TSG RAN Meeting #26 **RP-040445**
Vouliagmeni Athens, Greece, 08 - 10 December 2004

Title **CRs (Rel-6 Category F) to TS 25.463**
Source **TSG RAN WG3**
Agenda Item **8.10**

RAN3 Tdoc	Spec	curr. Vers.	new Vers.	CR	Rev	Cat	Rel	Title	Work item
R3-041423	25.463	6.0.0	6.1.0	2	-	F	Rel-6	Clarification of allowed tilt operation during self test	RANimp-TiltAnt
R3-041424	25.463	6.0.0	6.1.0	3	-	F	Rel-6	State Model for RET device	RANimp-TiltAnt
R3-041451	25.463	6.0.0	6.1.0	4	-	F	Rel-6	Corrections and editorial changes to 25.463 according to RAN3#44	RANimp-TiltAnt
R3-041531	25.463	6.0.0	6.1.0	17	-	F	Rel-6	Definition of response time in the appication layer.	RANimp-TiltAnt
R3-041626	25.463	6.0.0	6.1.0	1	2	F	Rel-6	Reduction of risk of accidentional erasure of Ret application SW	RANimp-TiltAnt
R3-041642	25.463	6.0.0	6.1.0	5	1	F	Rel-6	Antenna Send Configuration Data procedure missing	RANimp-TiltAnt
R3-041643	25.463	6.0.0	6.1.0	7	1	F	Rel-6	Introduction of Software Download State model	RANimp-TiltAnt
R3-041671	25.463	6.0.0	6.1.0	8	3	F	Rel-6	Alarm handling clarification	RANimp-TiltAnt
R3-041673	25.463	6.0.0	6.1.0	9	2	F	Rel-6	RET DC power consumption clarification	RANimp-TiltAnt
R3-041674	25.463	6.0.0	6.1.0	10	2	F	Rel-6	Response message format clarification	RANimp-TiltAnt
R3-041675	25.463	6.0.0	6.1.0	16	2	F	Rel-6	Maximum data payload size in elementary procedures.	RANimp-TiltAnt
R3-041676	25.463	6.0.0	6.1.0	12	2	F	Rel-6	Return code clean-up and clarification	RANimp-TiltAnt
R3-041677	25.463	6.0.0	6.1.0	15	2	F	Rel-6	Clarification on the intention of the elementary procedures ReadUserData and WriteUserData	RANimp-TiltAnt
R3-041679	25.463	6.0.0	6.1.0	18	2	F	Rel-6	Redefinition of the Elementary Procedures GetDeviceData and SetDeviceData	RANimp-TiltAnt

3GPP TSG-RAN3 Meeting #45 ***Tdoc ⌘ R3-041674***
Shin-Yokohama, Japan, 15th - 19th November 2004

CR-Form-v7.1

CHANGE REQUEST

⌘ **25.463** **CR** **10** ⌘ **rev** **2** ⌘ Current version: **6.0.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ ☐ ME ☐ Radio Access Network ☒ Core Network ☐

Title:	⌘	Response message format clarification
Source:	⌘	RAN3
Work item code:	⌘	RANimp-TiltAnt
Date:	⌘	18/11/2004
Category:	⌘	**F**
Release:	⌘	Rel-6

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:
Ph2 (GSM Phase 2)
R96 (Release 1996)
R97 (Release 1997)
R98 (Release 1998)
R99 (Release 1999)
Rel-4 (Release 4)
Rel-5 (Release 5)
Rel-6 (Release 6)
Rel-7 (Release 7)

Reason for change:	⌘	Unclear specification
Summary of change:	⌘	Response message format introduced. Data types introduced in definitions. Data types in response messages clarified. The number of return codes in fail responses is fixed.
Consequences if not approved:	⌘	Format and data types in response message remain undefined or unclear.
Clauses affected:	⌘	3.1, 5, 6.4 to 7

	Y	N		
Other specs affected: ⌘		X	Other core specifications	⌘
		X	Test specifications	
		X	O&M Specifications	

Other comments:	⌘	

How to create CRs using this form:
Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm.
Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.
2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be

downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

3.1 Definitions

For the purposes of the present document, the following terms and definitions apply.

ASCII character: A character forming part of the International Reference Version of the 7-bit character set defined in ISO/IEC 646:1991

Calibrate: Exercise the antenna drive unit over its entire range of travel to ensure fault-free operation and synchronise the measured and actual beam tilt of the antenna

Configuration data: A stored table or function defining the relationship between the physical position of the drive and electrical beam-tilt

Data type: A definition determining the value range and interpretation of a series of octets. The following specified data types are used in this TS:

Name:	Definition:
AlarmCode	1 octet unsigned enumerated code. All AlarmCode values are listed in annex A of this TS
FieldNumber	1 octet unsigned enumerated code All field number values are listed in annex B of this TS
ProcedureCode	1 octet unsigned enumerated code.
ReturnCode	1 octet unsigned enumerated code. All ReturnCode values are listed in annex A of this TS
TextString	Octets with integer values in the range of 32 to 126 to be interpreted as ASCII characters.

Device type: See section 4.7 in [3].

Elementary Procedure: The RETAP protocol consists of Elementary Procedures (EPs). An Elementary Procedure is a unit of interaction between the primary device (Node B) and the secondary devices (RET devices).

An EP consists of an initiating message and possibly a response message.

Two kinds of EPs are used:

- **Class 1**: Elementary Procedures with response (success or failure).
- **Class 2**: Elementary Procedures without response.

For **Class 1** EPs, the types of responses can be as follows:

Successful

- A signalling message explicitly indicates that the elementary procedure has been successfully completed with the receipt of the response.

Unsuccessful

- A signalling message explicitly indicates that the EP failed.

Class 2 EPs are considered always successful.

Little-endian: The order of transmission in which the least-significant octets of a multi-octet representation of a number are transmitted first. Little endian only applies to binary integer representations.

Procedure code: A code identifying an elementary procedure.

Return code: ~~A 1 octet enumerated .response message to an initiating message.~~A code which defines information about the outcome of an elementary procedure execution.

Tilt (also downtilt, tilt angle, beamtilt): The elevation angle between the direction orthogonal to the antenna element axis and the maximum of its main beam in the elevation plane. A positive electrical tilt angle means that the antenna beam is directed below the direction orthogonal to the antenna axis. An antenna has separate values for electrical and mechanical tilt. The mechanical tilt is fixed by the geometry of the installation. In this TS the tilt referred to is always the electrical tilt unless otherwise stated

Tilt value: A signed integer used in elementary procedures to define the electrical tilt setting of the antenna. The tilt value is 10 times the antenna electrical tilt angle.

----------NEXT CHANGED SECTION----------

5 Services expected from signalling transport

RETAP requires an assured in-sequence delivery service from the signalling transport and notification if the assured in-sequence delivery service is no longer available.

5.1 Elementary procedure format

Layer 2 provides a full-duplex link for the transmission of RETAP messages.

There are two types of RETAP elementary procedures:

Class 1: Initiating messages are sent either from the primary to a secondary device, or from a secondary to the primary device, in order to initiate some action within the receiving device. The other device sends a response message completing the procedure.

Class 2: Initiating messages are sent either from the primary to a secondary device, or from a secondary to the primary device. No response message is expected.

All RETAP messages use the same basic format:

Table 5.1.1: Basic format for all RETAP messages

Elementary Procedure	Number of data octets	Data
1 octet	2 octets	78 octets

NOTE1: The default frame length of 78 octets is used unless another frame length is negotiated during the Address Assignment procedure (see [3]).

NOTE2: Response messages have the same basic format as initiating messages. The elementary procedure code shall be the same in the response message as in the associated initiating message.

5.1.1 Initiating message

The data part of an initiating message may contain parameters as specified in section 6 of this TS.

5.1.2 Response message

If the class 1 elementary procedure requested by the initiating message was successfully executed, the response message data part from a single-antenna device shall be <OK>. Additional information may follow in the data part. The response message data part from a multi-antenna device starts with the antenna number followed by <OK> and optional additional information.

If the elementary procedure requested by the initiating message was not successfully executed, the response message data part from a single-antenna device shall within a default period of 1 second unless otherwise specified, be <FAIL>

The following octets ~~may~~ shall contain a ~~additional~~ return codes ~~which~~ describes why the execution of the requested procedure failed. The response message data part form a multi-antenna device starts with the antenna number followed by <OK> and a ~~optional additional~~ return codes ~~which~~ describes why the execution of the requested procedure failed.

Return codes marked with an X in the Alarm column of annex A in this TS are used to report operating conditions in alarm procedures (see sections 6.6.5 and 6.7.6 for details).

In some situations an initiating message can cause a change of operating conditions, for instance a SetTilt procedure might cause a RET device to discover that an adjuster is jammed or that a previously jammed adjuster works normally again. In these cases an alarm procedure reporting the change of operating conditions shall be used in addition to the regular <OK> or <FAIL> response message.

A complete annotated table of all return codes with their corresponding hexadecimal numbers is provided in annex A of this TS.

----------NEXT CHANGED SECTION----------

6.4 Description of elementary procedures

Table 6.4.1: Description of elementary procedures

Name: The name used to refer to the elementary procedure				
Code: The code is defined here. All other code references are informative	**Issued by:** Primary device or secondary device	**Procedure class:** Class 1 or Class 2	**Download operation:** FFS	**Download boot mode:** Defines whether the procedure shall be supported when the secondary device is in the download boot mode state

Table 6.4.2: Initiating message parameters and format

Number	Length	Type	Description
The enumerated order in which the parameter occurs in the data field of the message. The first number is 1.	**The length of the parameter, in number of octets, if defined.**	**The data type used in the parameter**	**Description of the parameter.**

Table 6.4.3: Response message parameters and format

Number	Length	Type	Description
The enumerated order in which the parameter occurs in the data field of the message. The first	**The length of the parameter, in number of octets, if defined.**	**The data type used in the parameter**	**Description of the parameter.**

number is 1.			

Table 6.4.4: Response message parameters and format for common class 1 elementary procedures upon error

Number	Length	Type	Description
1	**1 octet**	**ReturnCode**	**Return code FAIL**
2	**1 octet**	**ReturnCode**	**Reason for failure**

Table 6.4.5: Response message parameters and format for single antenna class 1 elementary procedures upon error

Number	Length	Type	Description
1	**1 octet**	**ReturnCode**	**Return code FAIL**
2	**1 octet**	**ReturnCode**	**Reason for failure**

Table 6.4.6: Response message parameters and format for multi-antenna class 1 elementary procedures upon error

Number	Length	Type	Description
1	**1 octet**	**Unsigned integer**	**Antenna number**
2	**1 octet**	**ReturnCode**	**Return code FAIL**
3	**1 octet**	**ReturnCode**	**Reason for failure**

NOTE: The response message in the elementary procedure AntennaGetAntennaNumber, has the format given in table 6.4.5, although it is defined as a multi-antenna class 1 elementary procedure.

Description:

Describes the purpose of the elementary procedure.

~~**Initiating message data format:**~~

~~Describes the initiating message parameter order.~~

~~**Response message data format:**~~

~~Describes the response message data parameter order in case of procedure success.~~

~~**Response message data format upon error:**~~

~~Describes the response message data parameter order in case of procedure failure.~~

Applicable return codes:

Lists all allowed return codes for the procedure.

6.5 Common elementary procedures

6.5.1 Reset Software

Table 6.5.1.1: Elementary procedure Reset Software

Name: **ResetSoftware**				
Code: **0x03**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **Yes**	Download boot mode: **Mandatory**

Table 6.5.1.2: Initiating message parameters and format for Reset Software

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Table 6.5.1.3: Response message parameters and format for Reset Software

Number	Length	Type	Description
1	**1 octet**	**ReturnCode**	**Return code OK**

Description:

On the receipt of the initiating message the secondary device shall set the HDLC address to the No-station address and place the device in the *No Address* state.

The device shall not execute the reset procedure before transport layer acknowledgement through sequence number update is received for the response. (See annex C in [2]).

The secondary device shall not fail to reset for any reason.

~~**Initiating message data format:**~~

~~No data carried in the message.~~

~~**Response message data format:**~~

~~<OK>~~

~~**Response message data format upon error:**~~

~~No error accepted.~~

Applicable return codes:

<OK>

6.5.2 Get Error Status

Table 6.5.2.1: Elementary procedure Get Error Status

Name: **GetErrorStatus**

Code: **0x04**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Mandatory**

Table 6.5.2.2: Initiating message parameters and format for Get Error Status

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Table 6.5.2.3: Response message parameters and format for Get Error Status

Number	Length	Type	Description
1	**1 octet**	**ReturnCode**	**Return code OK**
i + 1	**1 octet**	**AlarmCode**	**Active error number i**

i = 1 … N

Description:

On receipt of the initiating message the secondary device reports back the ~~return~~ alarm codes corresponding to the active errors in the secondary device to the primary device.

~~**Initiating message data format:**~~

~~No data carried.~~

~~**Response message data format:**~~

~~<OK><ReturnCode1>…<ReturnCodeN>~~

~~**Response message data format upon error:**~~

~~<FAIL><ReturnCode1><ReturnCode2>…<ReturnCodeN>~~

Applicable return codes:

ActuatorDetectionFail, ActuatorJamPermanent, ActuatorJamTemporary, EEPROMError, FlashEraseError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, PositionLost, RAMError, UARTError, Busy, DataError, DeviceDisabled, UnknownParameter, WorkingSoftwareMissing, DownloadInProgress

NOTE1: ActuatorDetectionFail, ActuatorJamPermanent, ActuatorJamTemporary, EEPROMError, FlashEraseError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, PositionLost, RAMError, UARTError WorkingSoftwareMissing may be part of OK response message

NOTE2: Busy, DataError, DeviceDisabled, UnknownParameter, OtherHardwareError, OtherSoftwareError, FlashError, RAMError, UARTError, DownloadInProgress may be part of FAIL response message.

6.5.3 Get Information

Table 6.5.3.1: Elementary procedure Get Information

Name: **GetInformation**				
Code: **0x05**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Mandatory**

Table 6.5.3.2: Initiating message parameters and format for Get Information

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Table 6.5.3.3: Response message parameters and format for Get Information

Number	Length	Type	Description
1	**1 octet**	**ReturnCode**	**Return code OK**
2	**1 octet**	**Unsigned integer**	**Length of parameter 3 in number of octets**
3		**TextString**	**Product number**
4	**1 octet**	**Unsigned integer**	**Length of parameter 5 in number of octets**
5		**TextString**	**Serial number**
6	**1 octet**	**Unsigned integer**	**Length of parameter 7 in number of octets**
7		**TextString**	**Hardware Version**
8	**1 octet**	**Unsigned integer**	**Length of parameter 9 in number of octets**
9		**TextString**	**Software Version**

Description:

On receipt of the initiating message the secondary device shall return the product number ProdNr and the serial number SerNr of the secondary device. If known, also the hardware version and the software version may be returned. The software version should indicate the version number of the currently executed software.

The parameters HWVersion and SWVersion in the response message refer to the version designators of the hardware and installed software of the secondary device. If the application is missing or no version number is found, then an empty string shall be returned as the version number.

~~**Initiating message data format:**~~

~~No data carried.~~

~~**Response message data format:**~~

~~<OK><LengthOctet><ProdNr><LengthOctet><SerNr><LengthOctet><HWVersion><LengthOctet><SWVersion>~~

~~**Response message data format upon error:**~~

~~<FAIL><ReturnCode1>…<ReturnCodeN>~~

Applicable return codes:

Busy, FlashError, OtherHardwareError, OtherSoftwareError, RAMError, UARTError. UnknownParameter, DownloadInProgress

6.5.4 Clear Active Alarms

Table 6.5.4.1: Elementary procedure Clear Active Alarms

Name: **ClearActiveAlarms**				
Code: **0x06**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Mandatory**

Table 6.5.4.2: Initiating message parameters and format for Clear Active Alarms

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Table 6.5.4.3: Response message parameters and format for Clear Active Alarms

Number	Length	Type	Description
1	**1 octet**	**ReturnCode**	**Return code OK**

Description:

On receipt of the initiating message the secondary device first clears all stored alarm information and then returns a procedure response message.

~~**Initiating message data format:**~~

~~No data carried.~~

~~**Response message data format:**~~

~~<OK>~~

~~**Response message data format upon error:**~~

~~<FAIL><ReturnCode1>…<ReturnCodeN>~~

Applicable return codes:

Busy, FlashEraseError, FlashError, OtherHardwareError, OtherSoftwareError, RAMError, UARTError, UnknownParameter, DownloadInProgress.

6.5.5 Alarm Subscribe

Table 6.5.5.1: Elementary procedure Alarm Subscribe

Name: **AlarmSubscribe**				
Code: **0x12**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Mandatory**

Table 6.5.5.2: Initiating ~~pessage~~ message parameters and format for Alarm Subscribe

Number	Length	Type	Description

None	**0 octets**	**None**	**No data carried**

Table 6.5.5.3: Response message parameters and format for Alarm Subscribe

Number	Length	Type	Description
1	**1 octet**	**ReturnCode**	**Return code OK**

Description:

On receipt of the initiating message the secondary device shall start executing its normal operation code.

~~**Initiating message data format:**~~

~~No Data carried.~~

~~**Response message data format:**~~

~~<OK>~~

~~**Response message data format upon error:**~~

~~<FAIL><ReturnCode1>…<ReturnCodeN>~~

Applicable return codes:

Invalid Software

NOTE1: UnknownParameter is the response code used if any data is carried in the initiating message.

6.5.6 Self Test

Table 6.5.6.1: Elementary procedure Self Test

Name: **SelfTest**				
Code: **0x0A**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.5.6.2: Initiating message parameters and format for Self Test

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Table 6.5.6.3: Response message parameters and format for Self Test

Number	Length	Type	Description
1	**1 octet**	**ReturnCode**	**Return code OK**
i + 1	**1 octet**	**AlarmCode**	**Alarm code for fault i detected during self test.**

i = 1 … N

Description:

On receipt of the initiating message the secondary device executes a test procedure which may include a check of physical and processor functions. The specific tests to be performed are implementation specific, and may include the movement of the adjuster up to <FFS> degrees.

The response message of the secondary device on the procedure provides information on detected faults or, if no fault is detected, with confidence that the operation of the device is normal in all respects.

During the test the operational parameters of the device shall not change beyond operationally acceptable limits and on completion all parameters shall be returned to their initial values.

In the normal response message, in which the self test was executed successfully, the return codes are set to report possible detected functional errors during the self test. If no errors are detected, this shall be signalled by no return codes following <OK>.

In the case of an error response message, the self test could not be executed and the return codes relate to the inability of the device to perform the requested self-test operation.

~~**Initiating message data format:**~~

~~No data carried.~~

~~**Response message data format:**~~

~~<OK><ReturnCode1>…<ReturnCodeN>~~

~~**Response message data format upon error:**~~

~~<FAIL><ReturnCode1>…<ReturnCodeN>~~

Applicable return codes:

ActuatorDetectionFail, ActuatorJamPermanent, ActuatorJamTemporary, Busy, DataError, DeviceDisabled, EEPROMError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, PositionLost, RAMError, UARTError, WorkingSoftwareMissing, DownloadInProgress.

NOTE1: Only Busy, DataError, DeviceDisabled, EEPROMError, FlashError, OtherHardwareError, OtherSoftwareError, RAMError, UARTError, WorkingSoftwareMissing, DownloadInProgress may be return codes in the fail response message.

6.5.7 Set Device Data

Table 6.5.7.1: Elementary procedure Set Device Data

Name: **SetDeviceData**				
Code: **0x0E**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.5.7.2: Initiating message parameters and format for Set Device Data

Number	Length	Type	Description
1	**1 octet**	**Hexadecimal**	**Field number, see annex B**
2	**See annex B**	**See annex B**	**Data to write**

Table 6.5.7.3: Response message parameters and format for Set Device Data

Number	Length	Type	Description
1	**1 octet**	**ReturnCode**	**Return code OK**

Description:

On receipt of the initiating message the secondary device should write the data given in the parameters of the initiating message into the fields optionally provided for configuration data and listed in annex B of this TS. If an attempt is made to write to fields which are designated as read only, the return code *ReadOnly* is returned and the data for those fields is ignored. If an attempt is made to write to fields which are not supported by the device the return code *UnknownParameter* is returned and the data for those fields is ignored.

~~**Initiating message data format:**~~

~~<field number><data octets for field>~~

~~**Response message data format:**~~

~~<OK>~~

~~**Response message data format upon error:**~~

~~<FAIL><ReturnCode1>...<ReturnCodeN>~~

Applicable return codes:

Busy, DataError, DeviceDisabled, EEPROMError, FlashError, FlashEraseError, OtherHardwareError, OtherSoftwareError, OutOfRange, RAMError, UARTError, ReadOnly, UnknownParameter, DownloadInProgress

6.5.8 Get Device Data

Table 6.5.8.1: Elementary procedure Get Device Data

Name: **GetDeviceData**				
Code: **0x0F**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.5.8.2: Initiating message parameters and format for Get Device Data

Number	Length	Type	Description
i	**1 octet**	**Hexadecimal**	**Field number; see annex B**

i = 1 … N

Table 6.5.8.3: Response message parameters and format for Get Device Data

Number	Length	Type	Description
1	**1 octet**	**ReturnCode**	**Return code OK**
2 i	**1 octet**	**Unsigned integer**	**Field number, see annex B**
2 i + 1	**See annex B**	**See annex B**	

i = 1 … N

Description:

In this procedure the secondary device shall return the data stored in the fields for configuration data specified by the field numbers in the procedure and listed in annex B of this TS. The field numbers are not necessarily contiguous or ordered. For field numbers which are not supported by the secondary device those parameters are not returned.

~~**Initiating message data format:**~~

~~<field number 1><field number 2>…<field number N>~~

~~**Response message data format:**~~

~~<OK><field number 1><data octets for field number 1><field number 2><data octets for field number 2> …<field number N><data octets for field number N>~~

~~**Response message data format upon error:**~~

~~<FAIL><ReturnCode1>…<ReturnCodeN>~~

Applicable return codes:

Busy, DeviceDisabled, EEPROMError, FlashError, OtherHardwareError, OtherSoftwareError, RAMError, UARTError, DownloadInProgress

6.5.9 Read User Data

Table 6.5.9.1: Elementary procedure Read User Data

Name: **ReadUserData**				
Code: **0x10**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.~~2~~5.9.2: Initiating message parameters and format for Read User Data

Number	Length	Type	Description
1	**2 octets**	**Integer**	**Memory offset**
2	**1 octet**	**Integer**	**Number of octets to read**

Table 6.5.9.3: Response message parameters and format for Read User Data

Number	Length	Type	Description
1	**1 octet**	**ReturnCode**	**Return code OK**
2	**Number of octets**	**User specific**	**User data**

Description:

On receipt of the initiating message the secondary device sends back stored user specific data to the primary device.

~~**Initiating message data format:**~~

~~<OffsetLowOctet><OffsetHighOctet><NumberOfOctetsToRead>~~

~~**Response message data format:**~~

~~<OK><octet 1> … <octet N>~~

~~**Response message data format upon error:**~~

~~<FAIL><ReturnCode1>…<ReturnCodeN>~~

Applicable return codes:

Busy, DataError, DeviceDisabled, EEPROMError, FlashError, OtherHardwareError, OtherSoftwareError, OutOfRange, RAMError, UARTError, DownloadInProgress

NOTE1: The return code OutOfRange is used if the given memory offset is outside the valid range.

6.5.10 Write User Data

Table 6.5.10.1: Elementary procedure Write User Data

Name: **WriteUserData**				
Code: **0x11**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.5.10.2: Initiating message parameters and format for Write User Data

Number	Length	Type	Description
1	**2 octets**	**Integer**	**Memory offset**
2	**1 octet**	**Integer**	**Number of octets to write**
3	**Message specific, given by parameter 2**	**~~Octets~~User specific**	**Data to write**

Table 6.5.10.3: Response message parameters and format for Write User Data

Number	Length	Type	Description
1	**1 octet**	**ReturnCode**	**Return code OK**

Description:

On receipt of the initiating message the secondary device shall store user specific data in non-volatile memory. The user data is stored using the relative memory address offset given in the initiating message and starting with zero.

~~**Initiating message data format:**~~

~~<OffsetLowOctet><OffsetHighOctet><NumberOfOctetsToWrite><octet 1> … <octet N>~~

~~**Response message data format:**~~

~~<OK>~~

~~**Response message data format upon error:**~~

~~<FAIL><ReturnCode1>…<ReturnCodeN>~~

Applicable return codes:

Busy, DeviceDisabled, EEPROMError, FlashError, OtherHardwareError, OtherSoftwareError, OutOfRange, RAMError, UARTError, DownloadInProgress

NOTE1: The return code OutOfRange is used if the needed data field is too long or if the given memory address is outside the valid address space.

6.5.11 Boot Mode Start

Table 6.5.11.1: Elementary procedure Boot Mode Start

Name: **BootModeStart**				
Code: **0x40**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **Yes**	Download boot mode: **Mandatory**

Table 6.5.11.2: Initiating message parameters and format for Boot Mode Start

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Table 6.5.11.3: Response message parameters and format for Boot Mode Start

Number	Length	Type	Description
1	**1 octet**	**ReturnCode**	**Return code OK**

Description:

On receipt of this initiating message the software download process shall be initiated. Following transition to the boot state, the secondary device sends <OK>. . Non-volatile memory in the secondary device may be erased after this procedure is completed.

~~**Initiating message data format:**~~

~~No data carried.~~

~~**Response message data format:**~~

~~<OK>~~

~~**Response message data format upon error:**~~

~~<FAIL><ReturnCode 1>…<ReturnCode N>~~

Applicable return codes:

Busy, FlashEraseError, OtherHardwareError, OtherSoftwareError, RAMError, UARTError

6.5.12 Download Application

Table 6.~~2~~5.12.1: Elementary procedure Download Application

Name: **DownloadApplication**				
Code: **0x41**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **Yes**	Download boot mode: **Mandatory**

Table 6.~~2~~5.12.2: Initiating message parameters and format for Download Application

Number	Length	Type	Description
None	**Vendor specific**	**Vendor specific**	**Software data**

Table 6.5.12.3: Response message parameters and format for Download application

Number	Length	Type	Description
1	**1 octet**	**ReturnCode**	**Return code OK**

Description:

Repeated use of this elementary procedure transfers software data from the primary device to the secondary device.

~~**Initiating message data format:**~~

~~<octet 1><octet 2>…<octet N>~~

~~**Response message data format:**~~

~~<OK>~~

~~**Response message data format upon error:**~~

~~<FAIL><ReturnCode 1>…<ReturnCode N>~~

Applicable return codes:

Busy, ChecksumError, DataError, EEPROMError, FlashEraseError, FlashError, OtherHardwareError, OtherSoftwareError, RAMError, UARTError, UnknownProcedure, InvalidSoftware, TooMuchData

NOTE1: UnknownProcedure may not be returned in boot mode.

6.5.13 Download End

Table 6.5.13.1: Elementary procedure Download End

Name: **DownloadEnd**				
Code: **0x42**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **Yes**	Download boot mode: **Mandatory**

Table 6.5.13.2: Initiating message parameters and format for Download End

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Table 6.5.13.3: Response message parameters and format for Download End

Number	Length	Type	Description
1	**1 octet**	**ReturnCode**	**Return code OK**

Description:

This elementary procedure signals the end of a multi-message data transfer to the secondary device. The secondary device responds after verifying the received data. If new software has been downloaded, the secondary device shall reset autonomously after completion of the layer 2 response to activate the new software.

~~**Initiating message data format:**~~

~~No data carried.~~

~~**Response message data format:**~~

~~<OK>~~

~~**Response message data format upon error:**~~

~~<FAIL><ReturnCode 1>…<ReturnCode N>~~

Applicable return codes:

Busy, ChecksumError, EEPROMError, FlashEraseError, FlashError, OtherHardwareError, OtherSoftwareError, RAMError, UARTError, UnknownProcedure, WorkingSoftwareMissing, InvalidSoftware, TooMuchData

NOTE1: UnknownProcedure may not be returned during software download.

6.6 Single-antenna elementary procedures

6.6.1 Calibrate

Table 6.6.1.1: Elementary procedure Calibrate

Name: **Calibrate**				
Code: **0x31**	Issued by: **Primary Device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.6.1.2: Initiating message parameters and format for Calibrate

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Table 6.6.1.3: Response message parameters and format for Calibrate

Number	Length	Type	Description
1	**1 octet**	**ReturnCode**	**Return code OK**

Description:

On receipt of the initiating message the secondary device shall perform a calibration of the RET antenna where the actuator is driven through its whole tilt range.

The response time to this Calibrate procedure shall be less than 4 minutes.

~~**Initiating message data format:**~~

~~No data carried.~~

~~**Response message data format:**~~

~~<OK>~~

~~**Response message data format upon error:**~~

~~<FAIL><ReturnCode1>…<ReturnCodeN>~~

Applicable return codes:

ActuatorDetectionFail, ActuatorJamPermanent, ActuatorJamTemporary, Busy, DeviceDisabled, EEPROMError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, PositionLost, RAMError, UARTError,UnknownProcedure, DownloadInProgress

6.6.2 Send Configuration Data

Table 6.6.2.1: Elementary procedure Send Configuration Data

Name: **SendConfigurationData**				
Code: **0x32**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.6.2.2: Initiating message parameters and format for Send Configuration Data

Number	Length	Type	Description
1	**Vendor specific; Maximum of 70 octets**	**Vendor specific**	**Configuration data**

Table 6.6.2.3: Response message parameters and format for Send Configuration Data

Number	Length	Type	Description
1	**1 octet**	**ReturnCode**	**Return code OK**

Description:

On receipt of the initiating message the secondary device shall store the provided vendor and antenna specific configuration data for the relationship between the movement of the drive system and the beam tilt position of the antenna.

If the configuration data exceeds 70 octets, the data shall be split into a number of 70 octet segments and one final segment with whatever is left. The primary device transmits the segments in order. The layer 2 sequence numbers guarantee that no segment will be lost or received out of order.

~~**Initiating message data format:**~~

~~<DataOctet1>…<DataOctetN>~~

~~**Response message data format:**~~

~~<OK>~~

~~**Response message data format upon error:**~~

~~<FAIL><ReturnCode1>…<ReturnCodeN>~~

Applicable return codes:

tbd

6.6.3 Set Tilt

Table 6.6.3.1: Elementary procedure Set Tilt

Name: **SetTilt**				
Code: **0x33**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.6.3.2: Initiating message parameters and format for Set Tilt

Number	Length	Type	Description
1	**2 octets**	**16 bit signed little-endian**	**Tilt value**

Table 6.6.3.3: Response message parameters and format for Set Tilt

Number	Length	Type	Description
1	**1 octet**	**ReturnCode**	**Return code OK**

Description:

On receipt of the initiating message the secondary device shall set the electrical tilt in increments of 0.1°. The electrical tilt value describes the elevation angle between the direction orthogonal to the antenna element axis and the maximum of its main beam in the elevation plane. A positive electrical tilt angle means that the antenna beam is directed below the direction orthogonal to the antenna axis.

The secondary device shall respond to the initiating message in less than 2 minutes.

The format of the provided tilt value is a 2-complement 16 bit signed number sent in little-endian order. Tilt values are given in 0.1° increments starting from zero, for example: Tilt 3.2° is 0x0020, Tilt – 3.2° is 0xFFE0.

~~**Initiating message data format:**~~

~~<TiltLowOctet><TiltHighOctet>~~

~~**Response message data format:**~~

~~<OK>~~

~~**Response message data format upon error:**~~

~~<FAIL><ReturnCode1>…<ReturnCodeN>~~

Applicable return codes:

ActuatorDetectionFail, ActuatorJamPermanent, ActuatorJamTemporary, Busy, DataError, DeviceDisabled, EEPROMError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, OutOfRange, PositionLost, RAMError, UARTError, UnknownProcedure, DownloadInProgress

6.6.4 Get Tilt

Table 6.6.4.1: Elementary procedure Get Tilt

Name: **GetTilt**				
Code: **0x34**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.6.4.2: Initiating message parameters and format for Get Tilt

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Table 6.6.4.3: Response message parameters and format for Get Tilt

Number	Length	Type	Description
1	**1 octet**	**ReturnCode**	**Return code OK**
2	**2 octets**	**Signed integer**	**Tilt value**

Description:

On receipt of the initiating message the secondary device will return the current tilt value.

The returned tilt value is given in increments of 0.1° in the format specified in section 6.6.3.

~~**Initiating message data format:**~~

~~No data carried.~~

~~**Response message data format:**~~

~~<OK><TiltLowOctet><TiltHighOctet>~~

~~**Response message data format upon error:**~~

~~<FAIL><ReturnCode1>…<ReturnCodeN>~~

Applicable return codes:

Busy, DeviceDisabled, EEPROMError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, PositionLost, RAMError, UARTError, UnknownProcedure, DownloadInProgress

6.6.5 Alarm

Table 6.6.5.1: Elementary procedure Alarm

Name: **Alarm**				
Code: **0x07**	Issued by: **Secondary device**	Procedure class: **2**	Download operation: **Yes**	Download boot mode: **Mandatory**

Table 6.6.5.2: Initiating message parameters and format for Alarm

Number	Length	Type	Description
2 i – 1	**1 octet**	**Hexadecimal**	**Return code i; see annex A**
2 I	**1 octet**	**Hexadecimal**	**State flag i**

i = 1 … N

Description:

The secondary device uses this procedure to report alarms to the primary device. An alarm procedure is performed if an error state has changed since the previous alarm message. All return codes marked as alarms in Annex A of this TS may be used in the initiating message.

~~**Initiating message data format:**~~

~~<ReturnCode1><StateFlag1>…<ReturnCodeN><StateFlagN>~~

6.7 Multi-antenna elementary procedures

6.7.1 Antenna Calibrate

Table 6.7.1.1: Elementary procedure Antenna Calibrate

Name: **AntennaCalibrate**				
Code: **0x80**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.7.1.2: Initiating message parameters and format for Antenna Calibrate

Number	Length	Type	Description
1	**1 octet**	**Integer**	**Antenna number**

Table 6.7.1.3: Response message parameters and format for Antenna Calibrate

Number	Length	Type	Description
1	**1 octet**	**Unsigned integer**	**Antenna number**
2	**1 octet**	**ReturnCode**	**Return code OK**

Description:

On receipt of the initiating message the secondary device shall perform a calibration of the antenna addressed by the antenna number. During the calibration the actuator is driven through the whole tilt range of the antenna.

The response time to this Antenna Calibrate procedure shall be less than 4 minutes.

~~**Initiating message data format:**~~

~~<antenna number>~~

~~**Response message data format:**~~

~~<antenna number><OK>~~

~~**Response message data format upon error:**~~

~~<antenna number><FAIL><ReturnCode1>…<ReturnCodeN>~~

Applicable return codes:

ActuatorDetectionFail, ActuatorJamPermanent, ActuatorJamTemporary, Busy, DataError, DeviceDisabled, EEPROMError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, PositionLost, RAMError, UARTError, UnknownProcedure, UnknownAntennaNumber, DownloadInProgress

6.7.2 Antenna Set Tilt

Table 6.7.2.1: Elementary procedure Antenna Set Tilt

Name: **AntennaSetTilt**				
Code: **0x81**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.7.2.2: Initiating message parameters and format for Antenna Set Tilt

Number	Length	Type	Description
1	**1 octet**	**Integer**	**Antenna number**
2	**2 octets**	**16-bit signed little-endian**	**Tilt value**

Table 6.7.2.3: Response message parameters and format for Antenna Set Tilt

Number	Length	Type	Description
1	**1 octet**	**Unsigned integer**	**Antenna number**

2	1 octet	ReturnCode	Return code OK

Description:

On receipt of the initiating message the secondary device shall set the electrical tilt of the antenna addressed by the antenna number in increments of 0.1°. The electrical tilt value describes the elevation angle between the direction orthogonal to the antenna element axis and the maximum of its main beam in the elevation plane. A positive electrical tilt angle means that the antenna beam is directed below the direction orthogonal to the antenna axis.

The secondary device shall respond to the initiating message in less than 2 minutes.

The format of the provided tilt value is a 2-complement 16 bit signed number sent in little-endian order. Tilt values are given in 0.1° increments starting from zero, for example: Tilt 3.2° is 0x0020, Tilt – 3.2° is 0xFFE0.

~~**Initiating message data format:**~~

~~<antenna number><TiltLowOctet><TiltHighOctet>~~

~~**Response message data format:**~~

~~<antenna number><OK>~~

~~**Response message data format upon error:**~~

~~<antenna number><FAIL><ReturnCode1>…<ReturnCodeN>~~

Applicable return codes:

ActuatorDetectionFail, ActuatorJamPermanent, ActuatorJamTemporary, Busy, DataError, DeviceDisabled, EEPROMError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, OutOfRange, PositionLost, RAMError, UARTError, UnknownProcedure, UnknownAntennaNumber, DownloadInProgress

6.7.3 Antenna Get Tilt

Table 6.7.3.1: Elementary procedure Antenna Get Tilt

Name: **AntennaGetTilt**				
Code: **0x82**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.7.3.2: Initiating message parameters and format for Antenna Get Tilt

Number	Length	Type	Description
1	**1 octet**	**Integer**	**Antenna number**

Table 6.7.3.3: Response message parameters and format for Antenna Get Tilt

Number	Length	Type	Description
1	1 octet	Unsigned integer	Antenna number
2	1 octet	ReturnCode	Return code OK
3	2 octets	Signed integer	Tilt value

Description:

On receipt of the initiating message the secondary device will return the current tilt value of the antenna addressed by the antenna number.

The returned tilt value is given in increments of 0.1° in the format specified in section 6.7.2.

~~**Initiating message data format:**~~

~~<antenna number>~~

~~**Response message data format:**~~

~~<antenna number><OK><TiltLowOctet><TiltHighOctet>~~

~~**Response message data format upon error:**~~

~~<antenna number><FAIL><ReturnCode1>…<ReturnCodeN>~~

Applicable return codes:

ActuatorDetectionFail, Busy, DataError, DeviceDisabled, EEPROMError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, PositionLost, RAMError, UARTError, UnknownProcedure, UnknownAntennaNumber, DownloadInProgress

6.7.4 Antenna Set Data

Table 6.7.4.1: Elementary procedure Antenna Set Data

Name: **AntennaSetData**				
Code: **0x83**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.7.4.2: Initiating message parameters and format for Antenna Set Data

Number	Length	Type	Description
1	**1 octet**	**Integer**	**Antenna number**
2	**1 octet**	**Hexadecimal**	**Field number; see annex B**
3	**See annex B**	**See annex B**	**Data to write**

Table 6.7.4.3: Response message parameters and format for Antenna Set Data

Number	Length	Type	Description
1	**1 octet**	**Unsigned integer**	**Antenna number**
2	**1 octet**	**ReturnCode**	**Return code OK**

Description:

On receipt of the initiating message the secondary device should write the provided data for the antenna addressed by the antenna number into the fields optionally provided for configuration data and listed in annex B of this TS. If an attempt is made to write to fields which are not supported by a particular device no error is returned but the data for

those fields is ignored. If an attempt is made to write to fields which are not supported for the addressed antenna the return code UnknownParameter is returned and the data for those fields is ignored.

~~**Initiating message data format:**~~

~~<antenna number><field number><data octets for field>~~

~~**Response message data format:**~~

~~<antenna number><OK>~~

~~**Response message data format upon error:**~~

~~<antenna number><FAIL><ReturnCode1>...<ReturnCodeN>~~

Applicable return codes:

Busy, DataError, DeviceDisabled, EEPROMError, FlasheError, OtherHardwareError, OtherSoftwareError, OutOfRange, RAMError, UARTError, UnknownProcedure, ReadOnly, UnknownParameter, UnknownAntennaNumber, DownloadInProgress

6.7.5 Antenna Get Data

Table 6.7.5.1: Elementary procedure Antenna Get Data

Name: **AntennaGetData**				
Code: **0x84**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.7.5.2: Initiating message parameters and format for Antenna Get Data

Number	Length	Type	Description
1	**1 octet**	**Integer**	**Antenna number**
–i	**1 octet**	**Hexadecimal**	**Field number to read; see annex B**

i = 1 … N

Table 6.7.5.3: Response message parameters and format for Antenna Get Data

Number	Length	Type	Description
1	**1 octet**	**Unsigned integer**	**Antenna number**
2	**1 octet**	**ReturnCode**	**Return code OK**
2 i + 1	**1 octet**	**Unsigned integer**	**Field number i, see annex B**
2 i + 2	**See annex B**	**See annex B**	**Field value i**

i = 1 … N

Description:

On receipt of the initiating message the secondary device shall return the data stored for the addressed antenna in the fields for configuration data specified by the field numbers in the initiating message and listed in annex B of this TS. The field numbers are not necessarily contiguous or ordered. For field numbers which are not supported by the secondary device for the addressed antenna no data is returned for that field.

~~**Initiating message data format:**~~

~~<antenna number><field number 1><field number 2> …<field number N>~~

~~**Response message data format:**~~

~~<antenna number><OK><field number 1><data octets for field number 1><field number 2><data octets for field number 2> … <field number N><data octets for field number N>~~

~~**Response message data format upon error:**~~

~~<antenna number><FAIL><ReturnCode1>...<ReturnCodeN>~~

Applicable return codes:

Busy, DataError, DeviceDisabled, EEPROMError, FlashError, OtherHardwareError, OtherSoftewareError, RAMError, UARTError, UnknownProcedure, UnknownAntennaNumber, DownloadInProgress

6.7.6 Antenna Alarm

Table 6.7.6.1: Elementary procedure Antenna Alarm

Name: **AntennaAlarm**				
Code: 0x85	Issued by: **Secondary device**	Procedure class: **2**	Download operation: **Yes**	Download boot mode: **Mandatory**

Table 6.7.6.2: Initiating message parameters and format for Antenna Alarm

Number	Length	Type	Description
1	**1 octet**	**Integer**	**Antenna number**
2 i – 1	**1 octet**	**Hexadecimal**	**Return code i; see annex A**
2 i	**1 octet**	**Hexadecimal**	**State flag i**

i = 1 … N

Description:

The multi-antenna secondary device uses this procedure to report antenna alarms to the primary device. This procedure shall only be performed if the secondary has performed an AlarmSubscribe procedure since its latest reset. Multi-antenna devices shall use this *AntennaAlarm* procedure only for multi-antenna specific alarms and the *Alarm* procedure in subclause 6.6.5 for the other alarms.

~~**Initiating message data format:**~~

~~<antenna number><ReturnCode1><StateFlag1>…<ReturnCodeN><StateFlagN>~~

6.7.7 Antenna Clear Active Alarms

Table 6.7.7.1: Elementary procedure Clear Antenna Alarms

Name: **AntennaClearActiveAlarms**				
Code: 0x86	Issued by: **Secondary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.7.7.2: Initiating message parameters and format for ClearAntenna Alarm

Number	Length	Type	Description
1	**1 octet**	**Integer**	**Antenna number**

Table 6.7.7.3: Response message parameters and format for Antenna Clear Active Alarms

Number	Length	Type	Description
1	**1 octet**	**Unsigned integer**	**Antenna number**
2	**1 octet**	**ReturnCode**	**Return code OK**

Description:

On receipt of the initiating message the secondary device first clears all stored alarm information for the addressed antenna and then returns a procedure response message.

~~**Initiating message data format:**~~

~~<antenna number>~~

~~**Response message data format:**~~

~~<antenna number><OK>~~

~~**Response message data format upon error:**~~

~~<antenna number><FAIL><ReturnCode1>…<ReturnCodeN>~~

Applicable return codes:

Busy, FlashEraseError, FlashError, OtherHardwareError, OtherSoftwareError, RAMError, UARTError, UnknownParameter, DownloadInProgress.

6.7.8 Antenna Get Error Status

Table 6.~~5.2~~7.8.1: Elementary procedure Antenna Get Error Status

Name: **AntennaGetErrorStatus**				
Code: **0x87**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **No**

Table 6.~~5.2~~7.8.2: Initiating message parameters and format for Get Error Status

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Table 6.7.8.3: Response message parameters and format for Antenna Get Error Status

Number	Length	Type	Description

1	1 octet	Unsigned integer	Antenna number
2	1 octet	ReturnCode	Return code OK
i + 2	1 octet	AlarmCode	Alarm code for error number i

i = 1 ... N

Description:

On receipt of the initiating message the secondary device reports back the return codes for the addressed antenna corresponding to the active errors in the secondary device to the primary device.

~~**Initiating message data format:**~~

~~No data carried.~~

~~**Response message data format:**~~

~~<antenna number><OK><ReturnCode1>…<ReturnCodeN>~~

~~**Response message data format upon error:**~~

~~<antenna number><FAIL><ReturnCode1><ReturnCode2>…<ReturnCodeN>~~

Applicable return codes:

ActuatorDetectionFail, ActuatorJamPermanent, ActuatorJamTemporary, EEPROMError, FlashEraseError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, PositionLost, RAMError, UARTError, Busy, DataError, DeviceDisabled, UnknownParameter, WorkingSoftwareMissing, DownloadInProgress

NOTE1: ActuatorDetectionFail, ActuatorJamPermanent, ActuatorJamTemporary, EEPROMError, FlashEraseError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, PositionLost, RAMError, UARTError WorkingSoftwareMissing may be part of OK response message

NOTE2: Busy, DataError, DeviceDisabled, UnknownParameter, OtherHardwareError, OtherSoftwareError, FlashError, RAMError, UARTError, DownloadInProgress may be part of FAIL response message.

6.7.9 Antenna Get Number Of Antennas

Table 6.7.5.1: Elementary procedure Antenna Get Number Of Antennas

Name: **AntennaGetNumberOfAntennas**				
Code: **0x88**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.7.5.2: Initiating message parameters and format for Antenna Get Data

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Table 6.7.9.3: Response message parameters and format for Antenna Get number of antennas

Number	Length	Type	Description
1	1 octet	ReturnCode	Return code OK

2	**1 octet**	**Unsigned integer**	**Number of antennas**

Description:

On receipt of the initiating message the secondary device shall return the number of antennas it controls.

~~**Initiating message data format:**~~

~~No data carried.~~

~~**Response message data format:**~~

~~<OK><number of antennas>~~

~~**Response message data format upon error:**~~

~~<FAIL><ReturnCode1>...<ReturnCodeN>~~

Applicable return codes:

Busy, DataError, DeviceDisabled, EEPROMError, FlashError, OtherHardwareError, OtherSoftewareError, RAMError, UARTError, UnknownProcedure, UnknownAntennaNumber, DownloadInProgress

7 Unknown elementary procedures

If a secondary device is unable to recognise an initiating message, the secondary device shall respond as follows:

Table 7.1: Response message parameters and format for unknown procedures

Number	Length	Type	Description
1	**1 octet**	**ReturnCode**	**Return code FAIL**
2	**1 octet**	**ReturnCode**	**Return code UnknownProcedure**

~~Response message data format:~~

~~<FAIL><ReturnCode for UnknownProcedure>~~

3GPP TSG-RAN3 Meeting #45 ***Tdoc ⌘ R3-041676***
Shin-Yokohama, Japan, 15th - 19th November 2004

CR-Form-v7.1

CHANGE REQUEST

⌘ **25.463** **CR** **12** ⌘ **rev** **2** ⌘ Current version: **6.0.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ [ ] ME [ ] Radio Access Network [X] Core Network [ ]

Title:	⌘	Return code clean-up and clarification
Source:	⌘	RAN3
Work item code:	⌘	RANimp-TiltAnt
Date:	⌘	18/11/2004
Category:	⌘	**F**
Release:	⌘	Rel-6

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

Ph2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*
Rel-7	*(Release 7)*

Reason for change:	⌘	Unclear specification
Summary of change:	⌘	The use of Return codes in message interpretation errors is introduced. The return code set is aligned with the present Elementary Procedure set. More details in the use of return codes included in the procedure definitions. Missing return codes are introduced. Redundant return codes are deleted.
Consequences if not approved:	⌘	The return code set is not suitable for the procedure set. Message interpretation error handling is undefined. This may raise interoperability issues.

Clauses affected:	⌘	6.2, 6.4 to 6.7, Annex A

Other specs affected: ⌘	Y	N	
		X	Other core specifications
		X	Test specifications
		X	O&M Specifications

⌘

Other comments: ⌘

How to create CRs using this form:
Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm.
Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.

2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

6 Control elementary procedures

6.1 State Model

The state model describing the secondary device is FFS.

Figure 6.1: State model for secondary device

6.2 General procedure handling

All procedures are blocking i.e. no new initiation messages will have to be executed before a response message has been delivered as result of the previously initiated procedure.

The Reset Software procedure shall always be handled in all states and never be blocked.

6.2.n Procedure message interpretation

The following message interpretation rules apply in the order mentioned:

1. Any message shorter than 3 octets shall be disregarded.
2. If a message has a length inconsistent with its “Number of data octets” field value it shall be responded with a failure message stating “FormatError” as the cause of failure. The response message shall be to the initiating message identified by the procedure code.
3. If a secondary device in the OperatingMode state is receiving a procedure message of an optional procedure not supported or if the procedure is inapplicable to the device type, it shall respond with a failure message stating “UnsupportedProcedure” as the cause of failure.
4. If a secondary device receives a procedure message, part of the software download procedure sequence described in Annex C, without having received the previous procedure messages in that sequence it shall respond with a failure message stating “InvalidProcedureSequence” as the cause of failure.
5. If a secondary device in the DownloadMode state is receiving a procedure message not supported in that state it shall respond with a failure message stating “WorkingSoftwareMissing” as the cause of failure.
6. If a secondary device in the OperatingMode state is receiving a correct procedure message with a procedure code not known it shall respond with a failure message stating “UnknownProcedure” as the cause of failure.
7. If a message has a length inconsistent with the defined message length in the procedure definition it shall be responded with a failure message stating “FormatError” as the cause of failure. The response message shall be to the initiating message identified by the procedure code.

6.3 Overview of elementary procedures

The set of elementary procedures for RET antenna control provides procedure-oriented instructions. An overview of the procedures is given in annex D. Table 6.3.1 lists all common elementary procedures described in section 6.5. Table 6.3.2 lists all elementary procedures specific for single-antenna device types described in section 6.6. Table 6.3.3 lists all elementary procedures specific for multi-antenna device types described in section 6.7. Section 6.4 describes how to interpret the elementary procedure definitions in sections 6.5 to 6.7.

Some elementary procedures shall be performed in sequence as described in Annex C for the software download.

Table 6.3.1: Common elementary procedure set for all device types

Command	Requirement	Comment
Reset Software	mandatory	
Get Error Status	mandatory	
Get Information	mandatory	
Clear Active Alarms	mandatory	
Alarm Subscribe	mandatory	
Read User Data	mandatory	
Write User Data	mandatory	
Self Test	mandatory	
Set Device Data	mandatory	
Get Device Data	mandatory	
Boot Mode Start	optional	This procedure is mandatory if the software download feature is supported.
Download Application	optional	This procedure is mandatory if the software download feature is supported.
Download End	optional	This procedure is mandatory if the software download feature is supported.

Table 6.3.2: Elementary procedure set for singe-antenna device type

Command	Requirement	Comment
Calibrate	mandatory	
Send Configuration Data	mandatory	
Set Tilt	mandatory	
Get Tilt	mandatory	
Alarm	mandatory	

Table 6.3.3: Elementary procedure set for multiple-antenna device type

Command	Requirement	Comment
Antenna Calibrate	mandatory	
Antenna Set Tilt	mandatory	
Antenna Get Tilt	mandatory	
Antenna Set Data	mandatory	
Antenna Get Data	mandatory	
Antenna Alarm	mandatory	
Antenna Clear Active Alarms	mandatory	
Antenna Get Error Status	mandatory	
Antenna Get Number Of Antennas	mandatory	

6.4 Description of elementary procedures

Table 6.4.1: Description of elementary procedures

Name: The name used to refer to the elementary procedure				
Code: The code is defined here. All other code references are informative	**Issued by:** Primary device or secondary device	**Procedure class:** Class 1 or Class 2	**Download operation:** FFS	**Download boot mode:** Defines whether the procedure shall be supported when the secondary device is in the download boot mode state

Table 6.4.2: Initiating message parameters

Number	Length	Type	Description

Description:

Describes the purpose of the elementary procedure.

Initiating message data format:

Describes the initiating message parameter order.

Response message data format:

Describes the response message data parameter order in case of procedure success.

Response message data format upon error:

Describes the response message data parameter order in case of procedure failure.

Table 6.4.x: Return codes

OK	FAIL	Comment
All return codes applicable in a response message to a successful procedure, except "OK", are listed here. The return codes are listed by name as defined in Annex A.	**All return codes applicable in a response message to a failing procedure, except "FAIL" are listed here. The return codes are listed by name as defined in Annex A.**	**Any comment needed for clarification.**

~~**Applicable return codes:**~~

~~Lists all allowed return codes for the procedure.~~

6.5 Common elementary procedures

6.5.1 Reset Software

Table 6.5.1.1: Elementary procedure Reset Software

Name: **ResetSoftware**				
Code: **0x03**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **Yes**	Download boot mode: **Mandatory**

Table 6.5.1.2: Initiating message parameters for Reset Software

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Description:

On the receipt of the initiating message the secondary device shall set the HDLC address to the No-station address and place the device in the *No Address* state.

The device shall not execute the reset procedure before transport layer acknowledgement through sequence number update is received for the response. (See annex C in [2]).

The secondary device shall not fail to reset for any reason.

Initiating message data format:

No data carried in the message.

Response message data format:

<OK>

Response message data format upon error:

No error accepted.

Table 6.5.1.x: Return codes for Reset Software

OK	FAIL	Comment
	FormatError	**In case of format error, the procedure code validity is not secured.**

~~**Applicable return codes:**~~

~~<OK>~~

6.5.2 Get Error Status

Table 6.5.2.1: Elementary procedure Get Error Status

Name: **GetErrorStatus**				
Code: **0x04**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Mandatory**

Table 6.5.2.2: Initiating message parameters for Get Error Status

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Description:

On receipt of the initiating message the secondary device reports back the return codes corresponding to the active errors in the secondary device to the primary device.

Initiating message data format:

No data carried.

Response message data format:

<OK><ReturnCode1>…<ReturnCodeN>

Response message data format upon error:

<FAIL><ReturnCode1><ReturnCode2>…<ReturnCodeN>

Table 6.5.2.x: Return codes for Get Error Status

OK	FAIL	Comment
All return codes marked as used for alarms in Annex A.	**FormatError** **Busy** **WorkingSoftwareMissing**	

~~**Applicable return codes:**~~

~~ActuatorDetectionFail, ActuatorJamPermanent, ActuatorJamTemporary, EEPROMError, FlashEraseError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, PositionLost, RAMError, UARTError, Busy, DataError, DeviceDisabled, UnknownParameter, WorkingSoftwareMissing, DownloadInProgress~~

~~NOTE1: ActuatorDetectionFail, ActuatorJamPermanent, ActuatorJamTemporary, EEPROMError, FlashEraseError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, PositionLost, RAMError, UARTError WorkingSoftwareMissing may be part of OK response message~~

~~NOTE2: Busy, DataError, DeviceDisabled, UnknownParameter, OtherHardwareError, OtherSoftwareError, FlashError, RAMError, UARTError, DownloadInProgress may be part of FAIL response message.~~

6.5.3 Get Information

Table 6.5.3.1: Elementary procedure Get Information

Name: **GetInformation**				
Code: **0x05**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Mandatory**

Table 6.5.3.2: Initiating message parameters for Get Information

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Description:

On receipt of the initiating message the secondary device shall return the product number ProdNr and the serial number SerNr of the secondary device. If known, also the hardware version and the software version may be returned. The software version should indicate the version number of the currently executed software.

The parameters HWVersion and SWVersion in the response message refer to the version designators of the hardware and installed software of the secondary device. If the application is missing or no version number is found, then an empty string shall be returned as the version number.

Initiating message data format:

No data carried.

Response message data format:

<OK><LengthOctet><ProdNr><LengthOctet><SerNr><LengthOctet><HWVersion><LengthOctet><SWVersion>

Response message data format upon error:

<FAIL><ReturnCode1>…<ReturnCodeN>

Table 6.5.3.x: Return codes for Get Information

OK	FAIL	Comment
	FormatError **Busy**	

~~**Applicable return codes:**~~

~~Busy, FlashError, OtherHardwareError, OtherSoftwareError, RAMError, UARTError. UnknownParameter, DownloadInProgress~~

6.5.4 Clear Active Alarms

Table 6.5.4.1: Elementary procedure Clear Active Alarms

Name: **ClearActiveAlarms**				
Code:	Issued by:	Procedure class:	Download operation:	Download boot mode:

0x06	**Primary device**	**1**	**No**	**Mandatory**

Table 6.5.4.2: Initiating message parameters for Clear Active Alarms

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Description:

On receipt of the initiating message the secondary device first clears all stored alarm information and then returns a procedure response message.

Initiating message data format:

No data carried.

Response message data format:

<OK>

Response message data format upon error:

<FAIL><ReturnCode1>…<ReturnCodeN>

Table 6.5.4.x: Return codes for Clear Active Alarms

OK	FAIL	Comment
	FormatError **Busy** **WorkingSoftwareMissing**	

~~**Applicable return codes:**~~

~~Busy, FlashEraseError, FlashError, OtherHardwareError, OtherSoftwareError, RAMError, UARTError, UnknownParameter, DownloadInProgress.~~

6.5.5 Alarm Subscribe

Table 6.5.5.1: Elementary procedure Alarm Subscribe

Name: **AlarmSubscribe**				
Code: **0x12**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Mandatory**

Table 6.5.5.2: Initiating pessage parameters for Alarm Subscribe

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Description:

On receipt of the initiating message the secondary device shall start executing its normal operation code.

Initiating message data format:

No Data carried.

Response message data format:

<OK>

Response message data format upon error:

<FAIL><ReturnCode1>…<ReturnCodeN>

Table 6.5.5.x: Return codes for Alarm Subscribe

OK	FAIL	Comment
	FormatError **Busy** **WorkingSoftwareMissing**	

~~**Applicable return codes:**~~

~~Invalid Software~~

~~NOTE1: UnknownParameter is the response code used if any data is carried in the initiating message.~~

6.5.6 Self Test

Table 6.5.6.1: Elementary procedure Self Test

Name: **SelfTest**				
Code: **0x0A**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.5.6.2: Initiating message parameters for Self Test

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Description:

On receipt of the initiating message the secondary device executes a test procedure which may include a check of physical and processor functions. The specific tests to be performed are implementation specific, and may include the movement of the adjuster up to <FFS> degrees.

The response message of the secondary device on the procedure provides information on detected faults or, if no fault is detected, with confidence that the operation of the device is normal in all respects.

During the test the operational parameters of the device shall not change beyond operationally acceptable limits and on completion all parameters shall be returned to their initial values.

In the normal response message, in which the self test was executed successfully, the return codes are set to report possible detected functional errors during the self test. If no errors are detected, this shall be signalled by no return codes following <OK>.

In the case of an error response message, the self test could not be executed and the return codes relate to the inability of the device to perform the requested self-test operation.

Initiating message data format:

No data carried.

Response message data format:

<OK><ReturnCode1>…<ReturnCodeN>

Response message data format upon error:

<FAIL><ReturnCode1>…<ReturnCodeN>

Table 6.5.6.x: Return codes for Self Test

OK	FAIL	Comment
All return codes marked as alarms in Annex A.	**FormatError** **Busy** **WorkingSoftwareMissing** **NotCalibrated** **NotScaled**	

~~**Applicable return codes:**~~

~~ActuatorDetectionFail, ActuatorJamPermanent, ActuatorJamTemporary, Busy, DataError, DeviceDisabled, EEPROMError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, PositionLost, RAMError, UARTError, WorkingSoftwareMissing, DownloadInProgress.~~

~~NOTE1: Only Busy, DataError, DeviceDisabled, EEPROMError, FlashError, OtherHardwareError, OtherSoftwareError, RAMError, UARTError, WorkingSoftwareMissing, DownloadInProgress may be return codes in the fail response message.~~

6.5.7 Set Device Data

Table 6.5.7.1: Elementary procedure Set Device Data

Name: **SetDeviceData**				
Code: **0x0E**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.5.7.2: Initiating message parameters for Set Device Data

Number	Length	Type	Description
1	**1 octet**	**Hexadecimal**	**Field number, see annex B**
2	**See annex B**	**See annex B**	**Data to write**

Description:

On receipt of the initiating message the secondary device should write the data given in the parameters of the initiating message into the fields optionally provided for configuration data and listed in annex B of this TS. If an attempt is made to write to fields which are designated as read only, the return code *ReadOnly* is returned and the data for those fields is ignored. If an attempt is made to write to fields which are not supported by the device the return code *UnknownParameter* is returned and the data for those fields is ignored.

Initiating message data format:

<field number><data octets for field>

Response message data format:

<OK>

Response message data format upon error:

<FAIL><ReturnCode1>...<ReturnCodeN>

Table 6.5.7.x: Return codes for Set Device Data

OK	FAIL	Comment
	FormatError **Busy** **WorkingSoftwareMissing** **HardwareError** **ReadOnly** **UnknownParameter**	

~~**Applicable return codes:**~~

~~Busy, DataError, DeviceDisabled, EEPROMError, FlashError, FlashEraseError, OtherHardwareError, OtherSoftwareError, OutOfRange, RAMError, UARTError, ReadOnly, UnknownParameter, DownloadInProgress~~

6.5.8 Get Device Data

Table 6.5.8.1: Elementary procedure Get Device Data

Name: **GetDeviceData**				
Code: **0x0F**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.5.8.2: Initiating message parameters for Get Device Data

Number	Length	Type	Description
i	**1 octet**	**Hexadecimal**	**Field number; see annex B**

i = 1 … N

Description:

In this procedure the secondary device shall return the data stored in the fields for configuration data specified by the field numbers in the procedure and listed in annex B of this TS. The field numbers are not necessarily contiguous or ordered. For field numbers which are not supported by the secondary device those parameters are not returned.

Initiating message data format:

<field number 1><field number 2>…<field number N>

Response message data format:

<OK><field number 1><data octets for field number 1><field number 2><data octets for field number 2> …<field number N><data octets for field number N>

Response message data format upon error:

<FAIL><ReturnCode1>...<ReturnCodeN>

Table 6.5.8.x: Return codes for Get Device Data

OK	FAIL	Comment
	FormatError **Busy** **WorkingSoftwareMissing**	

~~**Applicable return codes:**~~

~~Busy, DeviceDisabled, EEPROMError, FlashError, OtherHardwareError, OtherSoftwareError, RAMError, UARTError, DownloadInProgress~~

6.5.9 Read User Data

Table 6.5.9.1: Elementary procedure Read User Data

Name: **ReadUserData**				
Code: **0x10**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.2.9.2: Initiating message parameters for Read User Data

Number	Length	Type	Description
1	**2 octets**	**Integer**	**Memory offset**
2	**1 octet**	**Integer**	**Number of octets to read**

Description:

On receipt of the initiating message the secondary device sends back stored user specific data to the primary device.

Initiating message data format:

<OffsetLowOctet><OffsetHighOctet><NumberOfOctetsToRead>

Response message data format:

<OK><octet 1> ... <octet N>

Response message data format upon error:

<FAIL><ReturnCode1>…<ReturnCodeN>

~~**Applicable return codes:**~~

Table 6.5.9.x: Return codes for Read User Data

OK	FAIL	Comment
	FormatError **Busy** **WorkingSoftwareMissing** **OutOfRange**	**The return code OutOfRange is used if the given memory address range is outside the valid address space.**

~~Busy, DataError, DeviceDisabled, EEPROMError, FlashError, OtherHardwareError, OtherSoftwareError, OutOfRange, RAMError, UARTError, DownloadInProgress~~

~~NOTE1: The return code OutOfRange is used if the given memory offset is outside the valid range.~~

6.5.10 Write User Data

Table 6.5.10.1: Elementary procedure Write User Data

Name: **WriteUserData**				
Code: **0x11**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.5.10.2: Initiating message parameters for Write User Data

Number	Length	Type	Description
1	**2 octets**	**Integer**	**Memory offset**
2	**1 octet**	**Integer**	**Number of octets to write**
3	**Message specific, given by parameter 2**	**Octets**	**Data to write**

Description:

On receipt of the initiating message the secondary device shall store user specific data in non-volatile memory. The user data is stored using the relative memory address offset given in the initiating message and starting with zero.

Initiating message data format:

<OffsetLowOctet><OffsetHighOctet><NumberOfOctetsToWrite><octet 1> ... <octet N>

Response message data format:

<OK>

Response message data format upon error:

<FAIL><ReturnCode1>…<ReturnCodeN>

Table 6.5.10.x: Return codes for Write User Data

OK	FAIL	Comment
	FormatError **Busy** **WorkingSoftwareMissing** **HardwareError** **OutOfRange**	**The return code OutOfRange is used if the given memory address range is outside the valid address space.**

~~**Applicable return codes:**~~

~~Busy, DeviceDisabled, EEPROMError, FlashError, OtherHardwareError, OtherSoftwareError, OutOfRange, RAMError, UARTError, DownloadInProgress~~

~~NOTE1: The return code OutOfRange is used if the needed data field is too long or if the given memory address is outside the valid address space.~~

6.5.11 Boot Mode Start

Table 6.5.11.1: Elementary procedure Boot Mode Start

Name: **BootModeStart**				
Code: **0x40**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **Yes**	Download boot mode: **Mandatory**

Table 6.5.11.2: Initiating message parameters for Boot Mode Start

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Description:

On receipt of this initiating message the software download process shall be initiated. Following transition to the boot state, the secondary device sends <OK>. . Non-volatile memory in the secondary device may be erased after this procedure is completed.

Initiating message data format:

No data carried.

Response message data format:

<OK>

Response message data format upon error:

<FAIL><ReturnCode 1>…<ReturnCode N>

Table 6.5.11.x: Return codes for Boot Mode Start

OK	FAIL	Comment

	FormatError **Busy** **UnsupportedProcedure**	

~~**Applicable return codes:**~~

~~Busy, FlashEraseError, OtherHardwareError, OtherSoftwareError, RAMError, UARTError~~

6.5.12 Download Application

Table 6.2.12.1: Elementary procedure Download Application

Name: **DownloadApplication**				
Code: **0x41**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **Yes**	Download boot mode: **Mandatory**

Table 6.2.12.2: Initiating message parameters for Download Application

Number	Length	Type	Description
None	**Vendor specific**	**Vendor specific**	**Software data**

Description:

Repeated use of this elementary procedure transfers software data from the primary device to the secondary device.

Initiating message data format:

<octet 1><octet 2>…<octet N>

Response message data format:

<OK>

Response message data format upon error:

<FAIL><ReturnCode 1>…<ReturnCode N>

Table 6.5.12.x: Return codes for Download Application

OK	FAIL	Comment
	FormatError **Busy** **HardwareError** **InvalidFileContent** **InvalidProcedureSequence**	

~~**Applicable return codes:**~~

~~Busy, ChecksumError, DataError, EEPROMError, FlashEraseError, FlashError, OtherHardwareError, OtherSoftwareError, RAMError, UARTError, UnknownProcedure, InvalidSoftware, TooMuchData~~

~~NOTE1: UnknownProcedure may not be returned in boot mode.~~

6.5.13 Download End

Table 6.5.13.1: Elementary procedure Download End

Name: **DownloadEnd**				
Code: **0x42**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **Yes**	Download boot mode: **Mandatory**

Table 6.5.13.2: Initiating message parameters for Download End

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Description:

This elementary procedure signals the end of a multi-message data transfer to the secondary device. The secondary device responds after verifying the received data. If new software has been downloaded, the secondary device shall reset autonomously after completion of the layer 2 response to activate the new software.

Initiating message data format:

No data carried.

Response message data format:

<OK>

Response message data format upon error:

<FAIL><ReturnCode 1>…<ReturnCode N>

Table 6.5.13.x: Return codes for Download End

OK	FAIL	Comment
	FormatError **Busy** **HardwareError** **ChecksumError** **InvalidFileContent** **InvalidProcedureSequence**	

~~**Applicable return codes:**~~

~~Busy, ChecksumError, EEPROMError, FlashEraseError, FlashError, OtherHardwareError, OtherSoftwareError, RAMError, UARTError, UnknownProcedure, WorkingSoftwareMissing, InvalidSoftware, TooMuchData~~

~~NOTE1: UnknownProcedure may not be returned during software download.~~

6.6 Single-antenna elementary procedures

6.6.1 Calibrate

Table 6.6.1.1: Elementary procedure Calibrate

Name: **Calibrate**				
Code: **0x31**	Issued by: **Primary Device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.6.1.2: Initiating message parameters for Calibrate

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Description:

On receipt of the initiating message the secondary device shall perform a calibration of the RET antenna where the actuator is driven through its whole tilt range.

The response time to this Calibrate procedure shall be less than 4 minutes.

Initiating message data format:

No data carried.

Response message data format:

<OK>

Response message data format upon error:

<FAIL><ReturnCode1>…<ReturnCodeN>

Table 6.6.1.x: Return codes for Calibrate

OK	FAIL	Comment
	FormatError **Busy** **HardwareError** **WorkingSoftwareMissing** **MotorJam** **ActuatorJam** **NotConfigured** **UnsupportedProcedure**	

~~**Applicable return codes:**~~

~~ActuatorDetectionFail, ActuatorJamPermanent, ActuatorJamTemporary, Busy, DeviceDisabled, EEPROMError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, PositionLost, RAMError, UARTError,UnknownProcedure, DownloadInProgress~~

6.6.2 Send Configuration Data

Table 6.6.2.1: Elementary procedure Send Configuration Data

Name: **SendConfigurationData**				
Code: **0x32**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.6.2.2: Initiating message parameters for Send Configuration Data

Number	Length	Type	Description
1	**Vendor specific; Maximum of 70 octets**	**Vendor specific**	**Configuration data**

Description:

On receipt of the initiating message the secondary device shall store the provided vendor and antenna specific configuration data for the relationship between the movement of the drive system and the beam tilt position of the antenna.

If the configuration data exceeds 70 octets, the data shall be split into a number of 70 octet segments and one final segment with whatever is left. The primary device transmits the segments in order. The layer 2 sequence numbers guarantee that no segment will be lost or received out of order.

Initiating message data format:

<DataOctet1>…<DataOctetN>

Response message data format:

<OK>

Response message data format upon error:

<FAIL><ReturnCode1>…<ReturnCodeN>

Table 6.6.2.x: Return codes for Send Configuration Data

OK	FAIL	Comment
	FormatError **Busy** **HardwareError** **WorkingSoftwareMissing** **ChecksumError** **InvalidFileContent** **UnsupportedProcedure**	

~~**Applicable return codes:**~~

~~tbd~~

6.6.3 Set Tilt

Table 6.6.3.1: Elementary procedure Set Tilt

Name: **SetTilt**				
Code: **0x33**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.6.3.2: Initiating message parameters for Set Tilt

Number	Length	Type	Description
1	**2 octets**	**16 bit signed little-endian**	**Tilt value**

Description:

On receipt of the initiating message the secondary device shall set the electrical tilt in increments of 0.1°. The electrical tilt value describes the elevation angle between the direction orthogonal to the antenna element axis and the maximum of its main beam in the elevation plane. A positive electrical tilt angle means that the antenna beam is directed below the direction orthogonal to the antenna axis.

The secondary device shall respond to the initiating message in less than 2 minutes.

The format of the provided tilt value is a 2-complement 16 bit signed number sent in little-endian order. Tilt values are given in 0.1° increments starting from zero, for example: Tilt 3.2° is 0x0020, Tilt – 3.2° is 0xFFE0.

Initiating message data format:

<TiltLowOctet><TiltHighOctet>

Response message data format:

<OK>

Response message data format upon error:

<FAIL><ReturnCode1>…<ReturnCodeN>

Table 6.6.3.x: Return codes for Set Tilt

OK	FAIL	Comment
	FormatError **Busy** **HardwareError** **WorkingSoftwareMissing** **MotorJam** **ActuatorJam** **NotConfigured** **NotCalibrated** **OutOfRange** **UnsupportedProcedure**	

~~**Applicable return codes:**~~

~~ActuatorDetectionFail, ActuatorJamPermanent, ActuatorJamTemporary, Busy, DataError, DeviceDisabled, EEPROMError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, OutOfRange, PositionLost, RAMError, UARTError, UnknownProcedure, DownloadInProgress~~

6.6.4 Get Tilt

Table 6.6.4.1: Elementary procedure Get Tilt

Name: **GetTilt**				
Code: **0x34**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.6.4.2: Initiating message parameters for Get Tilt

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Description:

On receipt of the initiating message the secondary device will return the current tilt value.

The returned tilt value is given in increments of 0.1° in the format specified in section 6.6.3.

Initiating message data format:

No data carried.

Response message data format:

<OK><TiltLowOctet><TiltHighOctet>

Response message data format upon error:

<FAIL><ReturnCode1>…<ReturnCodeN>

~~**Applicable return codes:**~~

~~**Busy, DeviceDisabled, EEPROMError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, PositionLost, RAMError, UARTError, UnknownProcedure, DownloadInProgress**~~

Table 6.6.4.x: Return codes for Get Tilt

OK	FAIL	Comment
	FormatError **Busy** **HardwareError** **WorkingSoftwareMissing** **NotCalibrated** **NotConfigured** **UnsupportedProcedure**	**HardwareError shall only be used if error is detected in tilt detector.**

6.6.5 Alarm

Table 6.6.5.1: Elementary procedure Alarm

Name: **Alarm**				
Code: **0x07**	Issued by: **Secondary device**	Procedure class: **2**	Download operation: **Yes**	Download boot mode: **Mandatory**

Table 6.6.5.2: Initiating message parameters for Alarm

Number	Length	Type	Description
2 i – 1	**1 octet**	**Hexadecimal**	**Return code i; see annex A**
2 I	**1 octet**	**Hexadecimal**	**State flag i**

i = 1 … N

Description:

The secondary device uses this procedure to report alarms to the primary device. An alarm procedure is performed if an error state has changed since the previous alarm message. All return codes marked as alarms in Annex A of this TS may be used in the initiating message.

Initiating message data format:

<ReturnCode1><StateFlag1>…<ReturnCodeN><StateFlagN>

6.7 Multi-antenna elementary procedures

6.7.1 Antenna Calibrate

Table 6.7.1.1: Elementary procedure Antenna Calibrate

Name: **AntennaCalibrate**				
Code: **0x80**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.7.1.2: Initiating message parameters for Antenna Calibrate

Number	Length	Type	Description
1	**1 octet**	**Integer**	**Antenna number**

Description:

On receipt of the initiating message the secondary device shall perform a calibration of the antenna addressed by the antenna number. During the calibration the actuator is driven through the whole tilt range of the antenna.

The response time to this Antenna Calibrate procedure shall be less than 4 minutes.

Initiating message data format:

<antenna number>

Response message data format:

<antenna number><OK>

Response message data format upon error:

<antenna number><FAIL><ReturnCode1>…<ReturnCodeN>

Table 6.7.1.x: Return codes for Antenna Calibrate

OK	FAIL	Comment
	FormatError **Busy** **HardwareError** **WorkingSoftwareMissing** **MotorJam** **ActuatorJam** **NotConfigured** **UnsupportedProcedure**	**If the addressed antenna is not existing, FormatError is returned.**

~~**Applicable return codes:**~~

~~ActuatorDetectionFail, ActuatorJamPermanent, ActuatorJamTemporary, Busy, DataError, DeviceDisabled, EEPROMError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, PositionLost, RAMError, UARTError, UnknownProcedure, UnknownAntennaNumber, DownloadInProgress~~

6.7.2 Antenna Set Tilt

Table 6.7.2.1: Elementary procedure Antenna Set Tilt

Name: **AntennaSetTilt**				
Code: **0x81**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.7.2.2: Initiating message parameters for Antenna Set Tilt

Number	Length	Type	Description
1	**1 octet**	**Integer**	**Antenna number**
2	**2 octets**	**16-bit signed little-endian**	**Tilt value**

Description:

On receipt of the initiating message the secondary device shall set the electrical tilt of the antenna addressed by the antenna number in increments of 0.1°. The electrical tilt value describes the elevation angle between the direction orthogonal to the antenna element axis and the maximum of its main beam in the elevation plane. A positive electrical tilt angle means that the antenna beam is directed below the direction orthogonal to the antenna axis.

The secondary device shall respond to the initiating message in less than 2 minutes.

The format of the provided tilt value is a 2-complement 16 bit signed number sent in little-endian order. Tilt values are given in 0.1° increments starting from zero, for example: Tilt 3.2° is 0x0020, Tilt – 3.2° is 0xFFE0.

Initiating message data format:

<antenna number><TiltLowOctet><TiltHighOctet>

Response message data format:

<antenna number><OK>

Response message data format upon error:

<antenna number><FAIL><ReturnCode1>…<ReturnCodeN>

Table 6.7.2.x: Return codes for Antenna Set Tilt

OK	FAIL	Comment
	FormatError **Busy** **HardwareError** **WorkingSoftwareMissing** **MotorJam** **ActuatorJam** **NotConfigured** **NotCalibrated** **OutOfRange** **UnsupportedProcedure**	**If the addressed antenna is not existing, FormatError is returned.**

~~**Applicable return codes:**~~

~~ActuatorDetectionFail, ActuatorJamPermanent, ActuatorJamTemporary, Busy, DataError, DeviceDisabled, EEPROMError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, OutOfRange, PositionLost, RAMError, UARTError, UnknownProcedure, UnknownAntennaNumber, DownloadInProgress~~

6.7.3 Antenna Get Tilt

Table 6.7.3.1: Elementary procedure Antenna Get Tilt

Name: **AntennaGetTilt**				
Code: **0x82**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.7.3.2: Initiating message parameters for Antenna Get Tilt

Number	Length	Type	Description
1	**1 octet**	**Integer**	**Antenna number**

Description:

On receipt of the initiating message the secondary device will return the current tilt value of the antenna addressed by the antenna number.

The returned tilt value is given in increments of 0.1° in the format specified in section 6.7.2.

Initiating message data format:

<antenna number>

Response message data format:

<antenna number><OK><TiltLowOctet><TiltHighOctet>

Response message data format upon error:

<antenna number><FAIL><ReturnCode1>…<ReturnCodeN>

Table 6.7.3.x: Return codes for Antenna Get Tilt

OK	FAIL	Comment
	FormatError **Busy** **HardwareError** **WorkingSoftwareMissing** **NotConfigured** **NotCalibrated** **UnsupportedProcedure**	**If the addressed antenna is not existing, FormatError is returned.** **HardwareError shall only be used if error is detected in tilt detector.**

~~**Applicable return codes:**~~

~~ActuatorDetectionFail, Busy, DataError, DeviceDisabled, EEPROMError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, PositionLost, RAMError, UARTError, UnknownProcedure, UnknownAntennaNumber, DownloadInProgress~~

6.7.4 Antenna Set Data

Table 6.7.4.1: Elementary procedure Antenna Set Data

Name: **AntennaSetData**				
Code: **0x83**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.7.4.2: Initiating message parameters for Antenna Set Data

Number	Length	Type	Description
1	**1 octet**	**Integer**	**Antenna number**
2	**1 octet**	**Hexadecimal**	**Field number; see annex B**
3	**See annex B**	**See annex B**	**Data to write**

Description:

On receipt of the initiating message the secondary device should write the provided data for the antenna addressed by the antenna number into the fields optionally provided for configuration data and listed in annex B of this TS. If an attempt is made to write to fields which are not supported by a particular device no error is returned but the data for those fields is ignored. If an attempt is made to write to fields which are not supported for the addressed antenna the return code UnknownParameter is returned and the data for those fields is ignored.

Initiating message data format:

<antenna number><field number><data octets for field>

Response message data format:

<antenna number><OK>

Response message data format upon error:

<antenna number><FAIL><ReturnCode1>...<ReturnCodeN>

Table 6.7.4.x: Return codes for Antenna Set Data

OK	FAIL	Comment
	FormatError **Busy** **HardwareError** **WorkingSoftwareMissing** **ReadOnly** **UnknownParameter** **UnsupportedProcedure**	**If the addressed antenna is not existing, FormatError is returned.**

~~**Applicable return codes:**~~

~~Busy, DataError, DeviceDisabled, EEPROMError, FlasheError, OtherHardwareError, OtherSoftwareError, OutOfRange, RAMError, UARTError, UnknownProcedure, ReadOnly, UnknownParameter, UnknownAntennaNumber, DownloadInProgress~~

6.7.5 Antenna Get Data

Table 6.7.5.1: Elementary procedure Antenna Get Data

Name: **AntennaGetData**				
Code: **0x84**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.7.5.2: Initiating message parameters for Antenna Get Data

Number	Length	Type	Description
1	**1 octet**	**Integer**	**Antenna number**
–i	**1 octet**	**Hexadecimal**	**Field number to read; see annex B**

i = 1 … N

Description:

On receipt of the initiating message the secondary device shall return the data stored for the addressed antenna in the fields for configuration data specified by the field numbers in the initiating message and listed in annex B of this TS. The field numbers are not necessarily contiguous or ordered. For field numbers which are not supported by the secondary device for the addressed antenna no data is returned for that field.

Initiating message data format:

<antenna number><field number 1><field number 2> …<field number N>

Response message data format:

<antenna number><OK><field number 1><data octets for field number 1><field number 2><data octets for field number 2> … <field number N><data octets for field number N>

Response message data format upon error:

<antenna number><FAIL><ReturnCode1>...<ReturnCodeN>

Table 6.7.5.x: Return codes for Antenna Get Data

OK	FAIL	Comment
	FormatError **Busy** **WorkingSoftwareMissing** **UnsupportedProcedure**	**If the addressed antenna is not existing, FormatError is returned.**

~~**Applicable return codes:**~~

~~Busy, DataError, DeviceDisabled, EEPROMError, FlashError, OtherHardwareError, OtherSoftewareError, RAMError, UARTError, UnknownProcedure, UnknownAntennaNumber, DownloadInProgress~~

6.7.6 Antenna Alarm

Table 6.7.6.1: Elementary procedure Antenna Alarm

Name: **AntennaAlarm**				
Code: 0x85	Issued by: **Secondary device**	Procedure class: **2**	Download operation: **Yes**	Download boot mode: **Mandatory**

Table 6.7.6.2: Initiating message parameters for Antenna Alarm

Number	Length	Type	Description
1	**1 octet**	**Integer**	**Antenna number**
2 i – 1	**1 octet**	**Hexadecimal**	**Return code i; see annex A**
2 i	**1 octet**	**Hexadecimal**	**State flag i**

i = 1 … N

Description:

The multi-antenna secondary device uses this procedure to report antenna alarms to the primary device. This procedure shall only be performed if the secondary has performed an AlarmSubscribe procedure since its latest reset. Multi-antenna devices shall use this *AntennaAlarm* procedure only for multi-antenna specific alarms and the *Alarm* procedure in subclause 6.6.5 for the other alarms.

Initiating message data format:

<antenna number><ReturnCode1><StateFlag1>…<ReturnCodeN><StateFlagN>

6.7.7 Antenna Clear Active Alarms

Table 6.7.7.1: Elementary procedure Clear Antenna Alarms

Name: **AntennaClearActiveAlarms**				
Code: 0x86	Issued by: **Secondary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.7.6.2: Initiating message parameters for ClearAntenna Alarm

Number	Length	Type	Description
1	**1 octet**	**Integer**	**Antenna number**

Description:

On receipt of the initiating message the secondary device first clears all stored alarm information for the addressed antenna and then returns a procedure response message.

Initiating message data format:

<antenna number>

Response message data format:

<antenna number><OK>

Response message data format upon error:

<antenna number><FAIL><ReturnCode1>…<ReturnCodeN>

Table 6.7.7.x: Return codes for Antenna Clear Active Alarms

OK	FAIL	Comment
	FormatError **Busy** **WorkingSoftwareMissing** **UnsupportedProcedure**	**If the addressed antenna is not existing, FormatError is returned.**

~~**Applicable return codes:**~~

~~Busy, FlashEraseError, FlashError, OtherHardwareError, OtherSoftwareError, RAMError, UARTError, UnknownParameter, DownloadInProgress~~.

6.7.8 Antenna Get Error Status

Table 6.5.2.1: Elementary procedure Antenna Get Error Status

Name: **AntennaGetErrorStatus**				
Code: **0x87**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **No**

Table 6.5.2.2: Initiating message parameters for Get Error Status

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Description:

On receipt of the initiating message the secondary device reports back the return codes for the addressed antenna corresponding to the active errors in the secondary device to the primary device.

Initiating message data format:

No data carried.

Response message data format:

<antenna number><OK><ReturnCode1>…<ReturnCodeN>

Response message data format upon error:

<antenna number><FAIL><ReturnCode1><ReturnCode2>…<ReturnCodeN>

Table 6.7.8.x: Return codes for Antenna Get Error Status

OK	FAIL	Comment
All return codes marked as used for alarms in Annex A	**FormatError** **Busy** **WorkingSoftwareMissing** **UnsupportedProcedure**	**If the addressed antenna is not existing, FormatError is returned.**

~~**Applicable return codes:**~~

~~ActuatorDetectionFail, ActuatorJamPermanent, ActuatorJamTemporary, EEPROMError, FlashEraseError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, PositionLost, RAMError, UARTError, Busy, DataError, DeviceDisabled, UnknownParameter, WorkingSoftwareMissing, DownloadInProgress~~

~~NOTE1: ActuatorDetectionFail, ActuatorJamPermanent, ActuatorJamTemporary, EEPROMError, FlashEraseError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, PositionLost, RAMError, UARTError WorkingSoftwareMissing may be part of OK response message~~

~~NOTE2: Busy, DataError, DeviceDisabled, UnknownParameter, OtherHardwareError, OtherSoftwareError, FlashError, RAMError, UARTError, DownloadInProgress may be part of FAIL response message.~~

6.7.9 Antenna Get Number Of Antennas

Table 6.7.5.1: Elementary procedure Antenna Get Number Of Antennas

Name: **AntennaGetNumberOfAntennas**				
Code: **0x88**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.7.5.2: Initiating message parameters for Antenna Get Data

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Description:

On receipt of the initiating message the secondary device shall return the number of antennas it controls.

Initiating message data format:

No data carried.

Response message data format:

<OK><number of antennas>

Response message data format upon error:

<FAIL><ReturnCode1>...<ReturnCodeN>

Table 6.7.9.x: Return codes for Antenna Get Number Of Antennas

OK	FAIL	Comment
	FormatError **Busy** **WorkingSoftwareMissing** **UnsupportedProcedure**	**If the addressed antenna is not existing, FormatError is returned.**

~~**Applicable return codes:**~~

~~Busy, DataError, DeviceDisabled, EEPROMError, FlashError, OtherHardwareError, OtherSoftewareError, RAMError, UARTError, UnknownProcedure, UnknownAntennaNumber, DownloadInProgress~~

7 Unknown elementary procedures

If a secondary device is unable to recognise an initiating message, the secondary device shall respond as follows:

Response message data format:

<FAIL><ReturnCode for UnknownProcedure>

Annex A (normative): Return Codes for secondary devices

Table A.1: Return Codes for Secondary Devices

Code	Meaning		Alarm	~~Software~~ DownloadMode state	~~Download Boot Mode~~
0x00	OK	Normal response		X	~~X~~
0x01	~~Actuator Detection Fail~~	~~Signals from the actuator are detected but are abnormal, for example due to failed calibration.~~	~~X~~		
0x02	~~Actuator~~ MotorJam ~~Permanent~~	Motor cannot move.. ~~Actuator cannot be moved permanently.~~	X		
0x03	Actuator-Jam ~~Temporary~~	Actuator jam has been detected. No movement of the actuator, but movement of the motor was detected. ~~No movement was detectedin response to the normal stimulus.~~	X		
0x04	~~Block Number Sequence Error~~	~~Used in combination with software download; block number sequence is wrong.~~			
0x05	Busy	The device is busy and cannot respond until an activity is complete.			
0x06	Checksum-Error	~~Used in combination with software download;~~ Cchecksum incorrect for otherwise valid data.			
0x07	~~Procedure Sequence Error~~	~~Used in combination with software download; procedure sequence is not permitted, e.g. a SetTilt procedure is received during software update sequence.~~			
0x08	~~Data Error~~	~~RET AP data fault, e.g. length of data is inconsistent with length fields.~~			
0x09	~~Device Disabled~~	~~Device is in logical Disabled state and cannot execute Set procedures.~~			
0x0A	~~EEPROM Error~~	~~EEPROM error detected~~	~~X~~	~~X~~	~~X~~
0x0B	FAIL	Abnormal response. Indicates that a procedure has not been executed.		X	~~X~~
0x0C	~~Flash Erase Error~~	~~Used in combination with software download; indicates error when erasing flash memory.~~	~~X~~	~~X~~	~~X~~
0x0D	~~Flash Error~~	~~Used in combination with software download; indicates error when writing to flash memory.~~	~~X~~	~~X~~	~~X~~
0x0E	Not-Calibrated	The device has not completed a calibration operation, or calibration has been lost.	X		
0x0F	~~Not Scaled~~NotConfigured	~~No setup table~~Actuator configuration data ~~has been stored~~is missing ~~in the device~~.	X		
0x11	~~Other~~ Hardware-Error	Any hardware error which cannot be classified. May not be reported as an alarm until the fault is likely to be persistent.	X	X	~~X~~
0x12	~~Other Software Error~~	~~Any software error which cannot be classified.~~	~~X~~	~~X~~	~~X~~

0x13	Out-~~o~~Of-Range	A ~~given~~ parameter given by an operator (e.g. tilt value or memory offset) is out of range.			
0x14	~~Position Lost~~	~~RET controller is unable to return a correct position value, for example there was a power failure while a SetTilt procedure was being executed.~~	~~X~~		
0x15	~~RAM Error~~	~~An error was detected in reading data to/from RAM~~	~~X~~	~~X~~	~~X~~
0x16	~~Segment Number Sequence Error~~	~~Used in combination with software download; block sequence number is wrong.~~			
0x17	~~UART Error~~	~~Hardware specific. This error may be sent after recovery from a temporary error which has prevented the sending or receiving of data.~~	~~X~~	~~X~~	~~X~~
0x19	Unknown-Procedure	Received procedure code is not defined. ~~in the 3GPP release version~~		X	~~X~~
0x1D	Read-Only	Invalid device data parameter usage.~~Used in combination with SetDeviceData procedure when the device parameter cannot be changed~~		X	~~X~~
0x1E	Unknown-Parameter	Specified parameter is not supported for the used procedure. ~~Used as a response to SetDeviceData if an attempt is made to set an unsupported field~~		X	
0x1F	~~Unknown Antenna Number~~	~~Specified antenna number for mulit-antenna devices is not supported~~			
0x20	~~Too Much Data~~	~~More data received during software download than can be stored~~		~~X~~	~~X~~
0x21	WorkingSoftware-Missing	~~Application code is missing or broken.~~ The unit is in ~~boot mode~~DownloadMode ~~and may be supporting a limited set of commands~~. Returned upon ~~unknown~~ unsupported procedure when in ~~boot~~ DownloadM~~m~~ode state.	~~X~~	X	~~X~~
0x22	Invalid-~~Software~~FileContent	~~Application code~~The data being downloaded is detected to be of wrong ~~type~~format or size. ~~Download of the application code will not be permitted.~~		X	~~X~~
0x23	~~Download In Progress~~	~~Used instead of UnknownProcedure during software download as response to all commands not supported in boot mode. May also be useful for one physical unit co-siting of several logical units to indicate that other logical units cannot be operated until software download has finished.~~		~~X~~	
0x24	FormatError	Responded if the procedure message is inconsistent or if an addressed field or antenna is invalid or the data parameter field length is inconsistent with the corresponding field length parameter.		X	
0x25	UnsupportedProcedure	The procedure is optional and not supported or the procedure does not apply to this device type			
0x26	InvalidProcedureSequence	Responded to indicate that the procedure sequence as described in Annex C is expected but not experienced by the secondary device.			
0x27	ActuatorInterference	An actuator movement outside the control of the RET unit has been detected. Probable cause is manual interference.	X		

3GPP TSG-RAN3 Meeting #45
Shin-Yokohama, Japan, 15th - 19th November 2004

Tdoc ⌘ R3-041677

CR-Form-v7.1

CHANGE REQUEST

⌘ **25.463** **CR** **15** ⌘ **rev** **2** ⌘ Current version: **6.0.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ ☐ ME ☐ Radio Access Network ☒ Core Network ☐

Title:	⌘	Clarification on the intention of the elementary procedures ReadUserData and WriteUserData
Source:	⌘	RAN3
Work item code:	⌘	RANimp-TiltAnt
Date:	⌘	18/11/2004
Category:	⌘	**F**
Release:	⌘	Rel-6

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:
Ph2 (GSM Phase 2)
R96 (Release 1996)
R97 (Release 1997)
R98 (Release 1998)
R99 (Release 1999)
Rel-4 (Release 4)
Rel-5 (Release 5)
Rel-6 (Release 6)
Rel-7 (Release 7)

Reason for change:	⌘	Intention of the procedures ReadUserData and WriteUserData is unclear
Summary of change:	⌘	Specifying the intended use of the procedures
Consequences if not approved:	⌘	Intention of procedures is unclear. This may hamper a correct implementation.

Clauses affected:	⌘	6.5.9 and 6.5.10

Other specs affected: ⌘	Y	N	
		X	Other core specifications
		X	Test specifications
		X	O&M Specifications

⌘

Other comments: ⌘

How to create CRs using this form:
Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm.
Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.

2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be

downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

6.5.9 Read User Data

Table 6.5.9.1: Elementary Procedure Read User Data

Name: **ReadUserData**				
Code: **0x10**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.2.9.2: Initiating Message Parameters for Read User Data

Number	Length	Type	Description
1	**2 octets**	**Integer**	**Memory offset**
2	**1 octet**	**Integer**	**Number of octets to read**

Description:

On receipt of the initiating message the secondary device sends back ~~stored~~ user specific data stored in a user data area to the primary device.

The user data area is intended for storage of user defined data, e.g. inventory information.

Initiating message data format:

<OffsetLowOctet><OffsetHighOctet><NumberOfOctetsToRead>

Response message data format:

<OK><octet 1> ... <octet N>

Response message data format upon error:

<FAIL><ReturnCode1>…<ReturnCodeN>

Applicable return codes:

Busy, DataError, DeviceDisabled, EEPROMError, FlashError, OtherHardwareError, OtherSoftwareError, OutOfRange, RAMError, UARTError, DownloadInProgress

NOTE1: The return code OutOfRange is used if the given memory offset is outside the valid range.

6.5.10 Write User Data

Table 6.5.10.1: Elementary Procedure Write User Data

Name: **WriteUserData**				
Code: **0x11**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.5.10.2: Initiating Message Parameters for Write User Data

Number	Length	Type	Description

1	**2 octets**	**Integer**	**Memory offset**
2	**1 octet**	**Integer**	**Number of octets to write**
3	**Message specific, given by parameter 2**	**Octets**	**Data to write**

Description:

On receipt of the initiating message the secondary device shall store user ~~specific~~ data in non-volatile memory. The user data is stored in the user data area using the relative memory address offset given in the initiating message and starting with zero.

The user data area is intended for storage of user defined data, e.g. inventory information.

Initiating message data format:

<OffsetLowOctet><OffsetHighOctet><NumberOfOctetsToWrite><octet 1> ... <octet N>

Response message data format:

<OK>

Response message data format upon error:

<FAIL><ReturnCode1>…<ReturnCodeN>

Applicable return codes:

Busy, DeviceDisabled, EEPROMError, FlashError, OtherHardwareError, OtherSoftwareError, OutOfRange, RAMError, UARTError, DownloadInProgress

NOTE1: The return code OutOfRange is used if the needed data field is too long or if the given memory address is outside the valid address space.

3GPP TSG-RAN3 Meeting #45
Shin-Yokohama, Japan, 15th - 19th November 2004

Tdoc ⌘ R3-041675

CR-Form-v7.1

CHANGE REQUEST

⌘ **25.463** **CR** **16** ⌘ **rev** **2** ⌘ Current version: **6.0.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ ☐ ME ☐ Radio Access Network ☒ Core Network ☐

Title:	⌘	Maximum data payload size in elementary procedures.
Source:	⌘	RAN3
Work item code:	⌘	RANimp-TiltAnt
Date:	⌘	18/11/2004
Category:	⌘	**F**
Release:	⌘	Rel-6

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

Ph2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*
Rel-7	*(Release 7)*

Reason for change:	⌘	Data payload size in RETAP Elementary Procedures is unclear. Risk of depleting the Data Payload size in some procedures.
Summary of change:	⌘	Data payload size in is defined for every Elementary Procedure.
Consequences if not approved:	⌘	Data payload size in RETAP Elementary Procedures remains unclear. Risk of depleting the Data Payload size in some procedures remain.

Clauses affected:	⌘	3.1, 5.1, 6.5.3, 6.5.8 - 6.5.12, 6.6.2, and 6.7.5			
Other specs affected:	⌘	**Y** / **N**	Other core specifications	⌘	CR3 for TS25.462
		☐ / **X**	Test specifications		
		☐ / **X**	O&M Specifications		
Other comments:	⌘				

How to create CRs using this form:
Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm.
Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.
2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be

downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

3 Definitions, symbols and abbreviations

3.1 Definitions

For the purposes of the present document, the following terms and definitions apply.

ASCII character: A character forming part of the International Reference Version of the 7-bit character set defined in ISO/IEC 646:1991

Calibrate: Exercise the antenna drive unit over its entire range of travel to ensure fault-free operation and synchronise the measured and actual beam tilt of the antenna

Configuration data: A stored table or function defining the relationship between the physical position of the drive and electrical beam-tilt

Device type: See section 4.7 in [3].

Elementary Procedure: The RETAP protocol consists of Elementary Procedures (EPs). An Elementary Procedure is a unit of interaction between the primary device (Node B) and the secondary devices (RET devices).

An EP consists of an initiating message and possibly a response message.

Two kinds of EPs are used:

- **Class 1**: Elementary Procedures with response (success or failure).
- **Class 2**: Elementary Procedures without response.

For **Class 1** EPs, the types of responses can be as follows:

Successful

- A signalling message explicitly indicates that the elementary procedure has been successfully completed with the receipt of the response.

Unsuccessful

- A signalling message explicitly indicates that the EP failed.

Class 2 EPs are considered always successful.

Little-endian: The order of transmission in which the least-significant octets of a multi-octet representation of a number are transmitted first. Little endian only applies to binary integer representations.

Max Data Receive Length: Secondary Payload Receive Length − 3 octets. (see [3])

Max Data Transmit Length: Secondary Payload Transmit Length − 3 octets. (see [3])

Return code: A 1-octet enumerated response message to an initiating message.

Tilt (also downtilt, tilt angle, beamtilt): The elevation angle between the direction orthogonal to the antenna element axis and the maximum of its main beam in the elevation plane. A positive electrical tilt angle means that the antenna beam is directed below the direction orthogonal to the antenna axis. An antenna has separate values for electrical and mechanical tilt. The mechanical tilt is fixed by the geometry of the installation. In this TS the tilt referred to is always the electrical tilt unless otherwise stated

3.2 Symbols

Void

3.3 Abbreviations

For the purposes of the present document, the following abbreviations apply:

EP	Elementary Procedure
HDLC	High-Level Data Link Control
RET	Remote Electrical Tilting

----------NEXT CHANGED SECTION----------

5 Services expected from signalling transport

RETAP requires an assured in-sequence delivery service from the signalling transport and notification if the assured in-sequence delivery service is no longer available.

5.1 Elementary procedure format

Layer 2 provides a full-duplex link for the transmission of RETAP messages.

There are two types of RETAP elementary procedures:

Class 1: Initiating messages are sent either from the primary to a secondary device, or from a secondary to the primary device, in order to initiate some action within the receiving device. The other device sends a response message completing the procedure.

Class 2: Initiating messages are sent either from the primary to a secondary device, or from a secondary to the primary device. No response message is expected.

All RETAP messages use the same basic format:

Table 5.1.1: Basic format for all RETAP messages

Elementary Procedure	Number of data octets	Data
1 octet	2 octets	~~78 octets~~Max Data Receive Length or Max Data Transmit Length.

~~NOTE: The default frame length of 78 octets is used unless another frame length is negotiated during the Address Assignment procedure (see [3]).~~

5.1.1 Initiating message

The data part of an initiating message may contain parameters as specified in section 6 of this TS.

----------NEXT CHANGED SECTION----------

6.5.3 Get Information

Table 6.5.3.1: Elementary procedure Get Information

Name: **GetInformation**				
Code: **0x05**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Mandatory**

Table 6.5.3.2: Initiating message parameters for Get Information

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Description:

On receipt of the initiating message the secondary device shall return the product number ProdNr and the serial number SerNr of the secondary device. If known, also the hardware version and the software version may be returned. The software version should indicate the version number of the currently executed software.

The parameters HWVersion and SWVersion in the response message refer to the version designators of the hardware and installed software of the secondary device. If the application is missing or no version number is found, then an empty string shall be returned as the version number.

The response message length shall be less than or equal to the minimum Secondary Payload Transmit Length [3].

Initiating message data format:

No data carried.

Response message data format:

<OK><LengthOctet><ProdNr><LengthOctet><SerNr><LengthOctet><HWVersion><LengthOctet><SWVersion>

Response message data format upon error:

<FAIL><ReturnCode1>…<ReturnCodeN>

Applicable return codes:

Busy, FlashError, OtherHardwareError, OtherSoftwareError, RAMError, UARTError. UnknownParameter, DownloadInProgress

----------NEXT CHANGED SECTION----------

6.5.8 Get Device Data

Table 6.5.8.1: Elementary procedure Get Device Data

Name: **GetDeviceData**

Code: **0x0F**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.5.8.2: Initiating message parameters for Get Device Data

Number	Length	Type	Description
~~i~~1	**1 octet**	**Hexadecimal**	**Field number; see annex B**

~~i = 1 … N~~

Description:

In this procedure the secondary device shall return the data stored in the field~~s~~ for configuration data specified by the field number~~s~~ in the procedure and listed in annex B of this TS. ~~The field numbers are not necessarily contiguous or ordered. For field numbers which are not supported by the secondary device those parameters are not returned.~~

Initiating message data format:

<field number> ~~1><field number 2>…<field number N>~~

Response message data format:

<OK>~~<field number 1>~~<data octets for field> ~~number 1><field number 2><data octets for field number 2> …<field number N><data octets for field number N>~~

Response message data format upon error:

<FAIL><ReturnCode1>...<ReturnCodeN>

Applicable return codes:

Busy, DeviceDisabled, EEPROMError, FlashError, OtherHardwareError, OtherSoftwareError, RAMError, UARTError, DownloadInProgress, UnknownParameter

6.5.9 Read User Data

Table 6.5.9.1: Elementary procedure Read User Data

Name: **ReadUserData**				
Code: **0x10**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.2.9.2: Initiating message parameters for Read User Data

Number	Length	Type	Description
1	**2 octets**	**Integer**	**Memory offset**
2	**1 octet**	**Integer**	**Number of octets to read**

NOTE: Number of octets to read shall be less or equal to Max Data Transmit Length − 1.

Description:

On receipt of the initiating message the secondary device sends back stored user specific data to the primary device.

Initiating message data format:

<OffsetLowOctet><OffsetHighOctet><NumberOfOctetsToRead>

Response message data format:

<OK><octet 1> ... <octet N>

Response message data format upon error:

<FAIL><ReturnCode1>…<ReturnCodeN>

Applicable return codes:

Busy, DataError, DeviceDisabled, EEPROMError, FlashError, OtherHardwareError, OtherSoftwareError, OutOfRange, RAMError, UARTError, DownloadInProgress

NOTE1: The return code OutOfRange is used if the given memory offset is outside the valid range.

6.5.10 Write User Data

Table 6.5.10.1: Elementary procedure Write User Data

Name: **WriteUserData**				
Code: **0x11**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.5.10.2: Initiating message parameters for Write User Data

Number	Length	Type	Description
1	**2 octets**	**Integer**	**Memory offset**
2	**1 octet**	**Integer**	**Number of octets to write**
3	**Message specific, given by parameter 2**	**Octets**	**Data to write**

NOTE: Number of octets to write shall be less or equal to Max Data Receive Length − 3.

Description:

On receipt of the initiating message the secondary device shall store user specific data in non-volatile memory. The user data is stored using the relative memory address offset given in the initiating message and starting with zero.

Initiating message data format:

<OffsetLowOctet><OffsetHighOctet><NumberOfOctetsToWrite><octet 1> ... <octet N>

Response message data format:

<OK>

Response message data format upon error:

<FAIL><ReturnCode1>…<ReturnCodeN>

Applicable return codes:

Busy, DeviceDisabled, EEPROMError, FlashError, OtherHardwareError, OtherSoftwareError, OutOfRange, RAMError, UARTError, DownloadInProgress

NOTE1: The return code OutOfRange is used if the needed data field is too long or if the given memory address is outside the valid address space.

6.5.11 Boot Mode Start

Table 6.5.11.1: Elementary procedure Boot Mode Start

Name: **BootModeStart**				
Code: **0x40**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **Yes**	Download boot mode: **Mandatory**

Table 6.5.11.2: Initiating message parameters for Boot Mode Start

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Description:

On receipt of this initiating message the software download process shall be initiated. Following transition to the boot state, the secondary device sends <OK>. . Non-volatile memory in the secondary device may be erased after this procedure is completed.

Initiating message data format:

No data carried.

Response message data format:

<OK>

Response message data format upon error:

<FAIL><ReturnCode 1>…<ReturnCode N>

Applicable return codes:

Busy, FlashEraseError, OtherHardwareError, OtherSoftwareError, RAMError, UARTError

6.5.12 Download Application

Table 6.2.12.1: Elementary procedure Download Application

Name: **DownloadApplication**				
Code: **0x41**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **Yes**	Download boot mode: **Mandatory**

Table 6.2.12.2: Initiating message parameters for Download Application

Number	Length	Type	Description
None	≤ Max Data Receive Length ~~**Vendor specific**~~	**Vendor specific**	**Software data**

Description:

Repeated use of this elementary procedure transfers software data from the primary device to the secondary device.

Initiating message data format:

<octet 1><octet 2>…<octet N>

Response message data format:

<OK>

Response message data format upon error:

<FAIL><ReturnCode 1>…<ReturnCode N>

Applicable return codes:

Busy, ChecksumError, DataError, EEPROMError, FlashEraseError, FlashError, OtherHardwareError, OtherSoftwareError, RAMError, UARTError, UnknownProcedure, InvalidSoftware, TooMuchData

NOTE1: UnknownProcedure may not be returned in boot mode.

----------NEXT CHANGED SECTION----------

6.6.2 Send Configuration Data

Table 6.6.2.1: Elementary procedure Send Configuration Data

Name: **SendConfigurationData**				
Code: **0x32**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.6.2.2: Initiating message parameters for Send Configuration Data

Number	Length	Type	Description
1	**≤ Max Data Transmit Length~~Vendor specific; Maximum of 70 octets~~**	**Vendor specific**	**Configuration data**

Description:

On receipt of the initiating message the secondary device shall store the provided vendor and antenna specific configuration data for the relationship between the movement of the drive system and the beam tilt position of the antenna.

If the configuration data exceeds Max Data Transmit Length~~70 octets~~, the data shall be split into a number of Max Data Transmit Length ~~70 octet~~ segments and one final segment with whatever is left. The primary device transmits the segments in order. The layer 2 sequence numbers guarantee that no segment will be lost or received out of order.

Initiating message data format:

<DataOctet1>…<DataOctetN>

Response message data format:

<OK>

Response message data format upon error:

<FAIL><ReturnCode1>…<ReturnCodeN>

Applicable return codes:

tbd

----------NEXT CHANGED SECTION----------

6.7.5 Antenna Get Data

Table 6.7.5.1: Elementary procedure Antenna Get Data

Name: **AntennaGetData**				
Code: **0x84**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.7.5.2: Initiating message parameters for Antenna Get Data

Number	Length	Type	Description
1	**1 octet**	**Integer**	**Antenna number**
~~i~~2	**1 octet**	**Hexadecimal**	**Field number to read; see annex B**

~~**i = 1 … N**~~

Description:

On receipt of the initiating message the secondary device shall return the data stored for the addressed antenna in the field~~s~~ for configuration data specified by the field number~~s~~ in the initiating message and listed in annex B of this TS. ~~The field numbers are not necessarily contiguous or ordered. For field numbers which are not supported by the secondary device for the addressed antenna no data is returned for that field.~~

Initiating message data format:

<antenna number><field number ~~1~~>~~<field number 2> …<field number N>~~

Response message data format:

<antenna number><OK>~~<field number 1>~~<data octets for field ~~number 1~~>~~<field number 2><data octets for field number 2> … <field number N><data octets for field number N>~~

Response message data format upon error:

<antenna number><FAIL><ReturnCode1>...<ReturnCodeN>

Applicable return codes:

Busy, DataError, DeviceDisabled, EEPROMError, FlashError, OtherHardwareError, OtherSoftewareError, RAMError, UARTError, UnknownProcedure, UnknownAntennaNumber, DownloadInProgress, UnknownParameter

3GPP TSG-RAN3 Meeting #45
Shin-Yokohama, Japan, 15th - 19th November 2004

Tdoc ⌘ R3-041531

CR-Form-v7.1

CHANGE REQUEST

⌘ **25.463** **CR** **17** ⌘ **rev** **-** ⌘ Current version: **6.0.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ ☐ ME ☐ Radio Access Network ☒ Core Network ☐

Title:	⌘	Definition of response time in the appication layer.
Source:	⌘	RAN3
Work item code:	⌘	RANimp-TiltAnt
Date:	⌘	10/11/2004
Category:	⌘	**F**
Release:	⌘	Rel-6

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

Ph2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*
Rel-7	*(Release 7)*

Reason for change:	⌘	Response time requirements are stated, but no spefication on measurement points is given.
Summary of change:	⌘	Introduction of measurement points.
Consequences if not approved:	⌘	Response time requirements cannot be verified.

Clauses affected:	⌘	5.1.2			
Other specs affected:	⌘	**Y**	**N**		
			X	Other core specifications	⌘
			X	Test specifications	
			X	O&M Specifications	
Other comments:	⌘				

How to create CRs using this form:
Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm.
Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.

2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be

downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

5.1.2 Response message

Elementary procedures shall, unless otherwise specified, provide a response message within one second. The response time is measured from the time the message frame was received by the transport layer to the time the response message is ready for transmit by the transport layer.

If the elementary procedure requested by the initiating message was successfully executed, the response message data part from a single-antenna device shall be <OK>. Additional information may follow in the data part. The response message data part from a multi-antenna device starts with the antenna number followed by <OK> and optional additional information.

If the elementary procedure requested by the initiating message was not successfully executed, the response message data part from a single-antenna device shall within a default period of 1 second unless otherwise specified, be <FAIL>

The following octets may contain additional return codes which describe why the execution of the requested procedure failed. The response message data part form a multi-antenna device starts with the antenna number followed by <OK> and optional additional return codes which describe why the execution of the requested procedure failed.

Return codes marked with an X in the Alarm column of annex A in this TS are used to report operating conditions in alarm procedures (see sections 6.6.5 and 6.7.6 for details).

In some situations an initiating message can cause a change of operating conditions, for instance a SetTilt procedure might cause a RET device to discover that an adjuster is jammed or that a previously jammed adjuster works normally again. In these cases an alarm procedure reporting the change of operating conditions shall be used in addition to the regular <OK> or <FAIL> response message.

A complete annotated table of all return codes with their corresponding hexadecimal numbers is provided in annex A of this TS.

3GPP TSG-RAN3 Meeting #45
Shin-Yokohama, Japan, 15th - 19th November 2004

Tdoc ⌘ R3-041679

CR-Form-v7.1

CHANGE REQUEST

⌘ **25.463** **CR** **18** ⌘ **rev** **2** ⌘ Current version: **6.0.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ ☐ ME ☐ Radio Access Network ☒ Core Network ☐

Title:	⌘	Redefinition of the Elementary Procedures GetDeviceData and SetDeviceData
Source:	⌘	RAN3
Work item code:	⌘	RANimp-TiltAnt
Date:	⌘	18/11/2004
Category:	⌘	**F**
Release:	⌘	Rel-6

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:
Ph2 (GSM Phase 2)
R96 (Release 1996)
R97 (Release 1997)
R98 (Release 1998)
R99 (Release 1999)
Rel-4 (Release 4)
Rel-5 (Release 5)
Rel-6 (Release 6)
Rel-7 (Release 7)

Reason for change:	⌘	The Elementary Procedures GetDeviceData and SetDeviceData are defined as common procedures. The Elementary Procedures AntennaGetDeviceData and AntennaSetDeviceData area defined as multi-antenna procedures.
Summary of change:	⌘	Redefine to Elementary Procedures GetDeviceData and SetDeviceData as single-antenna procedures.
Consequences if not approved:	⌘	Unclear use of the Procedures GetDeviceData and SetDeviceData and respectively AntennaGetDeviceData and AntennaSetDeviceData.

Clauses affected:	⌘	6.3, 6.5.7, 6.5.8, 6.6, and Annex B

		Y	N		
Other specs affected:	⌘		X	Other core specifications	⌘
			X	Test specifications	
			X	O&M Specifications	

Other comments: ⌘

How to create CRs using this form:
Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm.
Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.

2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

6.3 Overview of elementary procedures

The set of elementary procedures for RET antenna control provides procedure-oriented instructions. An overview of the procedures is given in annex D. Table 6.3.1 lists all common elementary procedures described in section 6.5. Table 6.3.2 lists all elementary procedures specific for single-antenna device types described in section 6.6. Table 6.3.3 lists all elementary procedures specific for multi-antenna device types described in section 6.7. Section 6.4 describes how to interpret the elementary procedure definitions in sections 6.5 to 6.7.

Some elementary procedures shall be performed in sequence as described in Annex C for the software download.

Table 6.3.1: Common elementary procedure set for all device types

Command	Requirement	Comment
Reset Software	mandatory	
Get Error Status	mandatory	
Get Information	mandatory	
Clear Active Alarms	mandatory	
Alarm Subscribe	mandatory	
Read User Data	mandatory	
Write User Data	mandatory	
Self Test	mandatory	
~~Set Device Data~~	~~mandatory~~	
~~Get Device Data~~	~~mandatory~~	
Boot Mode Start	optional	This procedure is mandatory if the software download feature is supported.
Download Application	optional	This procedure is mandatory if the software download feature is supported.
Download End	optional	This procedure is mandatory if the software download feature is supported.

Table 6.3.2: Elementary procedure set for singe-antenna device type

Command	Requirement	Comment
Calibrate	mandatory	
Send Configuration Data	mandatory	
Set Tilt	mandatory	
Get Tilt	mandatory	
Alarm	mandatory	
Set Device Data	mandatory	
Get Device Data	mandatory	

Table 6.3.3: Elementary procedure set for multiple-antenna device type

Command	Requirement	Comment
Antenna Calibrate	mandatory	

Antenna Set Tilt	mandatory	
Antenna Get Tilt	mandatory	
Antenna Set Device Data	mandatory	
Antenna Get Device Data	mandatory	
Antenna Alarm	mandatory	
Antenna Clear Active Alarms	mandatory	
Antenna Get Error Status	mandatory	
Antenna Get Number Of Antennas	mandatory	

----------NEXT CHANGED SECTION----------

6.5 Common elementary procedures

----------NEXT CHANGED SECTION----------

6.5.6 Self Test

Table 6.5.6.1: Elementary procedure Self Test

Name: **SelfTest**				
Code: **0x0A**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.5.6.2: Initiating message parameters for Self Test

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Description:

On receipt of the initiating message the secondary device executes a test procedure which may include a check of physical and processor functions. The specific tests to be performed are implementation specific, and may include the movement of the adjuster up to <FFS> degrees.

The response message of the secondary device on the procedure provides information on detected faults or, if no fault is detected, with confidence that the operation of the device is normal in all respects.

During the test the operational parameters of the device shall not change beyond operationally acceptable limits and on completion all parameters shall be returned to their initial values.

In the normal response message, in which the self test was executed successfully, the return codes are set to report possible detected functional errors during the self test. If no errors are detected, this shall be signalled by no return codes following <OK>.

In the case of an error response message, the self test could not be executed and the return codes relate to the inability of the device to perform the requested self-test operation.

Initiating message data format:

No data carried.

Response message data format:

<OK><ReturnCode1>…<ReturnCodeN>

Response message data format upon error:

<FAIL><ReturnCode1>…<ReturnCodeN>

Applicable return codes:

ActuatorDetectionFail, ActuatorJamPermanent, ActuatorJamTemporary, Busy, DataError, DeviceDisabled, EEPROMError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, PositionLost, RAMError, UARTError, WorkingSoftwareMissing, DownloadInProgress.

NOTE1: Only Busy, DataError, DeviceDisabled, EEPROMError, FlashError, OtherHardwareError, OtherSoftwareError, RAMError, UARTError, WorkingSoftwareMissing, DownloadInProgress may be return codes in the fail response message.

6.5.7 ~~6.5.7~~ Void

~~Set Device Data~~

~~Table 6.5.7.1: Elementary procedure Set Device Data~~

~~Name:~~ **~~SetDeviceData~~**				
~~Code:~~ **~~0x0E~~**	~~Issued by:~~ **~~Primary device~~**	~~Procedure class:~~ **~~1~~**	~~Download operation:~~ **~~No~~**	~~Download boot mode:~~ **~~Optional~~**

~~Table 6.5.7.2: Initiating message parameters for Set Device Data~~

~~Number~~	~~Length~~	~~Type~~	~~Description~~
~~1~~	**~~1 octet~~**	**~~Hexadecimal~~**	**~~Field number, see annex B~~**
~~2~~	**~~See annex B~~**	**~~See annex B~~**	**~~Data to write~~**

Description:

On receipt of the initiating message the secondary device should write the data given in the parameters of the initiating message into the fields optionally provided for configuration data and listed in annex B of this TS. If an attempt is made to write to fields which are designated as read only, the return code *ReadOnly* is returned and the data for those fields is ignored. If an attempt is made to write to fields which are not supported by the device the return code *UnknownParameter* is returned and the data for those fields is ignored.

Initiating message data format:

<field number><data octets for field>

Response message data format:

<OK>

Response message data format upon error:

<FAIL><ReturnCode1>...<ReturnCodeN>

Applicable return codes:

Busy, DataError, DeviceDisabled, EEPROMError, FlashError, FlashEraseError, OtherHardwareError, OtherSoftwareError, OutOfRange, RAMError, UARTError, ReadOnly, UnknownParameter, DownloadInProgress

6.5.8 Void Get Device Data

Table 6.5.8.1: Elementary procedure Get Device Data

Name: GetDeviceData				
Code: 0x0F	Issued by: Primary device	Procedure class: 1	Download operation: No	Download boot mode: Optional

~~Table 6.5.8.2: Initiating message parameters for Get Device Data~~

~~Number~~	~~Length~~	~~Type~~	~~Description~~
~~i~~	~~1 octet~~	~~Hexadecimal~~	~~Field number; see annex B~~

~~i = 1 … N~~

~~Description:~~

~~In this procedure the secondary device shall return the data stored in the fields for configuration data specified by the field numbers in the procedure and listed in annex B of this TS. The field numbers are not necessarily contiguous or ordered. For field numbers which are not supported by the secondary device those parameters are not returned.~~

~~Initiating message data format:~~

~~<field number 1><field number 2>…<field number N>~~

~~Response message data format:~~

~~<OK><field number 1><data octets for field number 1><field number 2><data octets for field number 2> …<field number N><data octets for field number N>~~

~~Response message data format upon error:~~

~~<FAIL><ReturnCode1>...<ReturnCodeN>~~

~~Applicable return codes:~~

~~Busy, DeviceDisabled, EEPROMError, FlashError, OtherHardwareError, OtherSoftwareError, RAMError, UARTError, DownloadInProgress~~

6.5.9 Read User Data

Table 6.5.9.1: Elementary procedure Read User Data

Name: **ReadUserData**				
Code:	Issued by:	Procedure class:	Download operation:	Download boot mode:

0x10	**Primary device**	**1**	**No**	**Optional**

----------NEXT CHANGED SECTION----------

6.6 Single-antenna elementary procedures

6.6.1 Calibrate

Table 6.6.1.1: Elementary procedure Calibrate

Name: **Calibrate**				
Code: **0x31**	Issued by: **Primary Device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.6.1.2: Initiating message parameters for Calibrate

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Description:

On receipt of the initiating message the secondary device shall perform a calibration of the RET antenna where the actuator is driven through its whole tilt range.

The response time to this Calibrate procedure shall be less than 4 minutes.

Initiating message data format:

No data carried.

Response message data format:

<OK>

Response message data format upon error:

<FAIL><ReturnCode1>…<ReturnCodeN>

Applicable return codes:

ActuatorDetectionFail, ActuatorJamPermanent, ActuatorJamTemporary, Busy, DeviceDisabled, EEPROMError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, PositionLost, RAMError, UARTError,UnknownProcedure, DownloadInProgress

----------NEXT CHANGED SECTION----------

6.6.x Set Device Data

Table 6.6.x.x: Elementary procedure Set Device Data

Name: **SetDeviceData**				
Code: **0x0E**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.6.x.x: Initiating message parameters for Set Device Data

Number	Length	Type	Description
1	**1 octet**	**Hexadecimal**	**Field number, see annex B**
2	**See annex B**	**See annex B**	**Data to write**

Description:

On receipt of the initiating message the secondary device should write the data given in the parameters of the initiating message into the fields optionally provided for configuration data and listed in annex B of this TS. If an attempt is made to write to fields which are designated as read only, the return code *ReadOnly* is returned and the data for those fields is ignored. If an attempt is made to write to fields which are not supported by the device the return code *UnknownParameter* is returned and the data for those fields is ignored.

Initiating message data format:

<field number><data octets for field>

Response message data format:

<OK>

Response message data format upon error:

<FAIL><ReturnCode1>...<ReturnCodeN>

Applicable return codes:

Busy, DataError, DeviceDisabled, EEPROMError, FlashError, FlashEraseError, OtherHardwareError, OtherSoftwareError, OutOfRange, RAMError, UARTError, ReadOnly, UnknownParameter, DownloadInProgress

6.6.x Get Device Data

Table 6.6.x.x: Elementary procedure Get Device Data

Name: **GetDeviceData**				
Code: **0x0F**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.6.x.x: Initiating message parameters for Get Device Data

Number	Length	Type	Description

i	**1 octet**	**Hexadecimal**	**Field number; see annex B**

i = 1 … N

Description:

In this procedure the secondary device shall return the data stored in the fields for configuration data specified by the field numbers in the procedure and listed in annex B of this TS. The field numbers are not necessarily contiguous or ordered. For field numbers which are not supported by the secondary device those parameters are not returned.

Initiating message data format:

<field number 1><field number 2>…<field number N>

Response message data format:

<OK><field number 1><data octets for field number 1><field number 2><data octets for field number 2> …<field number N><data octets for field number N>

Response message data format upon error:

<FAIL><ReturnCode1>...<ReturnCodeN>

Applicable return codes:

Busy, DeviceDisabled, EEPROMError, FlashError, OtherHardwareError, OtherSoftwareError, RAMError, UARTError, DownloadInProgress

----------NEXT CHANGED SECTION----------

6.7.4 Antenna Set Device Data

Table 6.7.4.1: Elementary procedure Antenna Set Device Data

Name: **AntennaSetDeviceData**				
Code: **0x83**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.7.4.2: Initiating message parameters for Antenna Set Device Data

Number	Length	Type	Description
1	**1 octet**	**Integer**	**Antenna number**
2	**1 octet**	**Hexadecimal**	**Field number; see annex B**
3	**See annex B**	**See annex B**	**Data to write**

Description:

On receipt of the initiating message the secondary device should write the provided data for the antenna addressed by the antenna number into the fields optionally provided for configuration data and listed in annex B of this TS. If an attempt is made to write to fields which are not supported by a particular device no error is returned but the data for those fields is ignored. If an attempt is made to write to fields which are not supported for the addressed antenna the return code UnknownParameter is returned and the data for those fields is ignored.

Initiating message data format:

<antenna number><field number><data octets for field>

Response message data format:

<antenna number><OK>

Response message data format upon error:

<antenna number><FAIL><ReturnCode1>...<ReturnCodeN>

Applicable return codes:

Busy, DataError, DeviceDisabled, EEPROMError, FlasheError, OtherHardwareError, OtherSoftwareError, OutOfRange, RAMError, UARTError, UnknownProcedure, ReadOnly, UnknownParameter, UnknownAntennaNumber, DownloadInProgress

6.7.5 Antenna Get Device Data

Table 6.7.5.1: Elementary procedure Antenna Get Device Data

Name: **AntennaGetDeviceData**				
Code: **0x84**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.7.5.2: Initiating message parameters for Antenna Get Device Data

Number	Length	Type	Description
1	**1 octet**	**Integer**	**Antenna number**
–i	**1 octet**	**Hexadecimal**	**Field number to read; see annex B**

i = 1 … N

Description:

On receipt of the initiating message the secondary device shall return the data stored for the addressed antenna in the fields for configuration data specified by the field numbers in the initiating message and listed in annex B of this TS. The field numbers are not necessarily contiguous or ordered. For field numbers which are not supported by the secondary device for the addressed antenna no data is returned for that field.

Initiating message data format:

<antenna number><field number 1><field number 2> …<field number N>

Response message data format:

<antenna number><OK><field number 1><data octets for field number 1><field number 2><data octets for field number 2> … <field number N><data octets for field number N>

Response message data format upon error:

<antenna number><FAIL><ReturnCode1>...<ReturnCodeN>

Applicable return codes:

Busy, DataError, DeviceDisabled, EEPROMError, FlashError, OtherHardwareError, OtherSoftewareError, RAMError, UARTError, UnknownProcedure, UnknownAntennaNumber, DownloadInProgress

----------NEXT CHANGED SECTION----------

Annex B (normative): Assigned fields for additional data

The following standard fields have no operational impact and are used by the procedures SetDeviceData and GetDeviceData. Little-endian order is used for storage of multiple-octet numbers. Where ASCII variables are shorter than the assigned field lengths the characters are right aligned and leading blanks are filled with null characters (0x00).

Table B.1: Assigned fields for additional data

Field No.	Length (octets)	Format	Description
0x01	15	ASCII	Antenna model number
0x02	17	ASCII	Antenna serial number
0x03	2	16-bit unsigned	Antenna frequency band(s): see below
0x04	~~3~~1	~~3~~ 1 x 8-bit unsigned	Beamwidth for each band in frequency order (deg) (example 800/900MHz, 1800/1900MHz, 2100MHz)
0x05	3	3 x 8-bit unsigned	Gain for each band in frequency order (dB/10) (example 800/900MHz, 1800/1900MHz, 2100MHz)
0x06	2	16-bit signed	Maximum supported tilt (degrees/10), Format as in section 6.11
0x07	2	16-bit signed	Minimum supported tilt (degrees/10), Format as in section 6.11
0x21	6	ASCII	Installation date
0x22	5	ASCII	Installer's ID
0x23	12	ASCII	Base station ID
0x24	4	ASCII	Sector ID
0x25	2	16-bit unsigned	Antenna bearing
0x26	~~1~~2	~~8~~16-bit signed	Installed mechanical tilt (degrees/10)

Table B.2: Coding for antenna frequency bands in field 0x03

	Field 0x03
Bit No	Frequency band(MHz)
1	800
2	900
3	1500
4	1800
5	1900

6	2100
7 and above	Reserved

Examples of frequency bands: 0000 0000 0001 0000 = 1800MHz,

0000 0000 0001 1100 = 1800, 1900 and 2100MHz

NOTE: Field numbers 0x01, 0x02, and 0x21 to 0x26 in Table B:1 are common for multi-antenna device antennas. These fields may be addressed through any antenna number procedure.

3GPP TSG-RAN3 Meeting #45 ***Tdoc ⌘ R3-041626***
Shin-Yokohama, Japan, 15th - 19th November 2004

CR-Form-v7.1

CHANGE REQUEST

⌘ **25.463** **CR** **001** ⌘ **rev** **2** ⌘ Current version: **6.0.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ [ ] ME [ ] Radio Access Network [X] Core Network [ ]

Title:	⌘	Reduction of risk of accidentional erasure of RET application software
Source:	⌘	RAN3
Work item code:	⌘	RANimp-TiltAnt
Date:	⌘	15/11/2004
Category:	⌘	**F**
Release:	⌘	Rel-6

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

Ph2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*
Rel-7	*(Release 7)*

Reason for change:	⌘	The imminent risk of accidental erasure of RET application software
Summary of change:	⌘	The erasure of flash memory shall not be done at Boot Mode Start but when Download Application is received, thus when the validity of the application software can be done. Rev1: The first added sentence in Annex C was reworded for simplicity and clarity.
Consequences if not approved:	⌘	Higher risk of RET application software erasure when a wrong application software loadfile is attempted to be downloaded.

Clauses affected:	⌘	6.5.11 and Annex C

	Y	N		
Other specs affected: ⌘		X	Other core specifications	⌘
		X	Test specifications	
		X	O&M Specifications	

Other comments: ⌘

How to create CRs using this form:
Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm.
Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.

2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

6.5.11 Boot Mode Start

Table 6.5.11.1: Elementary procedure Boot Mode Start

Name: **BootModeStart**				
Code: **0x40**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **Yes**	Download boot mode: **Mandatory**

Table 6.5.11.2: Initiating message parameters for Boot Mode Start

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Description:

On receipt of this initiating message the software download process shall be initiated. Following transition to the ~~boot state~~BootMode state, the secondary device sends <OK>. ~~. Non-volatile memory in the secondary device may be erased after this procedure is completed.~~

Initiating message data format:

No data carried.

Response message data format:

<OK>

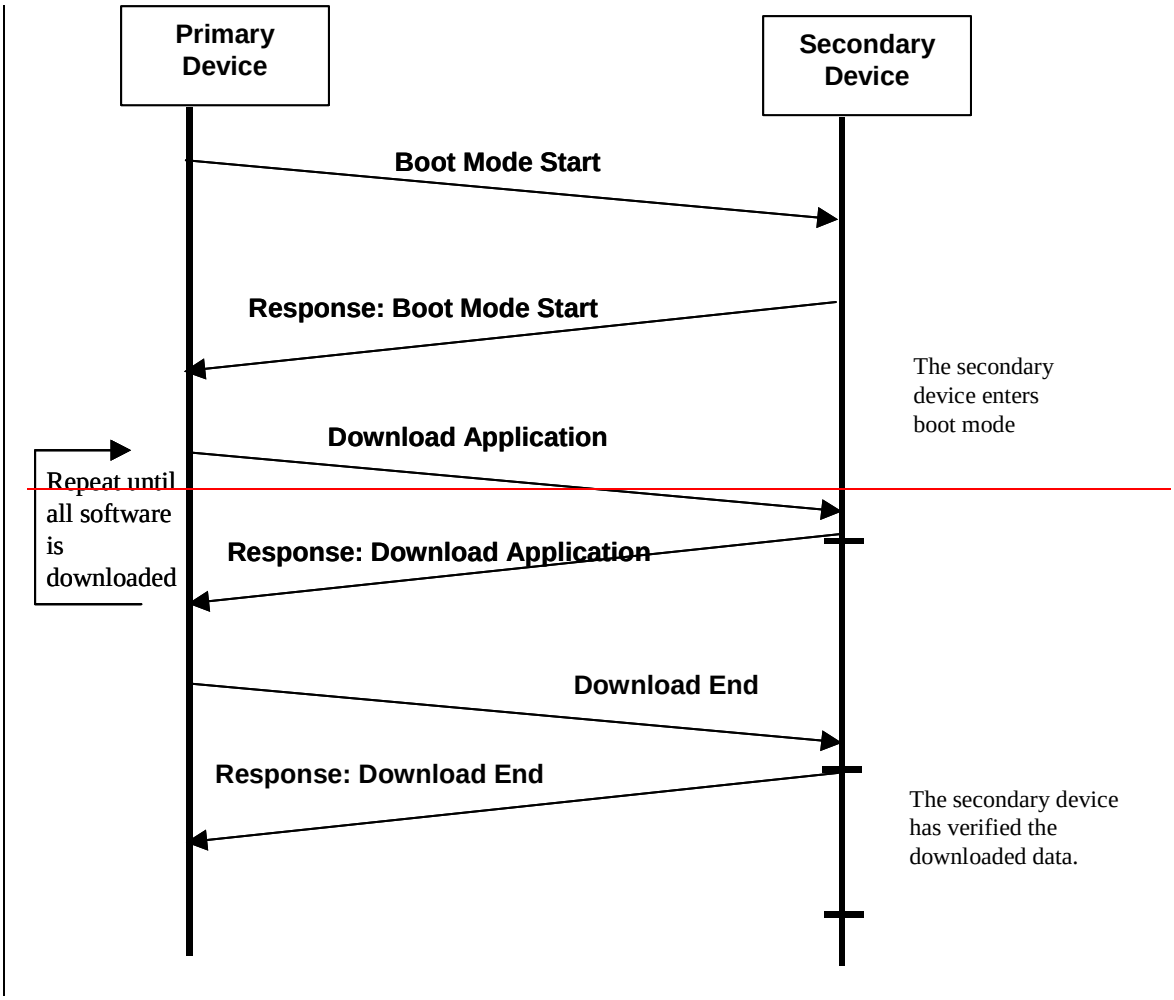
Response message data format upon error:

<FAIL><ReturnCode 1>…<ReturnCode N>

Applicable return codes:

Busy, FlashEraseError, OtherHardwareError, OtherSoftwareError, RAMError, UARTError

Annex C (normative): Procedure sequence for download of software to a secondary device



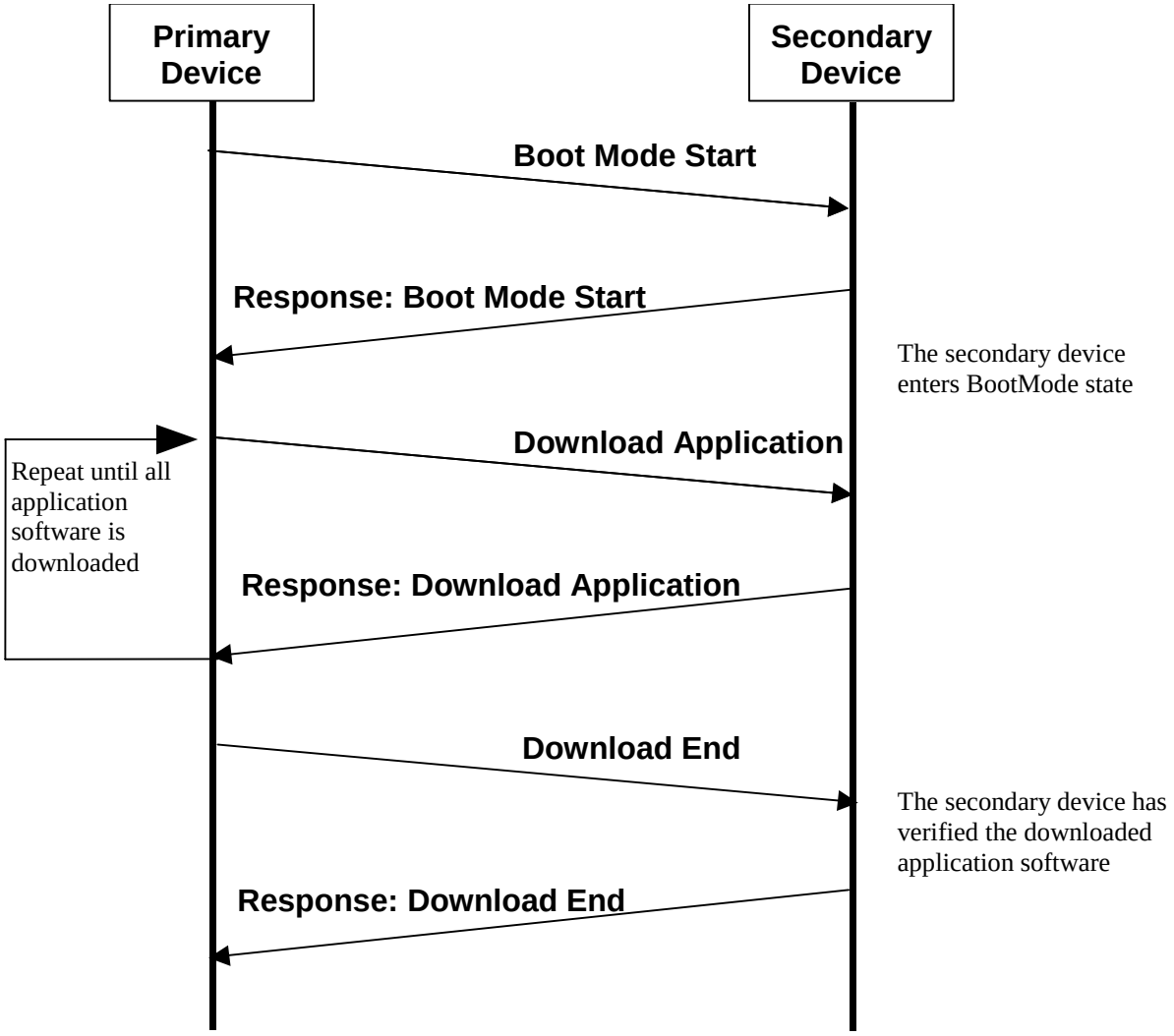


Figure C.1: Procedure sequence for Software Download.

The erasure of the secondary device application software shall not be done before the reception of the Download Application message. The data content of the Download Application message is implementation specific but it is recommended to support a application software validity feature that shall minimise the risk of downloading faulty or invalid application software.

3GPP TSG-RAN3 Meeting #43 **Tdoc ⌘041423**
Shin-Yokohama, Japan, 15th - 19th November 2004

CR-Form-v7.1

CHANGE REQUEST

⌘ **25.463** **CR** **002** ⌘ **rev** **-** ⌘ Current version: **6.0.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ ☐ ME ☐ Radio Access Network ☒ Core Network ☐

Title:	⌘	Clarification of allowed tilt operation during self test
Source:	⌘	RAN3
Work item code:	⌘	RANimp-TiltAnt
Date:	⌘	03/11/2004
Category:	⌘	**F**
Release:	⌘	Rel-6

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

Ph2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*
Rel-7	*(Release 7)*

Reason for change:	⌘	Removal of <FFS>
Summary of change:	⌘	Clarification of allowed tilt operation during RET self test
Consequences if not approved:	⌘	Risk for poor network performance during RET selftests

Clauses affected:	⌘	6.5.6			
Other specs affected:	⌘	**Y**	**N**		
			X	Other core specifications	⌘
			X	Test specifications	
			X	O&M Specifications	
Other comments:	⌘				

How to create CRs using this form:
Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm.
Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.

2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

6.5.6 Self Test

Table 6.5.6.1: Elementary procedure Self Test

Name: **SelfTest**				
Code: **0x0A**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.5.6.2: Initiating message parameters for Self Test

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Description:

On receipt of the initiating message the secondary device executes a test procedure which may include a check of physical and processor functions. The specific tests to be performed are implementation specific, and may include the movement of the adjuster, which shall not exceed ~~up to~~ +-5% of total available tilting range~~<FFS> degrees~~ starting from the current adjuster position.

The response message of the secondary device on the procedure provides information on detected faults or, if no fault is detected, with confidence that the operation of the device is normal in all respects.

During the test the operational parameters of the device shall not change beyond operationally acceptable limits and on completion all parameters shall be returned to their initial values.

In the normal response message, in which the self test was executed successfully, the return codes are set to report possible detected functional errors during the self test. If no errors are detected, this shall be signalled by no return codes following <OK>.

In the case of an error response message, the self test could not be executed and the return codes relate to the inability of the device to perform the requested self-test operation.

Initiating message data format:

No data carried.

Response message data format:

<OK><ReturnCode1>…<ReturnCodeN>

Response message data format upon error:

<FAIL><ReturnCode1>…<ReturnCodeN>

Applicable return codes:

ActuatorDetectionFail, ActuatorJamPermanent, ActuatorJamTemporary, Busy, DataError, DeviceDisabled, EEPROMError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, PositionLost, RAMError, UARTError, WorkingSoftwareMissing, DownloadInProgress.

NOTE1: Only Busy, DataError, DeviceDisabled, EEPROMError, FlashError, OtherHardwareError, OtherSoftwareError, RAMError, UARTError, WorkingSoftwareMissing, DownloadInProgress may be return codes in the fail response message.

3GPP TSG-RAN3 Meeting #45
Yokohama, Japan, 15th – 19th November 2004

Tdoc ⌘ R3-041424

CR-Form-v7.1

CHANGE REQUEST

⌘ **25.463** **CR** **003** ⌘ **rev** **-** ⌘ Current version: **6.0.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ ☐ ME ☐ Radio Access Network ☒ Core Network ☐

Title:	⌘	State model for RET device
Source:	⌘	RAN3
Work item code:	⌘	RANimp-TiltAnt
Date:	⌘	02/11/2004
Category:	⌘	**F**
Release:	⌘	Rel-6

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

Ph2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*
Rel-7	*(Release 7)*

Reason for change:	⌘	RET device state model is FFS.
Summary of change:	⌘	State model introduced, according to the information in Tdoc R3-041393.
Consequences if not approved:	⌘	Incomplete specification.

Clauses affected:	⌘	6.1

	Y	N			
Other specs affected:	X		Other core specifications	⌘	TS 25.460 CR2
		X	Test specifications		
		X	O&M Specifications		

Other comments: ⌘

6 Control elementary procedures

6.1 State Model

~~The state model describing the secondary device is FFS.~~

~~Figure 6.1: State Model for Secondary Device~~

The state model describing the RET device is shown in figure 6.1 with procedures written in *italic.*

The relation to the connection state model for layer 2 can be found in [3].

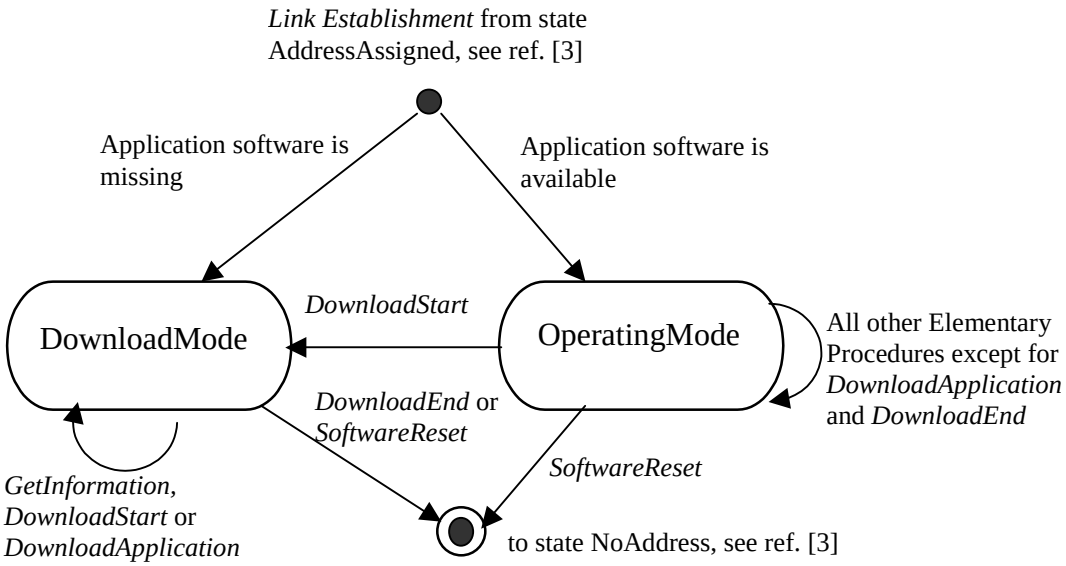


Figure 6.1: State Model for the RET Device

If an application software is not missing the RET device enters the state OperatingMode.

If an application software is missing, the RET device enters the state DownloadMode. In this state only software download functionality is supported in order to restore the application software.

The primary device will be notified that the RET device has entered the state DownloadMode when a procedure which only is supported in the state OperatingMode fails with the return code WorkingSoftwareMissing.

If no software download functionality is supported, then only the state OperatingMode for the RET device is supported.

3GPP TSG-RAN3 Meeting #45
Yokohama, Japan, 15th – 19th November 2004

Tdoc ⌘ R3-041451

CR-Form-v7.1

CHANGE REQUEST

⌘ **25.463** **CR** **004** ⌘ **rev** **-** ⌘ Current version: **6.0.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ ☐ ME ☐ Radio Access Network ☒ Core Network ☐

Title:	⌘	Corrections and editorial changes to 25.463 according to RAN3#44
Source:	⌘	RAN3
Work item code:	⌘	RANimp-TiltAnt
Date:	⌘	02/11/2004
Category:	⌘	**F**
Release:	⌘	Rel-6

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

Ph2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*
Rel-7	*(Release 7)*

Reason for change:	⌘	Corrections and editorial changes for a cleaner and more precise specification.
Summary of change:	⌘	Removal of section for Symbols and Re-structuring of section 3. Removal of definition for "Device tpye". Addition of definition for "Signed integer". Addition of new section for Integer Representation. Correction of description part for EP Alarm Subscribe. Several changes of parameter types (e.g. hexadecimal to 8 bit unsigned). Re-wording of description of parameter "tilt value" for clarification. Correction of description of EP Alarm to be more precise. Several editorial corrections. Addition of three missing EPs for Multi-Antenna-Devices in table D.1.
Consequences if not approved:	⌘	Incomplete, unclear or wrong specification.

Clauses affected:	⌘	3, 4, 5.1.2, 6.3, 6.5.1, 6.5.5, 6.5.6, 6.5.7, 6.5.8, 6.5.9, 6.5.10, 6.5.11, 6.5.12, 6.6.3, 6.6.5, 6.7.1, 6.7.2, 6.7.3, 6.7.4, 6.7.5, 6.7.6, 6.7.7, 6.7.8, 6.7.9, Annex B, Annex C, Annex D

		Y	N		
Other specs affected:	⌘		X	Other core specifications	⌘
			X	Test specifications	
			X	O&M Specifications	

Other comments:	⌘	

How to create CRs using this form:
Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm. Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.

2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

********** Unchanged Parts omitted **********

3 Definitions~~, symbols~~ and abbreviations

3.1 Definitions

For the purposes of the present document, the following terms and definitions apply.

ASCII character: A character forming part of the International Reference Version of the 7-bit character set defined in ISO/IEC 646:1991

Calibrate: Exercise the antenna drive unit over its entire range of travel to ensure fault-free operation and synchronise the measured and actual beam tilt of the antenna

Configuration data: A stored table or function defining the relationship between the physical position of the drive and electrical beam-tilt

~~**Device type:** See section 4.7 in [3].~~

Elementary Procedure: The RETAP protocol consists of Elementary Procedures (EPs). An Elementary Procedure is a unit of interaction between the primary device (Node B) and the secondary devices (RET devices).

An EP consists of an initiating message and possibly a response message.

Two kinds of EPs are used:

- **Class 1**: Elementary Procedures with response (success or failure).
- **Class 2**: Elementary Procedures without response.

For **Class 1** EPs, the types of responses can be as follows:

Successful

- A signalling message explicitly indicates that the elementary procedure has been successfully completed with the receipt of the response.

Unsuccessful

- A signalling message explicitly indicates that the EP failed.

Class 2 EPs are considered always successful.

Little-endian: The order of transmission in which the least-significant octets of a multi-octet representation of a number are transmitted first. Little endian only applies to binary integer representations.

Return code: A 1-octet enumerated response message to an initiating message.

Tilt (also downtilt, tilt angle, beamtilt): The elevation angle between the direction orthogonal to the antenna element axis and the maximum of its main beam in the elevation plane. A positive electrical tilt angle means that the antenna beam is directed below the direction orthogonal to the antenna axis. An antenna has separate values for electrical and mechanical tilt. The mechanical tilt is fixed by the geometry of the installation. In this TS the tilt referred to is always the electrical tilt unless otherwise stated

~~3.2 Symbols~~

~~Void~~

3.2~~3~~ Abbreviations

For the purposes of the present document, the following abbreviations apply:

EP	Elementary Procedure
HDLC	High-Level Data Link Control
RET	Remote Electrical Tilting

********** Unchanged Parts omitted **********

4.3 Multi-antenna units

The RETAP elementary procedures are split into a single-antenna oriented part, a multi-antenna oriented part and a common part for both device types in order to support RET units controlling single- or multi-antenna devices. The RET unit responds, upon request, the number of antennas it controls. All multi-antenna oriented elementary procedures include a parameter stating which antenna the elementary procedure addresses. Antennas are numbered 1 and upwards.

4.4 Integer Representation

Multi-octet integer values are transmitted in little-endian order. Signed integers are represented as 2-complement values.

********** Unchanged Parts omitted **********

5.1.2 Response message

If the elementary procedure requested by the initiating message was successfully executed, the response message data part from a single-antenna device shall be <OK>. Additional information may follow in the data part. The response message data part from a multi-antenna device starts with the antenna number followed by <OK> and optional additional information.

If the elementary procedure requested by the initiating message was not successfully executed, the response message data part from a single-antenna device shall be <FAIL>. Following the initiating message, a response message is expected within a default period of 1 second unless otherwise specified~~, be <FAIL>~~.

The following octets may contain additional return codes which describe why the execution of the requested procedure failed. The response message data part ~~form~~ from a multi-antenna device starts with the antenna number followed by <~~OK~~FAIL> and optional additional return codes which describe why the execution of the requested procedure failed.

Return codes marked with an X in the Alarm column of annex A in this TS are used to report operating conditions in alarm procedures (see sections 6.6.5 and 6.7.6 for details).

In some situations an initiating message can cause a change of operating conditions, for instance a SetTilt procedure might cause a RET device to discover that an adjuster is jammed or that a previously jammed adjuster works normally again. In these cases an alarm procedure reporting the change of operating conditions shall be used in addition to the regular <OK> or <FAIL> response message.

A complete annotated table of all return codes with their corresponding hexadecimal numbers is provided in annex A of this TS.

********** Unchanged Parts omitted **********

6.3 Overview of elementary procedures

The set of elementary procedures for RET antenna control provides procedure-oriented instructions. An overview of the procedures is given in annex D. Table 6.3.1 lists all common elementary procedures described in section 6.5. Table 6.3.2 lists all elementary procedures specific for single-antenna device types described in section 6.6. Table 6.3.3 lists all elementary procedures specific for multi-antenna device types described in section 6.7. Section 6.4 describes how to interpret the elementary procedure definitions in sections 6.5 to 6.7.

Some elementary procedures shall be performed in sequence as described in Annex C for the software download.

Table 6.3.1: Common elementary procedure set for all device types

Command	Requirement	Comment
Reset Software	mandatory	
Get Error Status	mandatory	
Get Information	mandatory	
Clear Active Alarms	mandatory	
Alarm Subscribe	mandatory	
Read User Data	mandatory	
Write User Data	mandatory	
Self Test	mandatory	
Set Device Data	mandatory	
Get Device Data	mandatory	
Boot Mode Start	optional	This procedure is mandatory if the software download feature is supported.
Download Application	optional	This procedure is mandatory if the software download feature is supported.
Download End	optional	This procedure is mandatory if the software download feature is supported.

Table 6.3.2: Elementary procedure set for single-antenna device type

Command	Requirement	Comment
Calibrate	mandatory	
Send Configuration Data	mandatory	
Set Tilt	mandatory	
Get Tilt	mandatory	
Alarm	mandatory	

Table 6.3.3: Elementary procedure set for multiple-antenna device type

Command	Requirement	Comment
Antenna Calibrate	mandatory	
Antenna Set Tilt	mandatory	
Antenna Get Tilt	mandatory	

Antenna Set Data	mandatory	
Antenna Get Data	mandatory	
Antenna Alarm	mandatory	
Antenna Clear Active Alarms	mandatory	
Antenna Get Error Status	mandatory	
Antenna Get Number Of Antennas	mandatory	

********** Unchanged Parts omitted **********

6.5 Common elementary procedures

6.5.1 Reset Software

Table 6.5.1.1: Elementary procedure Reset Software

Name: **ResetSoftware**				
Code: **0x03**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **Yes**	Download boot mode: **Mandatory**

Table 6.5.1.2: Initiating message parameters for Reset Software

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Description:

On the receipt of the initiating message the secondary device shall set the HDLC address to the No-station address and place the device in the *NoAddress* state.

The device shall not execute the reset procedure before transport layer acknowledgement through sequence number update is received for the response. ~~(See annex C in [2]).~~

The secondary device shall not fail to reset for any reason.

Initiating message data format:

No data carried in the message.

Response message data format:

<OK>

Response message data format upon error:

No error accepted.

Applicable return codes:

<OK>

********** Unchanged Parts omitted **********

6.5.5 Alarm Subscribe

Table 6.5.5.1: Elementary procedure Alarm Subscribe

Name: **AlarmSubscribe**				
Code: **0x12**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Mandatory**

Table 6.5.5.2: Initiating pessage parameters for Alarm Subscribe

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Description:

On receipt of the initiating message the secondary device shall start ~~executing its normal operation code~~ reporting alarms to the primary device.

Initiating message data format:

No Data carried.

Response message data format:

<OK>

Response message data format upon error:

<FAIL><ReturnCode1>…<ReturnCodeN>

Applicable return codes:

Busy, DeviceDisabled, EEPROMError, FlashError, OtherSoftwareError, OtherHardwareError, RAMError, UARTError, UnknownParameter, DownloadInProgress~~Invalid Software~~

NOTE1: UnknownParameter is the response code used if any data is carried in the initiating message.

6.5.6 Self Test

Table 6.5.6.1: Elementary procedure Self Test

Name: **SelfTest**				
Code: **0x0A**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.5.6.2: Initiating message parameters for Self Test

Number	Length	Type	Description

None	**0 octets**	**None**	**No data carried**

Description:

On receipt of the initiating message the secondary device executes a test procedure which may include a check of physical and processor functions. -The specific tests to be performed are implementation specific, and may include the movement of the adjuster up to <FFS> degrees.

********** Unchanged Parts omitted **********

6.5.7 Set Device Data

Table 6.5.7.1: Elementary procedure Set Device Data

Name: **SetDeviceData**				
Code: **0x0E**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.5.7.2: Initiating message parameters for Set Device Data

Number	Length	Type	Description
1	**1 octet**	**~~Hexadecimal~~Unsigned integer**	**Field number, see annex B**
2	**See annex B**	**See annex B**	**Data to write**

Description:

On receipt of the initiating message the secondary device should write the data given in the parameters of the initiating message into the fields optionally provided for configuration data and listed in annex B of this TS. If an attempt is made to write to fields which are designated as read only, the return code *ReadOnly* is returned and the data for those fields is ignored. If an attempt is made to write to fields which are not supported by the device the return code *UnknownParameter* is returned and the data for those fields is ignored.

Initiating message data format:

<field number><data octets for field>

Response message data format:

<OK>

Response message data format upon error:

<FAIL><ReturnCode1>...<ReturnCodeN>

Applicable return codes:

Busy, DataError, DeviceDisabled, EEPROMError, FlashError, FlashEraseError, OtherHardwareError, OtherSoftwareError, OutOfRange, RAMError, UARTError, ReadOnly, UnknownParameter, DownloadInProgress

6.5.8 Get Device Data

Table 6.5.8.1: Elementary procedure Get Device Data

Name: **GetDeviceData**				
Code: **0x0F**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.5.8.2: Initiating message parameters for Get Device Data

Number	Length	Type	Description
i	**1 octet**	**~~Hexadecimal~~Unsigned integer**	**Field number; see annex B**

i = 1 … N

Description:

In this procedure the secondary device shall return the data stored in the fields for configuration data specified by the field numbers in the procedure and listed in annex B of this TS. The field numbers are not necessarily contiguous or ordered. For field numbers which are not supported by the secondary device those parameters are not returned.

Initiating message data format:

<field number 1><field number 2>…<field number N>

Response message data format:

<OK><field number 1><data octets for field number 1><field number 2><data octets for field number 2> …<field number N><data octets for field number N>

Response message data format upon error:

<FAIL><ReturnCode1>...<ReturnCodeN>

Applicable return codes:

Busy, DeviceDisabled, EEPROMError, FlashError, OtherHardwareError, OtherSoftwareError, RAMError, UARTError, DownloadInProgress

6.5.9 Read User Data

Table 6.5.9.1: Elementary procedure Read User Data

Name: **ReadUserData**				
Code: **0x10**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.2.9.2: Initiating message parameters for Read User Data

Number	Length	Type	Description
1	**2 octets**	**~~Integer~~Unsigned integer**	**Memory offset**
2	**1 octet**	**~~Integer~~ Unsigned**	**Number of octets to**

		integer	**read**

Description:

On receipt of the initiating message the secondary device sends back stored user specific data to the primary device.

Initiating message data format:

<OffsetLowOctet><OffsetHighOctet><NumberOfOctetsToRead>

Response message data format:

<OK><octet 1> ... <octet N>

Response message data format upon error:

<FAIL><ReturnCode1>…<ReturnCodeN>

Applicable return codes:

Busy, DataError, DeviceDisabled, EEPROMError, FlashError, OtherHardwareError, OtherSoftwareError, OutOfRange, RAMError, UARTError, DownloadInProgress

NOTE1: The return code OutOfRange is used if the given memory offset is outside the valid range.

6.5.10 Write User Data

Table 6.5.10.1: Elementary procedure Write User Data

Name: **WriteUserData**				
Code: **0x11**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.5.10.2: Initiating message parameters for Write User Data

Number	Length	Type	Description
1	**2 octets**	~~Integer~~Unsigned integer	**Memory offset**
2	**1 octet**	~~Integer~~Unsigned integer	**Number of octets to write**
3	**Message specific, given by parameter 2**	**Octets**	**Data to write**

Description:

On receipt of the initiating message the secondary device shall store user specific data in non-volatile memory. The user data is stored using the relative memory address offset given in the initiating message and starting with zero.

Initiating message data format:

<OffsetLowOctet><OffsetHighOctet><NumberOfOctetsToWrite><octet 1> ... <octet N>

Response message data format:

<OK>

Response message data format upon error:

<FAIL><ReturnCode1>…<ReturnCodeN>

Applicable return codes:

Busy, DeviceDisabled, EEPROMError, FlashError, OtherHardwareError, OtherSoftwareError, OutOfRange, RAMError, UARTError, DownloadInProgress

NOTE1: The return code OutOfRange is used if the needed data field is too long or if the given memory address is outside the valid address space.

6.5.11 Boot Mode Start

Table 6.5.11.1: Elementary procedure Boot Mode Start

Name: **BootModeStart**				
Code: **0x40**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **Yes**	Download boot mode: **Mandatory**

Table 6.5.11.2: Initiating message parameters for Boot Mode Start

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Description:

On receipt of this initiating message the software download process shall be initiated. Following transition to the boot state, the secondary device sends <OK>. Non-volatile memory in the secondary device may be erased after this procedure is completed.

Initiating message data format:

No data carried.

Response message data format:

<OK>

Response message data format upon error:

<FAIL><ReturnCode 1>…<ReturnCode N>

Applicable return codes:

Busy, FlashEraseError, OtherHardwareError, OtherSoftwareError, RAMError, UARTError

6.5.12 Download Application

Table 6.2.12.1: Elementary procedure Download Application

Name: **DownloadApplication**				
Code: **0x41**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **Yes**	Download boot mode: **Mandatory**

Table 6.2.12.2: Initiating message parameters for Download Application

Number	Length	Type	Description
None	**Vendor specific**	**Vendor specific**	**Software data**

Description:

This elementary procedure is used once or several times to transfer ~~Repeated use of this elementary procedure transfers~~ software data from the primary device to the secondary device.

Initiating message data format:

<octet 1><octet 2>…<octet N>

Response message data format:

<OK>

Response message data format upon error:

<FAIL><ReturnCode 1>…<ReturnCode N>

Applicable return codes:

Busy, ChecksumError, DataError, EEPROMError, FlashEraseError, FlashError, OtherHardwareError, OtherSoftwareError, RAMError, UARTError, UnknownProcedure, InvalidSoftware, TooMuchData

NOTE1: UnknownProcedure may not be returned in boot mode.

********** Unchanged Parts omitted **********

6.6.3 Set Tilt

Table 6.6.3.1: Elementary procedure Set Tilt

Name: **SetTilt**				
Code: **0x33**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.6.3.2: Initiating message parameters for Set Tilt

Number	Length	Type	Description
1	**2 octets**	**~~16 bit signed~~Signed integer~~ little-endian~~**	**Tilt value**

Description:

On receipt of the initiating message the secondary device shall set the electrical tilt in increments of 0.1°. The electrical tilt value describes the elevation angle between the direction orthogonal to the antenna element axis and the maximum of its main beam in the elevation plane. A positive electrical tilt angle means that the antenna beam is directed below the direction orthogonal to the antenna axis.

The secondary device shall respond to the initiating message in less than 2 minutes.

The value of parameter 1 is 10 times the tilt in degrees.~~The format of the provided tilt value is a 2-complement 16 bit signed number sent in little-endian order. Tilt values are given in 0.1° increments starting from zero, for example: Tilt 3.2° is 0x0020, Tilt – 3.2° is 0xFFE0.~~

Initiating message data format:

<TiltLowOctet><TiltHighOctet>

Response message data format:

<OK>

Response message data format upon error:

<FAIL><ReturnCode1>…<ReturnCodeN>

Applicable return codes:

ActuatorDetectionFail, ActuatorJamPermanent, ActuatorJamTemporary, Busy, DataError, DeviceDisabled, EEPROMError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, OutOfRange, PositionLost, RAMError, UARTError, UnknownProcedure, DownloadInProgress

6.6.4 Get Tilt

Table 6.6.4.1: Elementary procedure Get Tilt

Name: **GetTilt**				
Code: **0x34**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.6.4.2: Initiating message parameters for Get Tilt

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Description:

On receipt of the initiating message the secondary device will return the current tilt value.

The returned tilt value is given ~~in increments of 0.1°~~ in the format specified in section 6.6.3.

Initiating message data format:

No data carried.

Response message data format:

<OK><TiltLowOctet><TiltHighOctet>

Response message data format upon error:

<FAIL><ReturnCode1>…<ReturnCodeN>

Applicable return codes:

Busy, DeviceDisabled, EEPROMError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, PositionLost, RAMError, UARTError, UnknownProcedure, DownloadInProgress

6.6.5 Alarm

Table 6.6.5.1: Elementary procedure Alarm

Name: **Alarm**				
Code: **0x07**	Issued by: **Secondary device**	Procedure class: **2**	Download operation: **Yes**	Download boot mode: **Mandatory**

Table 6.6.5.2: Initiating message parameters for Alarm

Number	Length	Type	Description
2 i – 1	**1 octet**	**~~Hexadecimal~~Unsigned integer**	**Return code i; see annex A**
2 I	**1 octet**	**~~Hexadecimal~~Unsigned integer**	**State flag i**

i = 1 … N

Description:

The secondary device uses this procedure to report alarms to the primary device. The secondary device shall only perform the alarm procedure if the primary device has performed the Alarm Subscribe procedure since the last reset. An alarm procedure is performed if an error state has changed since the previous alarm message. All return codes marked as alarms in Annex A of this TS may be used in the initiating message.

Initiating message data format:

<ReturnCode1><StateFlag1>…<ReturnCodeN><StateFlagN>

6.7 Multi-antenna elementary procedures

6.7.1 Antenna Calibrate

Table 6.7.1.1: Elementary procedure Antenna Calibrate

Name: **AntennaCalibrate**				
Code: **0x80**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.7.1.2: Initiating message parameters for Antenna Calibrate

Number	Length	Type	Description
1	**1 octet**	**~~Integer~~Unsigned integer**	**Antenna number**

Description:

On receipt of the initiating message the secondary device shall perform a calibration of the antenna addressed by the antenna number. During ~~the~~ calibration the actuator is driven through the whole tilt range of the antenna.

The response time to this Antenna Calibrate procedure shall be less than 4 minutes.

Initiating message data format:

<antenna number>

Response message data format:

<antenna number><OK>

Response message data format upon error:

<antenna number><FAIL><ReturnCode1>…<ReturnCodeN>

Applicable return codes:

ActuatorDetectionFail, ActuatorJamPermanent, ActuatorJamTemporary, Busy, DataError, DeviceDisabled, EEPROMError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, PositionLost, RAMError, UARTError, UnknownProcedure, UnknownAntennaNumber, DownloadInProgress

6.7.2 Antenna Set Tilt

Table 6.7.2.1: Elementary procedure Antenna Set Tilt

Name: **AntennaSetTilt**				
Code: **0x81**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.7.2.2: Initiating message parameters for Antenna Set Tilt

Number	Length	Type	Description
1	**1 octet**	**Signed i~~I~~nteger**	**Antenna number**
2	**2 octets**	**~~16-bit s~~Signed integer ~~little-endian~~**	**Tilt value**

Description:

On receipt of the initiating message the secondary device shall set the electrical tilt of the antenna addressed by the antenna number in increments of 0.1°. The electrical tilt value describes the elevation angle between the direction orthogonal to the antenna element axis and the maximum of its main beam in the elevation plane. A positive electrical tilt angle means that the antenna beam is directed below the direction orthogonal to the antenna axis.

The secondary device shall respond to the initiating message in less than 2 minutes.

The format of the value of parameter 2 is given in section 6.6.3. ~~The format of the provided tilt value is a 2-complement 16 bit signed number sent in little-endian order. Tilt values are given in 0.1° increments starting from zero, for example: Tilt 3.2° is 0x0020, Tilt −3.2° is 0xFFE0.~~

Initiating message data format:

<antenna number><TiltLowOctet><TiltHighOctet>

Response message data format:

<antenna number><OK>

Response message data format upon error:

<antenna number><FAIL><ReturnCode1>…<ReturnCodeN>

Applicable return codes:

ActuatorDetectionFail, ActuatorJamPermanent, ActuatorJamTemporary, Busy, DataError, DeviceDisabled, EEPROMError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, OutOfRange, PositionLost, RAMError, UARTError, UnknownProcedure, UnknownAntennaNumber, DownloadInProgress

6.7.3 Antenna Get Tilt

Table 6.7.3.1: Elementary procedure Antenna Get Tilt

Name: **AntennaGetTilt**				
Code: **0x82**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.7.3.2: Initiating message parameters for Antenna Get Tilt

Number	Length	Type	Description
1	**1 octet**	~~**Integer**~~**Unsigned interger**	**Antenna number**

Description:

On receipt of the initiating message the secondary device will return the current tilt value of the antenna addressed by the antenna number.

The returned tilt value is ~~given in increments of 0.1°~~ in the format specified in section 6.6~~7~~.3~~2~~.

Initiating message data format:

<antenna number>

Response message data format:

<antenna number><OK><TiltLowOctet><TiltHighOctet>

Response message data format upon error:

<antenna number><FAIL><ReturnCode1>…<ReturnCodeN>

Applicable return codes:

ActuatorDetectionFail, Busy, DataError, DeviceDisabled, EEPROMError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, PositionLost, RAMError, UARTError, UnknownProcedure, UnknownAntennaNumber, DownloadInProgress

6.7.4 Antenna Set Data

Table 6.7.4.1: Elementary procedure Antenna Set Data

Name: **AntennaSetData**				
Code: **0x83**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.7.4.2: Initiating message parameters for Antenna Set Data

Number	Length	Type	Description
1	**1 octet**	**~~Integer~~Unsigned integer**	**Antenna number**
2	**1 octet**	**~~Hexadecimal~~Unsigned integer**	**Field number; see annex B**
3	**See annex B**	**See annex B**	**Data to write**

Description:

On receipt of the initiating message the secondary device should write the provided data for the antenna addressed by the antenna number into the fields optionally provided for configuration data and listed in annex B of this TS. If an attempt is made to write to fields which are not supported by a particular device no error is returned but the data for those fields is ignored. If an attempt is made to write to fields which are not supported for the addressed antenna the return code UnknownParameter is returned and the data for those fields is ignored.

Initiating message data format:

<antenna number><field number><data octets for field>

Response message data format:

<antenna number><OK>

Response message data format upon error:

<antenna number><FAIL><ReturnCode1>...<ReturnCodeN>

Applicable return codes:

Busy, DataError, DeviceDisabled, EEPROMError, FlasheError, OtherHardwareError, OtherSoftwareError, OutOfRange, RAMError, UARTError, UnknownProcedure, ReadOnly, UnknownParameter, UnknownAntennaNumber, DownloadInProgress

6.7.5 Antenna Get Data

Table 6.7.5.1: Elementary procedure Antenna Get Data

Name: **AntennaGetData**				
Code: **0x84**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.7.5.2: Initiating message parameters for Antenna Get Data

Number	Length	Type	Description
1	**1 octet**	**~~Integer~~Unsigned integer**	**Antenna number**
–i	**1 octet**	**~~Hexadecimal~~Unsigned integer**	**Field number to read; see annex B**

i = 1 … N

Description:

On receipt of the initiating message the secondary device shall return the data stored for the addressed antenna in the fields for configuration data specified by the field numbers in the initiating message and listed in annex B of this TS.

The field numbers are not necessarily contiguous or ordered. For field numbers which are not supported by the secondary device for the addressed antenna no data is returned for that field.

Initiating message data format:

<antenna number><field number 1><field number 2> …<field number N>

Response message data format:

<antenna number><OK><field number 1><data octets for field number 1><field number 2><data octets for field number 2> … <field number N><data octets for field number N>

Response message data format upon error:

<antenna number><FAIL><ReturnCode1>...<ReturnCodeN>

Applicable return codes:

Busy, DataError, DeviceDisabled, EEPROMError, FlashError, OtherHardwareError, OtherSoftewareError, RAMError, UARTError, UnknownProcedure, UnknownAntennaNumber, DownloadInProgress

6.7.6 Antenna Alarm

Table 6.7.6.1: Elementary procedure Antenna Alarm

Name: **AntennaAlarm**				
Code: 0x85	Issued by: **Secondary device**	Procedure class: **2**	Download operation: **Yes**	Download boot mode: **Mandatory**

Table 6.7.6.2: Initiating message parameters for Antenna Alarm

Number	Length	Type	Description
1	**1 octet**	**~~Integer~~Unsigned integer**	**Antenna number**
2 i – 1	**1 octet**	**~~Hexadecimal~~Unsigned integer**	**Return code i; see annex A**
2 i	**1 octet**	**~~Hexadecimal~~Unsigned integer**	**State flag i**

i = 1 … N

Description:

The multi-antenna secondary device uses this procedure to report antenna alarms to the primary device. This procedure shall only be performed if the secondary has performed an AlarmSubscribe procedure since its latest reset. Multi-antenna devices shall use this *AntennaAlarm* procedure only for multi-antenna specific alarms and the *Alarm* procedure in subclause 6.6.5 for the other alarms.

Initiating message data format:

<antenna number><ReturnCode1><StateFlag1>…<ReturnCodeN><StateFlagN>

6.7.7 Antenna Clear Active Alarms

Table 6.7.7.1: Elementary procedure Antenna Clear ~~Antenna~~ Active Alarms

Name: **AntennaClearActiveAlarms**

Code: 0x86	Issued by: **Secondary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.7.76.2: Initiating message parameters for ~~ClearAntenna~~ Antenna Clear Active Alarms

Number	Length	Type	Description
1	**1 octet**	**~~Integer~~Unsigned integer**	**Antenna number**

Description:

On receipt of the initiating message the secondary device shall first clear~~s~~ all stored alarm information for the addressed antenna and then return~~s~~ a procedure response message.

Initiating message data format:

<antenna number>

Response message data format:

<antenna number><OK>

Response message data format upon error:

<antenna number><FAIL><ReturnCode1>…<ReturnCodeN>

Applicable return codes:

Busy, FlashEraseError, FlashError, OtherHardwareError, OtherSoftwareError, RAMError, UARTError, UnknownParameter, DownloadInProgress.

6.7.8 Antenna Get Error Status

Table 6.75.82.1: Elementary procedure Antenna Get Error Status

Name: **AntennaGetErrorStatus**				
Code: **0x87**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **No**

Table 6.75.82.2: Initiating message parameters for Antenna Get Error Status

Number	Length	Type	Description
1~~None~~	**1 octet~~0 octets~~**	**Unsigned integer~~None~~**	**Antenna number~~No data carried~~**

Description:

On receipt of the initiating message the secondary device shall report~~s~~ back the return codes for the addressed antenna corresponding to the active errors in the secondary device to the primary device.

Initiating message data format:

<antenna number>~~No data carried.~~

Response message data format:

<antenna number><OK><ReturnCode1>…<ReturnCodeN>

Response message data format upon error:

<antenna number><FAIL><ReturnCode1><ReturnCode2>…<ReturnCodeN>

Applicable return codes:

ActuatorDetectionFail, ActuatorJamPermanent, ActuatorJamTemporary, EEPROMError, FlashEraseError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, PositionLost, RAMError, UARTError, Busy, DataError, DeviceDisabled, UnknownParameter, WorkingSoftwareMissing, DownloadInProgress

NOTE1: ActuatorDetectionFail, ActuatorJamPermanent, ActuatorJamTemporary, EEPROMError, FlashEraseError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, PositionLost, RAMError, UARTError WorkingSoftwareMissing may be part of OK response message

NOTE2: Busy, DataError, DeviceDisabled, UnknownParameter, OtherHardwareError, OtherSoftwareError, FlashError, RAMError, UARTError, DownloadInProgress may be part of FAIL response message.

6.7.9 Antenna Get Number Of Antennas

Table 6.7.95.1: Elementary procedure Antenna Get Number Of Antennas

Name: **AntennaGetNumberOfAntennas**				
Code: **0x88**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.7.95.2: Initiating message parameters for Antenna Get ~~Data~~Number Of Antennas

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Description:

On receipt of the initiating message the secondary device shall return the number of antennas it controls.

Initiating message data format:

No data carried.

Response message data format:

<OK><number of antennas>

Response message data format upon error:

<FAIL><ReturnCode1>...<ReturnCodeN>

Applicable return codes:

Busy, DataError, DeviceDisabled, EEPROMError, FlashError, OtherHardwareError, OtherSoftewareError, RAMError, UARTError, UnknownProcedure, UnknownAntennaNumber, DownloadInProgress

********** Unchanged Parts omitted **********

Annex B (normative): Assigned fields for additional data

The following standard fields have no operational impact and are used by the procedures SetDeviceData and GetDeviceData. Little-endian order is used for storage of multiple-octet numbers. Where ASCII variables are shorter than the assigned field lengths the characters are right aligned and leading blanks are filled with null characters (0x00).

Table B.1: Assigned fields for additional data

Field No.	Length (octets)	Format	Description
0x01	15	ASCII	Antenna model number
0x02	17	ASCII	Antenna serial number
0x03	2	16-bit unsigned	Antenna frequency band(s): see below
0x04	3	3 x 8-bit unsigned	Beamwidth for each band in frequency order (deg) (example 800/900MHz, 1800/1900MHz, 2100MHz)
0x05	3	3 x 8-bit unsigned	Gain for each band in frequency order (dB/10) (example 800/900MHz, 1800/1900MHz, 2100MHz)
0x06	2	16-bit signed	Maximum supported tilt (degrees~~/~~ * 10), Format as in section 6.11
0x07	2	16-bit signed	Minimum supported tilt (degrees~~/~~ * 10), Format as in section 6.11
0x21	6	ASCII	Installation date
0x22	5	ASCII	Installer's ID
0x23	12	ASCII	Base station ID
0x24	4	ASCII	Sector ID
0x25	2	16-bit unsigned	Antenna bearing
0x26	1	8-bit signed	Installed mechanical tilt (degrees * ~~/~~10)

Table B.2: Coding for antenna frequency bands in field 0x03

	Field 0x03
Bit No	Frequency band(MHz)
1	800
2	900
3	1500
4	1800
5	1900
6	2100

7 and above	Reserved

Examples of frequency bands: 0000 0000 0001 0000 = 1800MHz,

0000 0000 0001 1100 = 1800, 1900 and 2100MHz

Annex C (normative): Procedure sequence for download of software to a secondary device

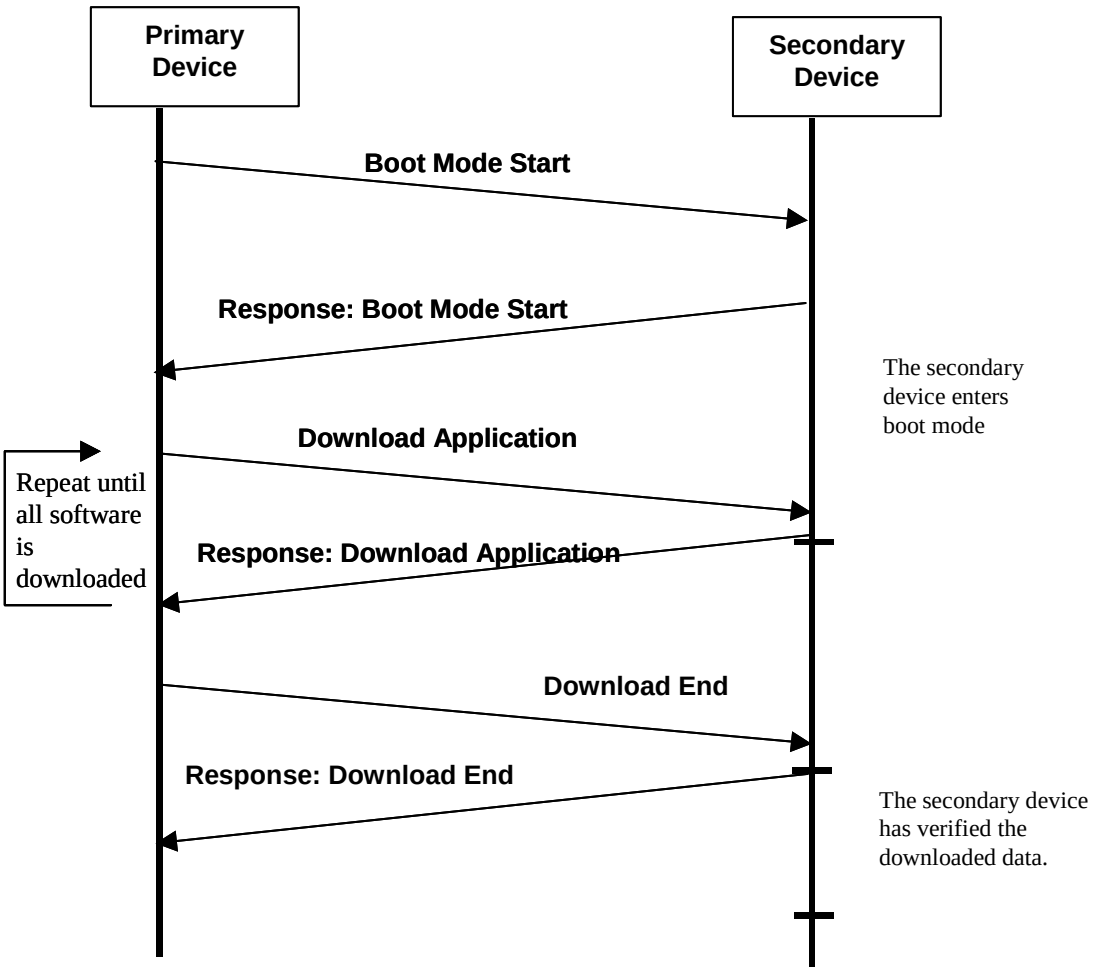


Figure C.1: Procedure sequence for Software Download.

Annex D (informative): Overview of elementary procedures

Table D.1: Elementary Procedures and Procedure Codes

Elementary Procedure	Procedure Code	Issued by	Download boot mode operation
Common Procedure Set			
(Reserved)	0x01		
Reset Software	0x03	primary device	mandatory
Get Error Status	0x04	primary device	mandatory
Get Information	0x05	primary device	mandatory
Clear Active Alarms	0x06	primary device	mandatory
Read User Data	0x10	primary device	optional
Write User Data	0x11	primary device	optional
Alarm Subscribe	0x12	primary device	mandatory
Self Test	0x0A	primary device	optional
Set Device Data	0x0E	primary device	optional
Get Device Data	0x0F	primary device	optional
Boot Mode Start	0x40	primary device	mandatory
Download Application	0x41	primary device	mandatory
Download End	0x42	primary device	mandatory
Single-Antenna Procedure Set			
Calibrate	0x31	primary device	optional
Send Configuration Data	0x32	primary device	optional
Set Tilt	0x33	primary device	optional
Get Tilt	0x34	primary device	optional
Alarm	0x07	secondary device	mandatory
Multi-Antenna Procedure Set			
Antenna Calibrate	0x80	primary device	optional
Antenna SetTilt	0x81	primary device	optional
Antenna GetTilt	0x82	primary device	optional
Antenna SetData	0x83	primary device	optional
Antenna GetData	0x84	primary device	optional
Antenna Alarm	0x85	secondary device	mandatory
Antenna Clear Active Alarms	0x86	primary device	optional
Antenna Get Error Status	0x87	primary device	optional
Antenna Get Number of Antennas	0x88	primary device	optional

NOTE: The notion mandatory in the download boot mode operation indicates that the listed procedures are mandatory if the download boot mode state can be entered by the secondary device.

********** End of Changes **********

3GPP TSG-RAN3 Meeting #45 ***Tdoc ⌘ R3-041642***
Shin-Yokohama, Japan, 15th - 19th November 2004

CR-Form-v7.1

CHANGE REQUEST

⌘ **25.463** **CR** **005** ⌘ **rev** **1** ⌘ Current version: **6.0.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ ☐ ME ☐ Radio Access Network ☒ Core Network ☐

Title:	⌘	Antenna Send Configuration Data procedure missing
Source:	⌘	RAN3
Work item code:	⌘	RANimp-TiltAnt
Date:	⌘	18/11/2004
Category:	⌘	**F**
Release:	⌘	Rel-6

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

Ph2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*
Rel-7	*(Release 7)*

Reason for change:	⌘	Missing procedure
Summary of change:	⌘	Procedure ‘Antenna Send Configuration Data’ defined.
Consequences if not approved:	⌘	Incomplete specification.

Clauses affected:	⌘	6.3, 6.7.n (new) and Annex D

Other specs affected: ⌘	Y	N		⌘
		X	Other core specifications	
		X	Test specifications	
		X	O&M Specifications	

Other comments: ⌘

How to create CRs using this form:
Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm. Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.
2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

6.3 Overview of elementary procedures

The set of elementary procedures for RET antenna control provides procedure-oriented instructions. An overview of the procedures is given in annex D. Table 6.3.1 lists all common elementary procedures described in section 6.5. Table 6.3.2 lists all elementary procedures specific for single-antenna device types described in section 6.6. Table 6.3.3 lists all elementary procedures specific for multi-antenna device types described in section 6.7. Section 6.4 describes how to interpret the elementary procedure definitions in sections 6.5 to 6.7.

Some elementary procedures shall be performed in sequence as described in Annex C for the software download.

Table 6.3.1: Common elementary procedure set for all device types

Command	Requirement	Comment
Reset Software	mandatory	
Get Error Status	mandatory	
Get Information	mandatory	
Clear Active Alarms	mandatory	
Alarm Subscribe	mandatory	
Read User Data	mandatory	
Write User Data	mandatory	
Self Test	mandatory	
Set Device Data	mandatory	
Get Device Data	mandatory	
Boot Mode Start	optional	This procedure is mandatory if the software download feature is supported.
Download Application	optional	This procedure is mandatory if the software download feature is supported.
Download End	optional	This procedure is mandatory if the software download feature is supported.

Table 6.3.2: Elementary procedure set for singe-antenna device type

Command	Requirement	Comment
Calibrate	mandatory	
Send Configuration Data	mandatory	
Set Tilt	mandatory	
Get Tilt	mandatory	
Alarm	mandatory	

Table 6.3.3: Elementary procedure set for Multiple-Antenna Device Type

Command	Requirement	Comment
Antenna Calibrate	mandatory	
Antenna Send Configuration Data	mandatory	
Antenna Set Tilt	mandatory	

Antenna Get Tilt	mandatory	
Antenna Set Data	mandatory	
Antenna Get Data	mandatory	
Antenna Alarm	mandatory	
Antenna Clear Active Alarms	mandatory	
Antenna Get Error Status	mandatory	
Antenna Get Number Of Antennas	mandatory	

6.7.9 Antenna Get Number Of Antennas

Table 6.7.5.1: Elementary procedure Antenna Get Number Of Antennas

Name: **AntennaGetNumberOfAntennas**				
Code: **0x88**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.7.5.2: Initiating message parameters for Antenna Get Data

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Description:

On receipt of the initiating message the secondary device shall return the number of antennas it controls.

Initiating message data format:

No data carried.

Response message data format:

<OK><number of antennas>

Response message data format upon error:

<FAIL><ReturnCode1>...<ReturnCodeN>

Applicable return codes:

Busy, DataError, DeviceDisabled, EEPROMError, FlashError, OtherHardwareError, OtherSoftewareError, RAMError, UARTError, UnknownProcedure, UnknownAntennaNumber, DownloadInProgress

6.7.n Antenna Send Configuration Data

Table 6.7.n.1: Elementary procedure Antenna Send Configuration Data

Name: **AntennaSendConfigurationData**				
Code: 0x86~~0xnn~~	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.7.n.2: Initiating message parameters for Send Configuration Data

Number	Length	Type	Description
1	**1 octet**	**Integer**	**Antenna number**
2	**Vendor specific; Maximum of 70 octets**	**Vendor specific**	**Configuration data**

Description:

On receipt of the initiating message the secondary device shall store the provided vendor and antenna specific configuration data for the relationship between the movement of the drive system and the beam tilt position of the antenna.

If the configuration data exceeds 70 octets, the data shall be split into a number of 70 octet segments and one final segment with whatever is left. The primary device transmits the segments in order. The layer 2 sequence numbers guarantee that no segment will be lost or received out of order.

Initiating message data format:

<antenna number><DataOctet1>…<DataOctetN>

Response message data format:

<antenna number><OK>

Response message data format upon error:

<antenna number><FAIL><ReturnCode1>…<ReturnCodeN>

Applicable return codes:

tbd

Annex D (informative): Overview of elementary procedures

Table D.1: Elementary Procedures and Procedure Codes

Elementary Procedure	Procedure Code	Issued by	Download boot mode operation
Common Procedure Set			
(Reserved)	0x01		
Reset Software	0x03	primary device	mandatory
Get Error Status	0x04	primary device	mandatory
Get Information	0x05	primary device	mandatory
Clear Active Alarms	0x06	primary device	mandatory
Read User Data	0x10	primary device	optional
Write User Data	0x11	primary device	optional
Alarm Subscribe	0x12	primary device	mandatory
Self Test	0x0A	primary device	optional
Set Device Data	0x0E	primary device	optional
Get Device Data	0x0F	primary device	optional
Boot Mode Start	0x40	primary device	mandatory
Download Application	0x41	primary device	mandatory
Download End	0x42	primary device	mandatory
Single-Antenna Procedure Set			
Calibrate	0x31	primary device	optional
Send Configuration Data	0x32	primary device	optional
Set Tilt	0x33	primary device	optional
Get Tilt	0x34	primary device	optional
Alarm	0x07	secondary device	mandatory
Multi-Antenna Procedure Set			
Antenna Calibrate	0x80	primary device	optional
Antenna Send Configuration Data	0x86	primary device	optional
Antenna SetTilt	0x81	primary device	optional
Antenna GetTilt	0x82	primary device	optional
Antenna SetData	0x83	primary device	optional
Antenna GetData	0x84	primary device	optional
Antenna Alarm	0x85	secondary device	mandatory

NOTE: The notion mandatory in the download boot mode operation indicates that the listed procedures are mandatory if the download boot mode state can be entered by the secondary device.

3GPP TSG-RAN3 Meeting #45
Shin-Yokohama, Japan, 15th - 19th November 2004

Tdoc ⌘ R3-041643

CR-Form-v7.1

CHANGE REQUEST

⌘ **25.463** **CR** **007** ⌘ **rev** **1** ⌘ Current version: **6.0.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ ☐ ME ☐ Radio Access Network ☒ Core Network ☐

Title:	⌘	Introduction of Software Download State model
Source:	⌘	RAN3
Work item code:	⌘	RANimp-TiltAnt
Date:	⌘	18/11/2004
Category:	⌘	**F**
Release:	⌘	Rel-6

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

Ph2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*
Rel-7	*(Release 7)*

Reason for change:	⌘	Alignment of RETAP specification with agreed State Model for RET.
Summary of change:	⌘	Implications of the state model are introduced through the RETAP specification.
Consequences if not approved:	⌘	The state model and the elementary procedure set will be in conflict. Unclear implementation.

Clauses affected:	⌘	6.3 to 7, Annex A, Annex C and Annex D

Other specs affected: ⌘	Y	N	
		X	Other core specifications
		X	Test specifications
		X	O&M Specifications

⌘

Other comments: ⌘

How to create CRs using this form:
Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm.
Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.

2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

6.3 Overview of elementary procedures

The set of elementary procedures for RET antenna control provides procedure-oriented instructions. An overview of the procedures is given in annex D. Table 6.3.1 lists all common elementary procedures described in section 6.5. Table 6.3.2 lists all elementary procedures specific for single-antenna device types described in section 6.6. Table 6.3.3 lists all elementary procedures specific for multi-antenna device types described in section 6.7. Section 6.4 describes how to interpret the elementary procedure definitions in sections 6.5 to 6.7.

Some elementary procedures shall be performed in sequence as described in Annex C for the software download.

Table 6.3.1: Common elementary procedure set for all device types

Command	Requirement	Comment
Reset Software	mandatory	
Get Error Status	mandatory	
Get Information	mandatory	
Clear Active Alarms	mandatory	
Alarm Subscribe	mandatory	
Read User Data	mandatory	
Write User Data	mandatory	
Self Test	mandatory	
Set Device Data	mandatory	
Get Device Data	mandatory	
~~Boot Mode~~Download Start	optional	This procedure is mandatory if the software download feature is supported.
Download Application	optional	This procedure is mandatory if the software download feature is supported.
Download End	optional	This procedure is mandatory if the software download feature is supported.

NEXT CHANGED SECTION

6.4 Description of elementary procedures

Table 6.4.1: Description of elementary procedures

Name: The name used to refer to the elementary procedure				
Code: The code is defined here. All other code references are informative	**Issued by:** Primary device or secondary device	**Procedure class:** Class 1 or Class 2	**Download~~-~~Mode state~~operation~~:** ~~FFS~~Defines whether the procedure shall be supported in the DownloadMode	~~**Download boot mode:**~~ ~~Defines whether the procedure shall be supported when the secondary device is in the download boot~~

			state.	~~mode state~~

Table 6.4.2: Initiating message parameters

Number	Length	Type	Description

Description:

Describes the purpose of the elementary procedure.

Initiating message data format:

Describes the initiating message parameter order.

Response message data format:

Describes the response message data parameter order in case of procedure success.

Response message data format upon error:

Describes the response message data parameter order in case of procedure failure.

Applicable return codes:

Lists all allowed return codes for the procedure.

6.5 Common elementary procedures

6.5.1 Reset Software

Table 6.5.1.1: Elementary procedure Reset Software

Name: **ResetSoftware**				
Code: **0x03**	Issued by: **Primary device**	Procedure class: **1**	~~Download~~DownloadMode state~~operation:~~: **Yes**	~~Download boot mode:~~ ~~**Mandatory**~~

Table 6.5.1.2: Initiating message parameters for Reset Software

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

NEXT CHANGED SECTION

6.5.2 Get Error Status

Table 6.5.2.1: Elementary procedure Get Error Status

Name: **GetErrorStatus**				
Code: **0x04**	Issued by: **Primary device**	Procedure class: **1**	~~Download~~ DownloadMode state~~operation:~~: **No**	~~Download boot mode: **Mandatory**~~

Table 6.5.2.2: Initiating message parameters for Get Error Status

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

NEXT CHANGED SECTION

6.5.3 Get Information

Table 6.5.3.1: Elementary procedure Get Information

Name: **GetInformation**				
Code: **0x05**	Issued by: **Primary device**	Procedure class: **1**	~~Download~~ DownloadMode state~~operation:~~: ~~**No**~~**Yes**	~~Download boot mode: **Mandatory**~~

Table 6.5.3.2: Initiating message parameters for Get Information

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Description:

NEXT CHANGED SECTION

6.5.4 Clear Active Alarms

Table 6.5.4.1: Elementary procedure Clear Active Alarms

Name: **ClearActiveAlarms**				
Code: **0x06**	Issued by: **Primary device**	Procedure class: **1**	~~Download~~DownloadMode state~~operation:~~: **No**	~~Download boot mode:~~ **~~Mandatory~~**

Table 6.5.4.2: Initiating message parameters for Clear Active Alarms

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

NEXT CHANGED SECTION

6.5.5 Alarm Subscribe

Table 6.5.5.1: Elementary procedure Alarm Subscribe

Name: **AlarmSubscribe**				
Code: **0x12**	Issued by: **Primary device**	Procedure class: **1**	~~Download~~ DownloadMode state~~operation:~~: **No**	~~Download boot mode:~~ **~~Mandatory~~**

Table 6.5.5.2: Initiating pessage parameters for Alarm Subscribe

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

NEXT CHANGED SECTION

6.5.6 Self Test

Table 6.5.6.1: Elementary procedure Self Test

Name: **SelfTest**				
Code: **0x0A**	Issued by: **Primary device**	Procedure class: **1**	~~Download~~ DownloadMode state~~operation:~~: **No**	~~Download boot mode:~~ **~~Optional~~**

Table 6.5.6.2: Initiating message parameters for Self Test

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

NEXT CHANGED SECTION

6.5.7 Set Device Data

Table 6.5.7.1: Elementary procedure Set Device Data

Name: **SetDeviceData**				
Code: **0x0E**	Issued by: **Primary device**	Procedure class: **1**	~~Download~~ DownloadMode state~~operation:~~: **No**	~~Download boot mode:~~ **~~Optional~~**

Table 6.5.7.2: Initiating message parameters for Set Device Data

Number	Length	Type	Description
1	**1 octet**	**Hexadecimal**	**Field number, see annex B**
2	**See annex B**	**See annex B**	**Data to write**

NEXT CHANGED SECTION

6.5.8 Get Device Data

Table 6.5.8.1: Elementary procedure Get Device Data

Name: **GetDeviceData**				
Code: **0x0F**	Issued by: **Primary device**	Procedure class: **1**	~~Download~~ DownloadMode state~~operation:~~: **No**	~~Download boot mode:~~ **~~Optional~~**

Table 6.5.8.2: Initiating message parameters for Get Device Data

Number	Length	Type	Description
i	**1 octet**	**Hexadecimal**	**Field number; see annex B**

i = 1 … N

NEXT CHANGED SECTION

6.5.9 Read User Data

Table 6.5.9.1: Elementary procedure Read User Data

Name: **ReadUserData**				
Code: **0x10**	Issued by: **Primary device**	Procedure class: **1**	~~Download~~ DownloadMode state~~operation:~~: **No**	~~Download boot mode:~~ **~~Optional~~**

Table 6.2.9.2: Initiating message parameters for Read User Data

Number	Length	Type	Description
1	**2 octets**	**Integer**	**Memory offset**
2	**1 octet**	**Integer**	**Number of octets to read**

NEXT CHANGED SECTION

6.5.10 Write User Data

Table 6.5.10.1: Elementary procedure Write User Data

Name: **WriteUserData**				
Code: **0x11**	Issued by: **Primary device**	Procedure class: **1**	~~Download~~ DownloadMode state~~operation:~~: **No**	~~Download boot mode:~~ ~~**Optional**~~

Table 6.5.10.2: Initiating message parameters for Write User Data

Number	Length	Type	Description
1	**2 octets**	**Integer**	**Memory offset**
2	**1 octet**	**Integer**	**Number of octets to write**
3	**Message specific, given by parameter 2**	**Octets**	**Data to write**

NEXT CHANGED SECTION

6.5.11 ~~Boot Mode~~Download Start

Table 6.5.11.1: Elementary procedure Download ~~Boot Mode~~ Start

Name: ~~**BootModeStart**~~**DownloadStart**				
Code: **0x40**	Issued by: **Primary device**	Procedure class: **1**	~~Download~~ DownloadMode state~~operation:~~: **Yes**	~~Download boot mode:~~ ~~**Mandatory**~~

Table 6.5.11.2: Initiating message parameters for Download ~~Boot Mode~~ Start

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Description:

On receipt of this initiating message the software download process shall be initiated. Following transition to the ~~boot~~ DownloadMode state, the secondary device sends <OK>. Previous subscription of alarms by use of the AlarmSubscribe procedure is cancelled~~.~~. Non-volatile memory in the secondary device may be erased after this procedure is completed.

Initiating message data format:

No data carried.

Response message data format:

<OK>

Response message data format upon error:

<FAIL><ReturnCode 1>…<ReturnCode N>

Applicable return codes:

Busy, FlashEraseError, OtherHardwareError, OtherSoftwareError, RAMError, UARTError

6.5.12 Download Application

Table 6.2.12.1: Elementary procedure Download Application

Name: **DownloadApplication**				
Code: **0x41**	Issued by: **Primary device**	Procedure class: **1**	~~Download~~ DownloadMode state~~operation:~~: **Yes**	~~Download boot mode:~~ **~~Mandatory~~**

Table 6.2.12.2: Initiating message parameters for Download Application

Number	Length	Type	Description
None	**Vendor specific**	**Vendor specific**	**Software data**

Description:

Repeated use of this elementary procedure transfers software data from the primary device to the secondary device.

Initiating message data format:

<octet 1><octet 2>…<octet N>

Response message data format:

<OK>

Response message data format upon error:

<FAIL><ReturnCode 1>…<ReturnCode N>

Applicable return codes:

Busy, ChecksumError, DataError, EEPROMError, FlashEraseError, FlashError, OtherHardwareError, OtherSoftwareError, RAMError, UARTError, UnknownProcedure, InvalidSoftware, TooMuchData

NOTE1: UnknownProcedure may not be returned in boot mode.

6.5.13 Download End

Table 6.5.13.1: Elementary procedure Download End

Name: **DownloadEnd**				
Code: **0x42**	Issued by: **Primary device**	Procedure class: **1**	~~Download~~ DownloadMode state~~operation:~~: **Yes**	~~Download boot mode:~~ ~~**Mandatory**~~

Table 6.5.13.2: Initiating message parameters for Download End

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Description:

This elementary procedure signals the end of a multi-message data transfer to the secondary device. The secondary device responds after verifying the received data. ~~If new software has been downloaded~~, ~~T~~the secondary device shall reset autonomously after completion of the layer 2 response ~~to~~and activate the new application software.

Initiating message data format:

No data carried.

Response message data format:

<OK>

Response message data format upon error:

<FAIL><ReturnCode 1>…<ReturnCode N>

Applicable return codes:

Busy, ChecksumError, EEPROMError, FlashEraseError, FlashError, OtherHardwareError, OtherSoftwareError, RAMError, UARTError, UnknownProcedure, WorkingSoftwareMissing, InvalidSoftware, TooMuchData

NOTE1: UnknownProcedure may not be returned during software download.

6.6 Single-antenna elementary procedures

6.6.1 Calibrate

Table 6.6.1.1: Elementary procedure Calibrate

Name: **Calibrate**				
Code:	Issued by:	Procedure class:	~~Download~~	~~Download boot mode:~~

0x31	**Primary Device**	**1**	DownloadMode state~~operation:~~ **No**	~~**Optional**~~

Table 6.6.1.2: Initiating message parameters for Calibrate

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

NEXT CHANGED SECTION

6.6.2 Send Configuration Data

Table 6.6.2.1: Elementary procedure Send Configuration Data

Name: **SendConfigurationData**				
Code: **0x32**	Issued by: **Primary device**	Procedure class: **1**	~~Download~~ DownloadMode state~~operation:~~ **No**	~~Download boot mode:~~ ~~**Optional**~~

Table 6.6.2.2: Initiating message parameters for Send Configuration Data

Number	Length	Type	Description
1	**Vendor specific;** **Maximum of 70 octets**	**Vendor specific**	**Configuration data**

NEXT CHANGED SECTION

6.6.3 Set Tilt

Table 6.6.3.1: Elementary procedure Set Tilt

Name: **SetTilt**				
Code: **0x33**	Issued by: **Primary device**	Procedure class: **1**	~~Download~~ DownloadMode state~~operation:~~: **No**	~~Download boot mode:~~ **~~Optional~~**

Table 6.6.3.2: Initiating message parameters for Set Tilt

Number	Length	Type	Description
1	**2 octets**	**16 bit signed little-endian**	**Tilt value**

NEXT CHANGED SECTION

6.6.4 Get Tilt

Table 6.6.4.1: Elementary procedure Get Tilt

Name: **GetTilt**				
Code: **0x34**	Issued by: **Primary device**	Procedure class: **1**	~~Download~~ DownloadMode state~~operation:~~: **No**	~~Download boot mode:~~ **~~Optional~~**

Table 6.6.4.2: Initiating message parameters for Get Tilt

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

NEXT CHANGED SECTION

6.6.5 Alarm

Table 6.6.5.1: Elementary procedure Alarm

Name: **Alarm**				
Code: **0x07**	Issued by: **Secondary device**	Procedure class: **2**	~~Download~~ DownloadMode state~~operation:~~: ~~**Yes**~~**No**	~~Download boot mode:~~ ~~**Mandatory**~~

Table 6.6.5.2: Initiating message parameters for Alarm

Number	Length	Type	Description
2 i – 1	**1 octet**	**Hexadecimal**	**Return code i; see annex A**
2 I	**1 octet**	**Hexadecimal**	**State flag i**

i = 1 … N

NEXT CHANGED SECTION

6.7 Multi-antenna elementary procedures

6.7.1 Antenna Calibrate

Table 6.7.1.1: Elementary procedure Antenna Calibrate

Name: **AntennaCalibrate**				
Code: **0x80**	Issued by: **Primary device**	Procedure class: **1**	~~Download~~ DownloadMode state~~operation:~~: **No**	~~Download boot mode:~~ ~~**Optional**~~

Table 6.7.1.2: Initiating message parameters for Antenna Calibrate

Number	Length	Type	Description
1	**1 octet**	**Integer**	**Antenna number**

NEXT CHANGED SECTION

6.7.2 Antenna Set Tilt

Table 6.7.2.1: Elementary procedure Antenna Set Tilt

Name: **AntennaSetTilt**				
Code: **0x81**	Issued by: **Primary device**	Procedure class: **1**	~~Download~~ DownloadMode state~~operation:~~: **No**	~~Download boot mode:~~ **~~Optional~~**

Table 6.7.2.2: Initiating message parameters for Antenna Set Tilt

Number	Length	Type	Description
1	**1 octet**	**Integer**	**Antenna number**
2	**2 octets**	**16-bit signed little-endian**	**Tilt value**

Description:

NEXT CHANGED SECTION

6.7.3 Antenna Get Tilt

Table 6.7.3.1: Elementary procedure Antenna Get Tilt

Name: **AntennaGetTilt**				
Code: **0x82**	Issued by: **Primary device**	Procedure class: **1**	~~Download~~ DownloadMode state~~operation:~~: **No**	~~Download boot mode:~~ **~~Optional~~**

Table 6.7.3.2: Initiating message parameters for Antenna Get Tilt

Number	Length	Type	Description

1	**1 octet**	**Integer**	**Antenna number**

NEXT CHANGED SECTION

6.7.4 Antenna Set Data

Table 6.7.4.1: Elementary procedure Antenna Set Data

Name: **AntennaSetData**				
Code: **0x83**	Issued by: **Primary device**	Procedure class: **1**	~~Download~~ DownloadMode stateoperation:: **No**	~~Download boot mode:~~ **~~Optional~~**

Table 6.7.4.2: Initiating message parameters for Antenna Set Data

Number	Length	Type	Description
1	**1 octet**	**Integer**	**Antenna number**
2	**1 octet**	**Hexadecimal**	**Field number; see annex B**
3	**See annex B**	**See annex B**	**Data to write**

NEXT CHANGED SECTION

6.7.5 Antenna Get Data

Table 6.7.5.1: Elementary procedure Antenna Get Data

Name: **AntennaGetData**				
Code: **0x84**	Issued by: **Primary device**	Procedure class: **1**	~~Download~~ DownloadMode stateoperation:: **No**	~~Download boot mode:~~ **~~Optional~~**

Table 6.7.5.2: Initiating message parameters for Antenna Get Data

Number	Length	Type	Description
1	**1 octet**	**Integer**	**Antenna number**
–i	**1 octet**	**Hexadecimal**	**Field number to read; see annex B**

i = 1 … N

NEXT CHANGED SECTION

6.7.6 Antenna Alarm

Table 6.7.6.1: Elementary procedure Antenna Alarm

Name: **AntennaAlarm**				
Code: 0x85	Issued by: **Secondary device**	Procedure class: **2**	~~Download~~ DownloadMode stateoperation:: **~~Yes~~No**	~~Download boot mode:~~ **~~Mandatory~~**

Table 6.7.6.2: Initiating message parameters for Antenna Alarm

Number	Length	Type	Description
1	**1 octet**	**Integer**	**Antenna number**
2 i – 1	**1 octet**	**Hexadecimal**	**Return code i; see annex A**
2 i	**1 octet**	**Hexadecimal**	**State flag i**

i = 1 … N

NEXT CHANGED SECTION

6.7.7 Antenna Clear Active Alarms

Table 6.7.7.1: Elementary procedure Clear Antenna Alarms

Name: **AntennaClearActiveAlarms**				
Code: 0x86	Issued by: **Secondary device**	Procedure class: **1**	~~Download~~ DownloadMode state~~operation:~~: **No**	~~Download boot mode:~~ **~~Optional~~**

Table 6.7.6.2: Initiating message parameters for ClearAntenna Alarm

Number	Length	Type	Description
1	**1 octet**	**Integer**	**Antenna number**

NEXT CHANGED SECTION

6.7.8 Antenna Get Error Status

Table 6.5.2.1: Elementary procedure Antenna Get Error Status

Name: **AntennaGetErrorStatus**				
Code: **0x87**	Issued by: **Primary device**	Procedure class: **1**	~~Download~~ DownloadMode state~~operation:~~: **No**	~~Download boot mode:~~ **~~No~~**

Table 6.5.2.2: Initiating message parameters for Get Error Status

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

NEXT CHANGED SECTION

6.7.9 Antenna Get Number Of Antennas

Table 6.7.5.1: Elementary procedure Antenna Get Number Of Antennas

Name: **AntennaGetNumberOfAntennas**				
Code: **0x88**	Issued by: **Primary device**	Procedure class: **1**	~~Download~~ DownloadMode state~~operation:~~: **No**	~~Download boot mode:~~ ~~**Optional**~~

Table 6.7.5.2: Initiating message parameters for Antenna Get Data

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

7 Unknown elementary procedures

If a secondary device in the OperatingMode state is receiving a correct procedure message with a procedure code not known it shall respond with a failure message stating "UnknownProcedure" as the cause of failure.

~~If a secondary device is unable to recognise an initiating message, the secondary device shall respond as follows:~~

Response message data format:

<FAIL><ReturnCode for UnknownProcedure>

Annex A (normative): Return Codes for secondary devices

Table A.1: Return Codes for Secondary Devices

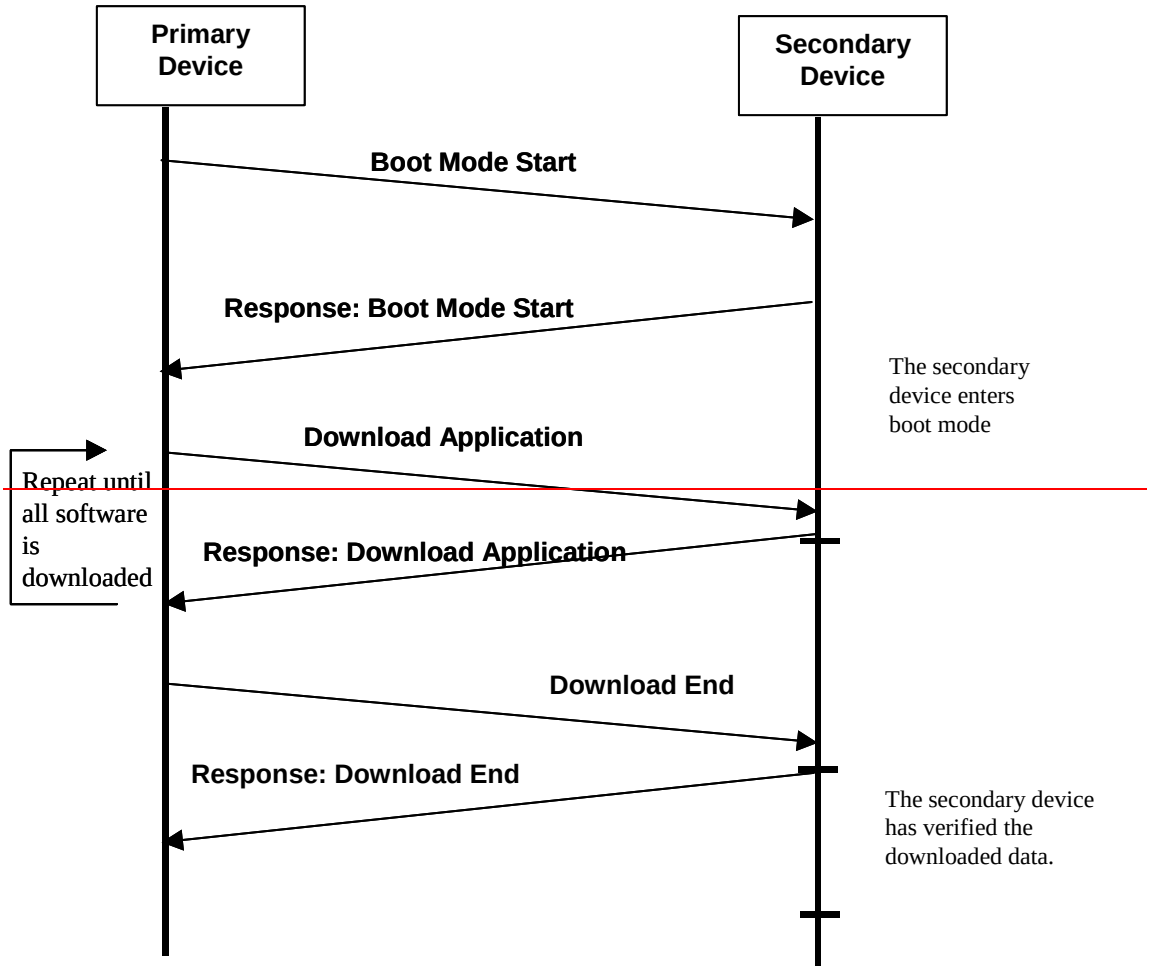
Code	Meaning		Alarm	Download Mode state~~Software Download~~	~~Download Boot Mode~~
0x00	OK	Normal response		X	~~X~~
0x01	Actuator Detection Fail	Signals from the actuator are detected but are abnormal, for example due to failed calibration.	X		
0x02	Actuator Jam Permanent	Actuator cannot be moved permanently.	X		
0x03	Actuator Jam Temporary	Actuator jam has been detected. No movement was detected in response to the normal stimulus.	X		

0x04	Block Number Sequence Error	Used in combination with software download; block number sequence is wrong.			
0x05	Busy	The device is busy and cannot respond until an activity is complete.			
0x06	Checksum Error	Used in combination with software download; checksum incorrect.			
0x07	Procedure Sequence Error	Used in combination with software download; procedure sequence is not permitted, e.g. a SetTilt procedure is received during software update sequence.			
0x08	Data Error	RET AP data fault, e.g. length of data is inconsistent with length fields.			
0x09	Device Disabled	Device is in logical Disabled state and cannot execute Set procedures.			
0x0A	EEPROM Error	EEPROM error detected	X	X	~~X~~
0x0B	FAIL	Abnormal response. Indicates that a procedure has not been executed.		X	~~X~~
0x0C	Flash Erase Error	Used in combination with software download; indicates error when erasing flash memory.	X	X	~~X~~
0x0D	Flash Error	Used in combination with software download; indicates error when writing to flash memory.	X	X	~~X~~
0x0E	Not Calibrated	The device has not completed a calibration operation, or calibration has been lost.	X		
0x0F	Not Scaled	No setup table has been stored in the device.	X		
0x11	Other Hardware Error	Any hardware error which cannot be classified.	X	X	~~X~~
0x12	Other Software Error	Any software error which cannot be classified.	X	X	~~X~~
0x13	Out of Range	A given parameter (e.g. tilt value or memory offset) is out of range.			
0x14	Position Lost	RET controller is unable to return a correct position value, for example there was a power failure while a SetTilt procedure was being executed.	X		
0x15	RAM Error	An error was detected in reading data to/from RAM	X	X	~~X~~
0x16	Segment Number Sequence Error	Used in combination with software download; block sequence number is wrong.			
0x17	UART Error	Hardware specific. This error may be sent after recovery from a temporary error which has prevented the sending or receiving of data.	X	X	~~X~~
0x19	Unknown Procedure	Received procedure is not defined in the 3GPP release version		X	~~X~~
0x1D	Read Only	Used in combination with SetDeviceData procedure when the device parameter cannot be changed		X	~~X~~
0x1E	Unknown Parameter	Specified parameter is not supported for the used procedure. Used as a response to SetDeviceData if an attempt is made to set an unsupported field			

0x1F	Unknown Antenna Number	Specified antenna number for mulit-antenna devices is not supported			
0x20	Too Much Data	More data received during software download than can be stored		X	~~X~~
0x21	Working Software Missing	Application code is missing or broken. The unit is in ~~boot m~~DownloadMode state and may be supporting a limited set of commands. Returned upon unknown procedure when in ~~boot mode~~DownloadMode state.	X		~~X~~
0x22	Invalid Software	Application code being downloaded is detected to be of wrong type. Download of the application code will not be permitted.		X	~~X~~
0x23	Download In Progress	Used instead of UnknownProcedure during software download as response to all commands not supported in boot mode. May also be useful for one physical unit co-siting of several logical units to indicate that other logical units cannot be operated until software download has finished.		X	

NEXT CHANGED SECTION

Annex C (normative): Procedure sequence for download of software to a secondary device



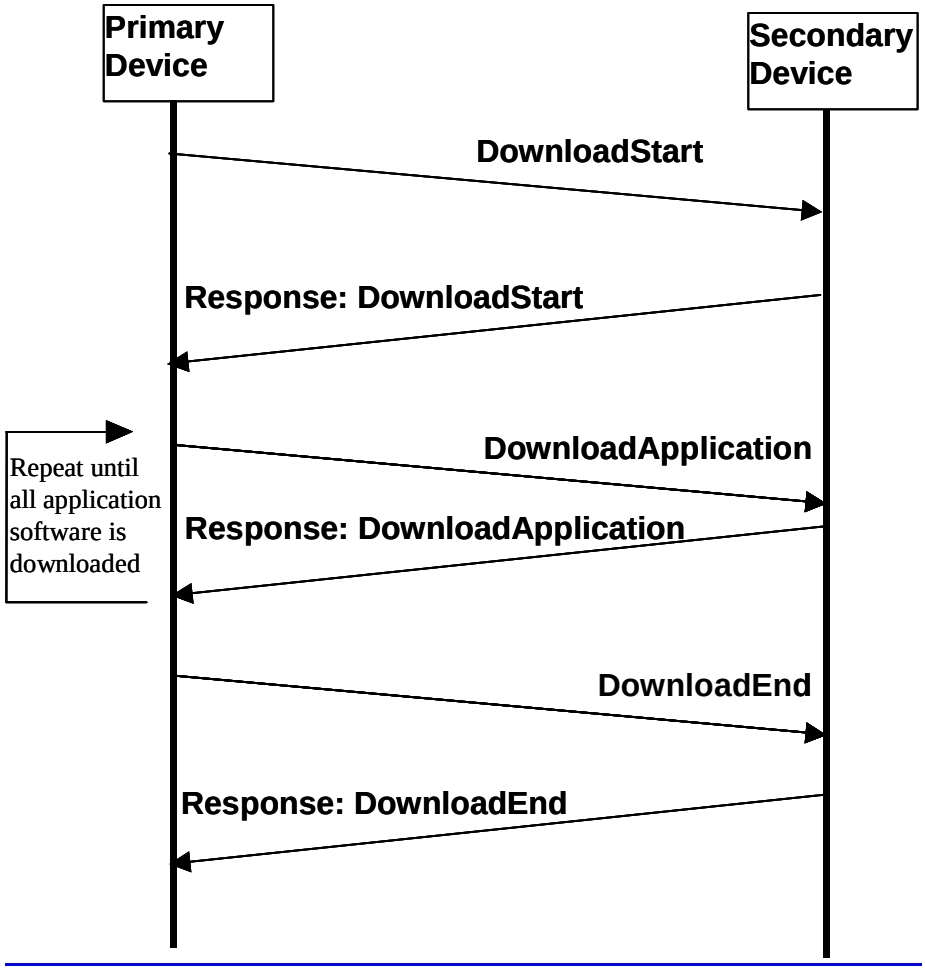


Figure C.1: Procedure sequence for Software Download .

Annex D (informative): Overview of elementary procedures

Table D.1: Elementary Procedures and Procedure Codes

Elementary Procedure	Procedure Code	Issued by	Download ~~boot mode~~Mode state ~~operation~~
Common Procedure Set			
(Reserved)	0x01		
Reset Software	0x03	primary device	~~mandatory~~yes
Get Error Status	0x04	primary device	~~mandatory~~no
Get Information	0x05	primary device	~~mandatory~~yes
Clear Active Alarms	0x06	primary device	~~mandatory~~no
Read User Data	0x10	primary device	~~optional~~no
Write User Data	0x11	primary device	~~optional~~no
Alarm Subscribe	0x12	primary device	~~mandatory~~no
Self Test	0x0A	primary device	~~optional~~no
Set Device Data	0x0E	primary device	~~optional~~no
Get Device Data	0x0F	primary device	~~optional~~no
~~Boot Mode~~Download Start	0x40	primary device	~~mandatory~~yes
Download Application	0x41	primary device	~~mandatory~~yes
Download End	0x42	primary device	~~mandatory~~yes
Single-Antenna Procedure Set			
Calibrate	0x31	primary device	~~optional~~no
Send Configuration Data	0x32	primary device	~~optional~~no
Set Tilt	0x33	primary device	~~optional~~no
Get Tilt	0x34	primary device	~~optional~~no
Alarm	0x07	secondary device	~~mandatory~~no
Multi-Antenna Procedure Set			
Antenna Calibrate	0x80	primary device	~~optional~~no
Antenna SetTilt	0x81	primary device	~~optional~~no
Antenna GetTilt	0x82	primary device	~~optional~~no
Antenna SetData	0x83	primary device	~~optional~~no
Antenna GetData	0x84	primary device	~~optional~~no
Antenna Alarm	0x85	secondary device	~~mandatory~~no

NOTE: The notion ~~mandatory~~ "yes" in the D~~d~~ownload-M~~boot m~~ode ~~operation~~ state column indicates that the listed procedures are mandatory if the ~~d~~Download-~~boot~~ M~~m~~ode state can be entered by the secondary device.

3GPP TSG-RAN3 Meeting #45 ***Tdoc ⌘ R3-041671***
Shin-Yokohama, Japan, 15th - 19th November 2004

CR-Form-v7.1

CHANGE REQUEST

⌘ **25.463** **CR** **8** ⌘ **rev** **3** ⌘ Current version: **6.0.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ ☐ ME ☐ Radio Access Network ☒ Core Network ☐

Title:	⌘	Alarm handling clarification
Source:	⌘	RAN3
Work item code:	⌘	RANimp-TiltAnt
Date:	⌘	18/11/2004
Category:	⌘	**F**
Release:	⌘	Rel-6

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:
Ph2 (GSM Phase 2)
R96 (Release 1996)
R97 (Release 1997)
R98 (Release 1998)
R99 (Release 1999)
Rel-4 (Release 4)
Rel-5 (Release 5)
Rel-6 (Release 6)
Rel-7 (Release 7)

Reason for change:	⌘	Unclear specification of alarm handling
Summary of change:	⌘	Definition of alarm related terms are added. Alarm procedures are renamed (to avoid confusion with the definitions). Alarm status flags are defined. Alarm handling is clarified.
Consequences if not approved:	⌘	Alarm handling is undefined. Correct interpretation of alarms is not possible.

Clauses affected:	⌘	3.1, 6.2, 6.3, 6.5.2, 6.5.6, 6.6.5, 6.7.6, 6.7.8 and Annex D

Other specs affected: ⌘	Y	N	
		X	Other core specifications
		X	Test specifications
		X	O&M Specifications

⌘

Other comments:	⌘	

How to create CRs using this form:
Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm.
Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.
2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be

downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

3 Definitions, symbols and abbreviations

3.1 Definitions

For the purposes of the present document, the following terms and definitions apply.

Active alarm: An alarm which has an alarm state that has been raised, but not cleared.

Alarm: Persistent indication of a fault.

Alarm code: A code that identifies a specific alarm. The alarm code set is a subset of the return code set. The alarm codes are listed in Annex A.

Alarm state: A condition or state in the existence of an alarm. Alarm states are *raised* and *cleared*.

ASCII character: A character forming part of the International Reference Version of the 7-bit character set defined in ISO/IEC 646:1991

Calibrate: Exercise the antenna drive unit over its entire range of travel to ensure fault-free operation and synchronise the measured and actual beam tilt of the antenna

Configuration data: A stored table or function defining the relationship between the physical position of the drive and electrical beam-tilt

Device type: See section 4.7 in [3].

Elementary Procedure: The RETAP protocol consists of Elementary Procedures (EPs). An Elementary Procedure is a unit of interaction between the primary device (Node B) and the secondary devices (RET devices).

An EP consists of an initiating message and possibly a response message.

Two kinds of EPs are used:

- **Class 1**: Elementary Procedures with response (success or failure).
- **Class 2**: Elementary Procedures without response.

For **Class 1** EPs, the types of responses can be as follows:

Successful

- A signalling message explicitly indicates that the elementary procedure has been successfully completed with the receipt of the response.

Unsuccessful

- A signalling message explicitly indicates that the EP failed.

Class 2 EPs are considered always successful.

Error: Deviation of a system from normal operation.

Fault: Lasting error condition.

Little-endian: The order of transmission in which the least-significant octets of a multi-octet representation of a number are transmitted first. Little endian only applies to binary integer representations.

Return code: A 1-octet enumerated response message to an initiating message.

Tilt (also downtilt, tilt angle, beamtilt): The elevation angle between the direction orthogonal to the antenna element axis and the maximum of its main beam in the elevation plane. A positive electrical tilt angle means that the antenna beam is directed below the direction orthogonal to the antenna axis. An antenna has separate values for electrical and mechanical tilt. The mechanical tilt is fixed by the geometry of the installation. In this TS the tilt referred to is always the electrical tilt unless otherwise stated

----------NEXT CHANGED SECTION----------

6.2 General Procedure Handling

All procedures are blocking i.e. no new initiation messages will have to be executed before a response message has been delivered as result of the previously initiated procedure.

The Reset Software procedure shall always be handled in all states and never be blocked.

6.2.n Alarms

When a fault is detected, the corresponding alarm state shall be changed to state *raised* by the secondary device. When the fault no longer exists, the corresponding alarm state shall be changed to state *cleared* by the secondary device. Alarm changes are reported through the AlarmIndication or AntennaAlarmIndication elementary procedures. Whenever an AlarmIndication or AntennaAlarmIndication elementary procedure message is transmitted, it shall contain all the alarm states changed that have not yet been reported as described in sections 6.6.5 and 6.7.6.

6.3 Overview of elementary procedures

The set of elementary procedures for RET antenna control provides procedure-oriented instructions. An overview of the procedures is given in annex D. Table 6.3.1 lists all common elementary procedures described in section 6.5. Table 6.3.2 lists all elementary procedures specific for single-antenna device types described in section 6.6. Table 6.3.3 lists all elementary procedures specific for multi-antenna device types described in section 6.7. Section 6.4 describes how to interpret the elementary procedure definitions in sections 6.5 to 6.7.

Some elementary procedures shall be performed in sequence as described in Annex C for the software download.

Table 6.3.1: Common elementary procedure set for all device types

Command	Requirement	Comment
Reset Software	mandatory	
Get ~~Error~~ Alarm Status	mandatory	
Get Information	mandatory	
Clear Active Alarms	mandatory	
Alarm Subscribe	mandatory	
Read User Data	mandatory	
Write User Data	mandatory	
Self Test	mandatory	
Set Device Data	mandatory	
Get Device Data	mandatory	
Boot Mode Start	optional	This procedure is mandatory if the software download feature is supported.
Download Application	optional	This procedure is mandatory if the software download feature is supported.
Download End	optional	This procedure is mandatory if the software download feature is supported.

Table 6.3.2: Elementary procedure set for singe-antenna device type

Command	Requirement	Comment
Calibrate	mandatory	
Send Configuration Data	mandatory	
Set Tilt	mandatory	
Get Tilt	mandatory	
Alarm Indication	mandatory	

Table 6.3.3: Elementary procedure set for multiple-antenna device type

Command	Requirement	Comment
Antenna Calibrate	mandatory	
Antenna Set Tilt	mandatory	
Antenna Get Tilt	mandatory	
Antenna Set Data	mandatory	
Antenna Get Data	mandatory	
Antenna Alarm Indication	mandatory	
Antenna Clear Active Alarms	mandatory	
Antenna Get Error Status	mandatory	
Antenna Get Number Of Antennas	mandatory	

----------NEXT CHANGED SECTION----------

6.5.2 Get ~~Error~~ Alarm Status

Table 6.5.2.1: Elementary Procedure Get ~~Error~~ Alarm Status

Name: ~~GetErrorStatus~~GetAlarmStatus				
Code: **0x04**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Mandatory**

Table 6.5.2.2: Initiating Message Parameters for Get ~~Error~~ Alarm Status

Number	Length	Type	Description

None	**0 octets**	**None**	**No data carried**

Description:

On receipt of the initiating message the secondary device reports ~~back~~ the alarm codes of ~~corresponding to~~ the active alarms ~~in the secondary device to the primary device~~.

Initiating message data format:

No data carried.

Response message data format:

<OK><ReturnCode1>…<ReturnCodeN>

Response message data format upon error:

<FAIL><ReturnCode1><ReturnCode2>…<ReturnCodeN>

Applicable return codes:

ActuatorDetectionFail, ActuatorJamPermanent, ActuatorJamTemporary, EEPROMError, FlashEraseError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, PositionLost, RAMError, UARTError, Busy, DataError, DeviceDisabled, UnknownParameter, WorkingSoftwareMissing, DownloadInProgress

NOTE1: ActuatorDetectionFail, ActuatorJamPermanent, ActuatorJamTemporary, EEPROMError, FlashEraseError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, PositionLost, RAMError, UARTError WorkingSoftwareMissing may be part of OK response message

NOTE2: Busy, DataError, DeviceDisabled, UnknownParameter, OtherHardwareError, OtherSoftwareError, FlashError, RAMError, UARTError, DownloadInProgress may be part of FAIL response message.

----------NEXT CHANGED SECTION----------

6.5.6 Self Test

Table 6.5.6.1: Elementary Procedure Self Test

Name: **SelfTest**				
Code: **0x0A**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **Optional**

Table 6.5.6.2: Initiating Message Parameters for Self Test

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Description:

On receipt of the initiating message the secondary device executes a test procedure which may include a check of physical and processor functions. The specific tests to be performed are implementation specific, and may include the movement of the adjuster up to <FFS> degrees.

The response message of the secondary device on the procedure provides information on detected faults or, if no fault is detected, with confidence that the operation of the device is normal in all respects.

During the test the operational parameters of the device shall not change beyond operationally acceptable limits and on completion all parameters shall be returned to their initial values.

In the normal response message, in which the self test was executed successfully, the return codes are set to report possible detected ~~functional errors~~faults during the self test. If no ~~errors~~ faults are detected, this shall be signalled by no return codes following <OK>.

In the case of a~~n error~~ failure response message, the self test could not be executed and the return code~~s~~ relate~~s~~ to the inability of the device to perform the requested self-test operation.

Initiating message data format:

No data carried.

Response message data format:

<OK><ReturnCode1>…<ReturnCodeN>

Response message data format upon error:

<FAIL><ReturnCode1>…<ReturnCodeN>

Applicable return codes:

ActuatorDetectionFail, ActuatorJamPermanent, ActuatorJamTemporary, Busy, DataError, DeviceDisabled, EEPROMError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, PositionLost, RAMError, UARTError, WorkingSoftwareMissing, DownloadInProgress.

NOTE1: Only Busy, DataError, DeviceDisabled, EEPROMError, FlashError, OtherHardwareError, OtherSoftwareError, RAMError, UARTError, WorkingSoftwareMissing, DownloadInProgress may be return codes in the fail response message.

----------NEXT CHANGED SECTION----------

6.6.5 Alarm Indication

Table 6.6.5.1: Elementary Procedure Alarm Indication

Name: **AlarmIndication**				
Code: **0x07**	Issued by: **Secondary device**	Procedure class: **2**	Download operation: **Yes**	Download boot mode: **Mandatory**

Table 6.6.5.2: Initiating Message Parameters for Alarm Indication

Number	Length	Type	Description

2 i – 1	**1 octet**	**Hexadecimal**	**Return code i; see annex A**
2 I	**1 octet**	**Hexadecimal**	**State flag i**

i = 1 … N

Description:

The secondary device uses this procedure to report alarm state changes to the primary device. This procedure shall only be performed if the secondary has performed an AlarmSubscribe procedure since its latest reset.

For each alarm, the current alarm state and alarm code shall be reported if and only if any change in its state has occurred during the period of time since the last reported state. An AlarmIndication procedure shall be performed if at least one alarm shall be reported. The first AlarmIndication procedure after the AlarmSubscribe procedure shall report the active alarm states.

Alarm state changes are considered as reported at the time the message is passed to the transport layer.

~~The secondary device uses this procedure to report alarms to the primary device. An alarm procedure is performed if an error state has changed since the previous alarm message. All return codes marked as alarms in Annex A of this TS may be used in the initiating message.~~

Initiating message data format:

<ReturnCode1><StateFlag1>…<ReturnCodeN><StateFlagN>

State flag = 0 represents alarm state *cleared*.

State flag = 1 represents alarm state *raised*.

----------NEXT CHANGED SECTION----------

6.7.6 Antenna Alarm Indication

Table 6.7.6.1: Elementary Procedure Antenna Alarm Indication

Name: **AntennaAlarmIndication**				
Code: 0x85	Issued by: **Secondary device**	Procedure class: **2**	Download operation: **Yes**	Download boot mode: **Mandatory**

Table 6.7.6.2: Initiating Message Parameters for Antenna Alarm Indication

Number	Length	Type	Description
1	**1 octet**	**Integer**	**Antenna number**
2 i – 1	**1 octet**	**Hexadecimal**	**Return code i; see annex A**
2 i	**1 octet**	**Hexadecimal**	**State flag i**

i = 1 … N

Description:

The multi-antenna secondary device uses this procedure to report antenna alarm state changes to the primary device. This procedure shall only be performed if the secondary has performed an AlarmSubscribe procedure since its latest reset. Multi-antenna devices shall use this *AntennaAlarmIndication* procedure only for multi-antenna specific alarms and the *AlarmIndication* procedure in subclause 6.6.5 for the other alarms.

For each alarm, the current alarm state and alarm code shall be reported if and only if any change in its state has occurred during the period of time since the last reported state. An AntennaAlarmIndication procedure shall be performed if at least one alarm shall be reported. The first AntennaAlarmIndication procedure after the AlarmSubscribe procedure shall report the active alarm states.

Alarm state changes are considered as reported at the time the message is passed to the transport layer.

Initiating message data format:

<antenna number><ReturnCode1><StateFlag1>…<ReturnCodeN><StateFlagN>

State flag = 0 represents alarm state *cleared*.

State flag = 1 represents alarm state *raised*.

----------NEXT CHANGED SECTION----------

6.7.8 Antenna Get ~~Error~~ Alarm Status

Table 6.5.2.1: Elementary Procedure Antenna Get ~~Error~~ Alarm Status

Name: ~~AntennaGetErrorStatus~~**AntennaGetAlarmStatus**				
Code: **0x87**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Download boot mode: **No**

Table 6.5.2.2: Initiating Message Parameters for Get ~~Error~~ Alarm Status

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

Description:

On receipt of the initiating message the secondary device reports ~~back~~ the ~~return~~ alarm codes of the active alarms for the addressed antenna ~~corresponding to the active errors in the secondary device to the primary device~~.

Initiating message data format:

No data carried.

Response message data format:

<antenna number><OK><ReturnCode1>…<ReturnCodeN>

Response message data format upon error:

<antenna number><FAIL><ReturnCode1><ReturnCode2>…<ReturnCodeN>

Applicable return codes:

ActuatorDetectionFail, ActuatorJamPermanent, ActuatorJamTemporary, EEPROMError, FlashEraseError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, PositionLost, RAMError, UARTError, Busy, DataError, DeviceDisabled, UnknownParameter, WorkingSoftwareMissing, DownloadInProgress

NOTE1: ActuatorDetectionFail, ActuatorJamPermanent, ActuatorJamTemporary, EEPROMError, FlashEraseError, FlashError, NotCalibrated, NotScaled, OtherHardwareError, OtherSoftwareError, PositionLost, RAMError, UARTError WorkingSoftwareMissing may be part of OK response message

NOTE2: Busy, DataError, DeviceDisabled, UnknownParameter, OtherHardwareError, OtherSoftwareError, FlashError, RAMError, UARTError, DownloadInProgress may be part of FAIL response message.

----------NEXT CHANGED SECTION----------

Annex D (informative): Overview of elementary procedures

Table D.1: Elementary Procedures and Procedure Codes

Elementary Procedure	Procedure Code	Issued by	Download boot mode operation
Common Procedure Set			
(Reserved)	0x01		
Reset Software	0x03	primary device	mandatory
Get ~~Error~~ Alarm Status	0x04	primary device	mandatory
Get Information	0x05	primary device	mandatory
Clear Active Alarms	0x06	primary device	mandatory
Read User Data	0x10	primary device	optional
Write User Data	0x11	primary device	optional
Alarm Subscribe	0x12	primary device	mandatory
Self Test	0x0A	primary device	optional
Set Device Data	0x0E	primary device	optional
Get Device Data	0x0F	primary device	optional
Boot Mode Start	0x40	primary device	mandatory
Download Application	0x41	primary device	mandatory
Download End	0x42	primary device	mandatory
Single-Antenna Procedure Set			
Calibrate	0x31	primary device	optional
Send Configuration Data	0x32	primary device	optional
Set Tilt	0x33	primary device	optional
Get Tilt	0x34	primary device	optional
Alarm Indication	0x07	secondary device	mandatory
Multi-Antenna Procedure Set			

Antenna Calibrate	0x80	primary device	optional
Antenna SetTilt	0x81	primary device	optional
Antenna GetTilt	0x82	primary device	optional
Antenna SetData	0x83	primary device	optional
Antenna GetData	0x84	primary device	optional
Antenna Alarm Indication	0x85	secondary device	mandatory
Antenna Get Alarm Status	0x87	primary device	no

NOTE: The notion mandatory in the download boot mode operation indicates that the listed procedures are mandatory if the download boot mode state can be entered by the secondary device.

3GPP TSG-RAN3 Meeting #45 ***Tdoc ⌘ R3-041673***
Shin-Yokohama, Japan, 15th - 19th November 2004

CR-Form-v7.1

CHANGE REQUEST

⌘ **25.463** **CR** **009** ⌘ **rev** **2** ⌘ Current version: **6.0.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ ☐ ME ☐ Radio Access Network ☒ Core Network ☐

Title:	⌘	RET DC power consumption clarification
Source:	⌘	RAN3
Work item code:	⌘	RANimp-TiltAnt
Date:	⌘	18/11/2004
Category:	⌘	**F**
Release:	⌘	Rel-6

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:
Ph2 (GSM Phase 2)
R96 (Release 1996)
R97 (Release 1997)
R98 (Release 1998)
R99 (Release 1999)
Rel-4 (Release 4)
Rel-5 (Release 5)
Rel-6 (Release 6)
Rel-7 (Release 7)

Reason for change:	⌘	Unclear specification
Summary of change:	⌘	Requirement when the RET is allowed to consume high power included in the elementary procedure definitions
Consequences if not approved:	⌘	Unclear when RET is allowed to consume high power. This has the consequence that it becomes more difficult to superwise other antenna equipment using the same power supply.

Clauses affected:	⌘	2 and 6.4 to 6.7

Other specs affected: ⌘	Y	N		⌘
	X		Other core specifications	CR4 to 25.461
		X	Test specifications	
		X	O&M Specifications	

Other comments: ⌘

How to create CRs using this form:
Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm.
Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.
2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be

downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.

[1] 3GPP TS 25.460: "UTRAN Iuant Interface: General Aspects and Principles".

[2] ISO/IEC 13239 (2nd Edition, March 2000): "Information Technology – Telecommunications and information exchange between systems – High-level data link control (HDLC) procedures".

[3] 3GPP TS 25.462: "UTRAN Iuant Interface: Signalling Transport".

[x] 3GPP TS 25.461: ”UTRAN Iuant Interface: Layer 1”

----------Next Changed Section----------

6.4 Description of elementary procedures

Table 6.4.1: Description of elementary procedures

Name: The name used to refer to the elementary procedure				
Code: The code is defined here. All other code references are informative	**Issued by:** Primary device or secondary device	**Procedure class:** Class 1 or Class 2	**Download operation:** FFS	**Power mode:** Defines the secondary device power consumption as described in [x] during the execution of the Elementary Procedure. ~~Download boot mode: Defines whether the procedure shall be supported when the secondary device is in the download boot mode state~~

Table 6.4.2: Initiating message parameters

Number	Length	Type	Description

Description:

Describes the purpose of the elementary procedure.

Initiating message data format:

Describes the initiating message parameter order.

Response message data format:

Describes the response message data parameter order in case of procedure success.

Response message data format upon error:

Describes the response message data parameter order in case of procedure failure.

Applicable return codes:

Lists all allowed return codes for the procedure.

6.5 Common elementary procedures

6.5.1 Reset Software

Table 6.5.1.1: Elementary procedure Reset Software

Name: **ResetSoftware**				
Code: **0x03**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **Yes**	Power mode: ~~**Download boot mode: Mandatory**~~**Low**

Table 6.5.1.2: Initiating message parameters for Reset Software

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

----------Next Changed Section----------

6.5.2 Get Error Status

Table 6.5.2.1: Elementary procedure Get Error Status

Name: **GetErrorStatus**				
Code: **0x04**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Power mode: **Low**~~Download boot mode: Mandatory~~

Table 6.5.2.2: Initiating message parameters for Get Error Status

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

----------Next Changed Section----------

6.5.3 Get Information

Table 6.5.3.1: Elementary procedure Get Information

Name: **GetInformation**				
Code: **0x05**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Power mode: **Low**~~Download boot mode: **Mandatory**~~

Table 6.5.3.2: Initiating message parameters for Get Information

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

----------Next Changed Section----------

6.5.4 Clear Active Alarms

Table 6.5.4.1: Elementary procedure Clear Active Alarms

Name: **ClearActiveAlarms**				
Code: **0x06**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Power mode: **Low**~~Download boot mode: **Mandatory**~~

Table 6.5.4.2: Initiating message parameters for Clear Active Alarms

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

----------Next Changed Section----------

6.5.5 Alarm Subscribe

Table 6.5.5.1: Elementary procedure Alarm Subscribe

Name: **AlarmSubscribe**				
Code: **0x12**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Power mode: **Low**~~Download boot mode: **Mandatory**~~

Table 6.5.5.2: Initiating pessage parameters for Alarm Subscribe

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

----------Next Changed Section----------

6.5.6 Self Test

Table 6.5.6.1: Elementary procedure Self Test

Name: **SelfTest**				
Code: **0x0A**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Power mode: **High**~~Download boot mode: **Optional**~~

Table 6.5.6.2: Initiating message parameters for Self Test

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

----------Next Changed Section----------

6.5.7 Set Device Data

Table 6.5.7.1: Elementary procedure Set Device Data

Name: **SetDeviceData**				
Code:	Issued by:	Procedure class:	Download operation:	Power mode:

0x0E	**Primary device**	**1**	**No**	**Low**~~Download boot mode:~~ ~~**Optional**~~

Table 6.5.7.2: Initiating message parameters for Set Device Data

Number	Length	Type	Description
1	**1 octet**	**Hexadecimal**	**Field number, see annex B**
2	**See annex B**	**See annex B**	**Data to write**

----------Next Changed Section----------

6.5.8 Get Device Data

Table 6.5.8.1: Elementary procedure Get Device Data

Name: **GetDeviceData**				
Code: **0x0F**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Power mode: **Low**~~Download boot mode:~~ ~~**Optional**~~

Table 6.5.8.2: Initiating message parameters for Get Device Data

Number	Length	Type	Description
i	**1 octet**	**Hexadecimal**	**Field number; see annex B**

i = 1 … N

----------Next Changed Section----------

6.5.9 Read User Data

Table 6.5.9.1: Elementary procedure Read User Data

Name: **ReadUserData**				
Code: **0x10**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Power mode: **Low**~~Download boot mode:~~ ~~**Optional**~~

Table 6.2.9.2: Initiating message parameters for Read User Data

Number	Length	Type	Description
1	**2 octets**	**Integer**	**Memory offset**
2	**1 octet**	**Integer**	**Number of octets to read**

----------Next Changed Section----------

6.5.10 Write User Data

Table 6.5.10.1: Elementary procedure Write User Data

Name: **WriteUserData**				
Code: **0x11**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Power mode: **Low**~~Download boot mode: **Optional**~~

Table 6.5.10.2: Initiating message parameters for Write User Data

Number	Length	Type	Description
1	**2 octets**	**Integer**	**Memory offset**
2	**1 octet**	**Integer**	**Number of octets to write**
3	**Message specific, given by parameter 2**	**Octets**	**Data to write**

----------Next Changed Section----------

6.5.11 Boot Mode Start

Table 6.5.11.1: Elementary procedure Boot Mode Start

Name: **BootModeStart**				
Code: **0x40**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **Yes**	Power mode: **Low**~~Download boot mode: **Mandatory**~~

Table 6.5.11.2: Initiating message parameters for Boot Mode Start

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

----------Next Changed Section----------

6.5.12 Download Application

Table 6.2.12.1: Elementary procedure Download Application

Name: **DownloadApplication**				
Code: **0x41**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **Yes**	Power mode: **Low**~~Download boot mode:~~ ~~**Mandatory**~~

Table 6.2.12.2: Initiating message parameters for Download Application

Number	Length	Type	Description
None	**Vendor specific**	**Vendor specific**	**Software data**

----------Next Changed Section----------

6.5.13 Download End

Table 6.5.13.1: Elementary procedure Download End

Name: **DownloadEnd**				
Code: **0x42**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **Yes**	Power mode: **Low**~~Download boot mode:~~ ~~**Mandatory**~~

Table 6.5.13.2: Initiating message parameters for Download End

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

----------Next Changed Section----------

6.6 Single-antenna elementary procedures

6.6.1 Calibrate

Table 6.6.1.1: Elementary procedure Calibrate

Name: **Calibrate**				
Code: **0x31**	Issued by: **Primary Device**	Procedure class: **1**	Download operation: **No**	Power mode: **High** ~~Download boot mode:~~ ~~**Optional**~~

Table 6.6.1.2: Initiating message parameters for Calibrate

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

----------Next Changed Section----------

6.6.2 Send Configuration Data

Table 6.6.2.1: Elementary procedure Send Configuration Data

Name: **SendConfigurationData**				
Code: **0x32**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Power mode: **Low** ~~Download boot mode:~~ ~~**Optional**~~

Table 6.6.2.2: Initiating message parameters for Send Configuration Data

Number	Length	Type	Description
1	**Vendor specific; Maximum of 70 octets**	**Vendor specific**	**Configuration data**

----------Next Changed Section----------

6.6.3 Set Tilt

Table 6.6.3.1: Elementary procedure Set Tilt

Name: **SetTilt**				
Code: **0x33**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Power mode: **High**~~Download boot mode:~~ ~~**Optional**~~

Table 6.6.3.2: Initiating message parameters for Set Tilt

Number	Length	Type	Description
1	**2 octets**	**16 bit signed little-endian**	**Tilt value**

----------Next Changed Section----------

6.6.4 Get Tilt

Table 6.6.4.1: Elementary procedure Get Tilt

Name: **GetTilt**				
Code: **0x34**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Power mode: **Low**~~Download boot mode:~~ ~~**Optional**~~

Table 6.6.4.2: Initiating message parameters for Get Tilt

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**

----------Next Changed Section----------

6.6.5 Alarm

Table 6.6.5.1: Elementary procedure Alarm

Name: **Alarm**

Code: **0x07**	Issued by: **Secondary device**	Procedure class: **2**	Download operation: **Yes**	Power mode: **Low**~~Download boot mode:~~ ~~**Mandatory**~~

Table 6.6.5.2: Initiating message parameters for Alarm

Number	Length	Type	Description
2 i – 1	**1 octet**	**Hexadecimal**	**Return code i; see annex A**
2 I	**1 octet**	**Hexadecimal**	**State flag i**

i = 1 … N

----------Next Changed Section----------

6.7 Multi-antenna elementary procedures

6.7.1 Antenna Calibrate

Table 6.7.1.1: Elementary procedure Antenna Calibrate

Name: **AntennaCalibrate**				
Code: **0x80**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Power mode: **High**~~Download boot mode:~~ ~~**Optional**~~

Table 6.7.1.2: Initiating message parameters for Antenna Calibrate

Number	Length	Type	Description
1	**1 octet**	**Integer**	**Antenna number**

----------Next Changed Section----------

6.7.2 Antenna Set Tilt

Table 6.7.2.1: Elementary procedure Antenna Set Tilt

Name: **AntennaSetTilt**				
Code:	Issued by:	Procedure class:	Download operation:	Power mode: **High**~~Download boot~~

0x81	**Primary device**	**1**	**No**	~~mode:~~ ~~**Optional**~~

Table 6.7.2.2: Initiating message parameters for Antenna Set Tilt

Number	Length	Type	Description
1	**1 octet**	**Integer**	**Antenna number**
2	**2 octets**	**16-bit signed little-endian**	**Tilt value**

----------Next Changed Section----------

6.7.3 Antenna Get Tilt

Table 6.7.3.1: Elementary procedure Antenna Get Tilt

Name: **AntennaGetTilt**				
Code: **0x82**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Power mode: **Low**~~Download boot mode:~~ ~~**Optional**~~

Table 6.7.3.2: Initiating message parameters for Antenna Get Tilt

Number	Length	Type	Description
1	**1 octet**	**Integer**	**Antenna number**

----------Next Changed Section----------

6.7.4 Antenna Set Data

Table 6.7.4.1: Elementary procedure Antenna Set Data

Name: **AntennaSetData**				
Code: **0x83**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Power mode: **Low**~~Download boot mode:~~ ~~**Optional**~~

Table 6.7.4.2: Initiating message parameters for Antenna Set Data

Number	Length	Type	Description
1	**1 octet**	**Integer**	**Antenna number**
2	**1 octet**	**Hexadecimal**	**Field number; see annex B**
3	**See annex B**	**See annex B**	**Data to write**

----------Next Changed Section----------

6.7.5 Antenna Get Data

Table 6.7.5.1: Elementary procedure Antenna Get Data

Name: **AntennaGetData**				
Code: **0x84**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Power mode: **Low** ~~Download boot mode: **Optional**~~

Table 6.7.5.2: Initiating message parameters for Antenna Get Data

Number	Length	Type	Description
1	**1 octet**	**Integer**	**Antenna number**
–i	**1 octet**	**Hexadecimal**	**Field number to read; see annex B**

i = 1 … N

----------Next Changed Section----------

6.7.6 Antenna Alarm

Table 6.7.6.1: Elementary procedure Antenna Alarm

Name: **AntennaAlarm**				
Code: 0x85	Issued by: **Secondary device**	Procedure class: **2**	Download operation: **Yes**	Power mode: **Low** ~~Download boot mode: **Mandatory**~~

Table 6.7.6.2: Initiating message parameters for Antenna Alarm

Number	Length	Type	Description
1	**1 octet**	**Integer**	**Antenna number**
2 i – 1	**1 octet**	**Hexadecimal**	**Return code i; see annex A**
2 i	**1 octet**	**Hexadecimal**	**State flag i**

i = 1 … N

----------Next Changed Section----------

6.7.7 Antenna Clear Active Alarms

Table 6.7.7.1: Elementary procedure Clear Antenna Alarms

Name: **AntennaClearActiveAlarms**				
Code: 0x86	Issued by: **Secondary device**	Procedure class: **1**	Download operation: **No**	Power mode: **Low**~~Download boot mode: **Optional**~~

Table 6.7.6.2: Initiating message parameters for ClearAntenna Alarm

Number	Length	Type	Description
1	**1 octet**	**Integer**	**Antenna number**

----------Next Changed Section----------

6.7.8 Antenna Get Error Status

Table 6.5.2.1: Elementary procedure Antenna Get Error Status

Name: **AntennaGetErrorStatus**				
Code: **0x87**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Power mode: **Low**~~Download boot mode: **No**~~

Table 6.5.2.2: Initiating message parameters for Get Error Status

Number	Length	Type	Description

None	**0 octets**	**None**	**No data carried**

----------Next Changed Section----------

6.7.9 Antenna Get Number Of Antennas

Table 6.7.5.1: Elementary procedure Antenna Get Number Of Antennas

Name: **AntennaGetNumberOfAntennas**				
Code: **0x88**	Issued by: **Primary device**	Procedure class: **1**	Download operation: **No**	Power mode: **Low**~~Download boot mode: **Optional**~~

Table 6.7.5.2: Initiating message parameters for Antenna Get Data

Number	Length	Type	Description
None	**0 octets**	**None**	**No data carried**