1	No time mask field	83
SINTL_MESSAGE_TIME_MASK_FIELD_ABSENT_2		84
SINTL_MESSAGE_TIME_MASK_FIELD_INVALID_1	Time mask field	85
SINTL_MESSAGE_TIME_MASK_FIELD_INVALID_2	Invalid	86
SINTL_MESSAGE_SECURE_MODE	Secure Mode :	87
SINTL_MESSAGE_SECURE_KEYS	Keys	88
SINTL_MESSAGE_SECURE_UPDATED	Updated	89
SINTL_MESSAGE_SECURE_WARNING	Warning !	90
SINTL_MESSAGE_SECURE_ERROR	Error !	91

**SAGEM SA**

SINTL_MESSAGE_SECURE_REQUEST	Bad Request.	92
SINTL_MESSAGE_SECURE_NO_KEY	No Key.	93
SINTL_MESSAGE_SECURE_BAD_SIGNATURE	Bad Signature.	94
SINTL_MESSAGE_MPILV_1	Remote management.	95
SINTL_MESSAGE_MPILV_2	Please Wait.	96
SINTL_MESSAGE_IDLE_1	Press any key	97
SINTL_MESSAGE_IDLE_2	To start ident.	98
SINTL_MESSAGE_IDLE_3	Please	99

## 5.19 CONSTANTS

Constant	Value
DDB_MAX_FIELD_NAME_LEN	6

## 5.20 ID VALUE

Constant ID macro	Definition	Value
ID_PKCOMP	Identifies a PKCOMP ILV (minutiae)	2
ID_PKMAT	Identifies a PKMAT ILV (minutiae)	3
ID_USER_ID	Identifies a user data ILV	4
ID_USER_DATA	Identifies a user ID ILV	5
ID_COM1	Identifies the first serial link of the Terminal	6
ID_COM2	Identifies the second serial link of the Terminal	7
ID_COM3	Identifies the third serial link of the Terminal	8
ID_WIEGAND	Identifies the Wiegand Driver of the MorphoAccess	9
ID_TCPIP_PC	Identifies the TCP/IP configuration of the MorphoAccess	10
ID_FIELD	Identifies the name of a field in database	15
ID_FIELD_SIZE	Identifies the size of a field in database	16
ID_TIMESTAMP	Identifies a time stamp for database maintenance	17
ID_RELAY	Identifies a relay configuration	18
ID_DATE	Identifies a date configuration	19
ID_PUC_DATA	Identifies that the value is a string	20
ID_C_DATA	Identifies that the value is a character	21
ID_S_DATA	Identifies that the value is a short	22
ID_L_DATA	Identifies that the value is a long	23

*SAGEM SA*

ID_DATACLOCK	Identifies the DataClock driver of the MorphoAccess	27
ID_WIEGAND_OPTION	Identifies the DataClock driver options	28
ID_APPLI_OPTION	Identifies the application configuration	29
ID_LANGUAGE	Identifies the language service configuration	30
ID_BASE_CONF	Identifies the base configuration structure.	31
ID_USER_MESSAGE	Identifies a user message declaration	32
ID_FACILITY_CODE	Identifies a facility code ILV	33
ID_TMSK	Identifies a time mask	36
ID_GUI	Identifies user interface configuration.	37

## 5.21 CONTACTLESS CODE VALUES

Constant ID macro	Definition	Value
STATUS_CARD_PRESENT	A card is present, ready to be read	0
STATUS_CARD_NO_PRESENT	No card present	11
STATUS_SUCCESS	Read performed successfully	0
STATUS_AUTHENT_ERROR	Set of keys to decrypt datas is not OK	12
STATUS_WRONG_CARD	The serial number of the card is not OK	13
STATUS_CARD_CLOSED	The card to be read is not opened	14
STATUS_ERROR	Error while reading	15
AC_READ_SERIAL_NUMBER	Serial number of card reader	27
AC_READ_DATA_CARD	Read data in a card	28

## 5.22 CONFIGURATION TABLE

### 5.22.1 Serial Link Configuration

Parameters	Value	Comment
<b>Speed</b>	0x00	1200 bauds
	0x01	2400 bauds
	0x02	4800 bauds
	0x03	9600 bauds
	0x04	19200 bauds
	0x05	28800 bauds
	0x06	38400 bauds
	0x07	57600 bauds
	0x08	115200 bauds
<b>Databits</b>	0x05	5 databits
	0x06	6 databits
	0x07	7 databits
	0x08	8 databits
<b>Stopbits</b>	0x01	1 stopbit
	0x02	2 stopbits
<b>Parity</b>	0x00	No
	0x01	Odd
	0x02	Even
<b>Flow Ctrl Type</b>	0x00	No
	0x01	RTS / CTS (for COM2 only)
	0x02	XON / XOFF
<b>Send</b>	0x00	The COM1 or COM2 output is not activate <sup>6</sup>
	0x01	The COM1 or COM2 output is activate
<b>Format</b>	0x00	ID is sent in ASCII format
	0x01	ID is sent in Binary format
	0x02	ID is sent in BCD format <sup>8</sup>

<sup>6</sup> See the Terminal block board in Installation guide document.

<sup>7</sup> See the Terminal block board in Installation guide document.

<sup>8</sup> Binary Code Decimal



## 5.22.2 Wiegand/Dataclock Configuration

parameters	Value	Comment
<b>Send</b>	0x00	Deactivate the Wiegand/DataClock <sup>9</sup> output
	0x01	Activate the Wiegand/DataClock output
<b>Format</b>	0x00	Fixed Value

## 5.22.3 Setting Up Relay Parameters

Parameters	Value	Comment
<b>Activate</b>	0x00	Relay not active
	0x02	Relay active

<sup>9</sup> See the Terminal block board in Installation guide document.