Description
HA87
Terminal Software HA400

Application Note 1900

Revision: 2.0
February 2019

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

0 History

Date	Revision	Author	Comments
30.10.2009	1.0	RN	Last Release
18.02.2019	2.0	CS	New CD; added Introduction

Table 1: History

0.1 Related Documents

No.	Name	Remarks
1	Description HA87 Terminal Software HA209	

Table 2: Related Documents

1 Introduction

The handset HA87 is a classic. For this reason, we decided to republish the software documentation. Please find an overview about currently available variations of the HA87 including software versions and assembly variants in table 3 below.

No.	Article Number	Article Description	Buttons	Software Version	Assembly Variant
1	01 6508 00500051	HA87/21T/PTT/V6011/C6259-5/RJ50/A8	21	HA209	V6011
2	01 6508 00500551	HA87/21T/PTT/V6011-1/C6259-3/RJ50/A8	21	HA400	V6011-1
3	01 6508 00500851	HA87/PTT/V6011-1/C6259-3/RJ50/A8 flame protection	20	HA209	V6011-1

V6011	• PTT hardware output (PIN 1 of the RJ50 plug) • ON/OFF software output (no PIN-out)
V6011-1	• PTT software output (no PIN-out) • ON/OFF hardware output (PIN 1 of the RJ50 plug)

Table 3: Article overview of the HA87 (Revision: January 2019)

2 Preface

The Terminal-Software is a special firmware running on the processor of a handset HA87 which allows you to use the handset as a "Terminal" for **input** and/or **output**. Furthermore, there is also an audio-interface on the connector available.

For example, you could send a text to be displayed in the Handset via a serial connection. A pressed key is analysed in the handset and then a key-code is sent via the serial link to the host-application where it can be further processed. The handset is hereafter considered as a client and your application is the host-system. For the communication between the host and the client there is a special protocol defined, which will be described in this document.

The simplest host-system could be Putty, for example.

Client

(Handset)

Hello

Host-System

(your application)

Serial Connection

+ Audio Interface

Operating Voltage

10-pole-

Western-plug

Key number "1" pressed

display "Hello"
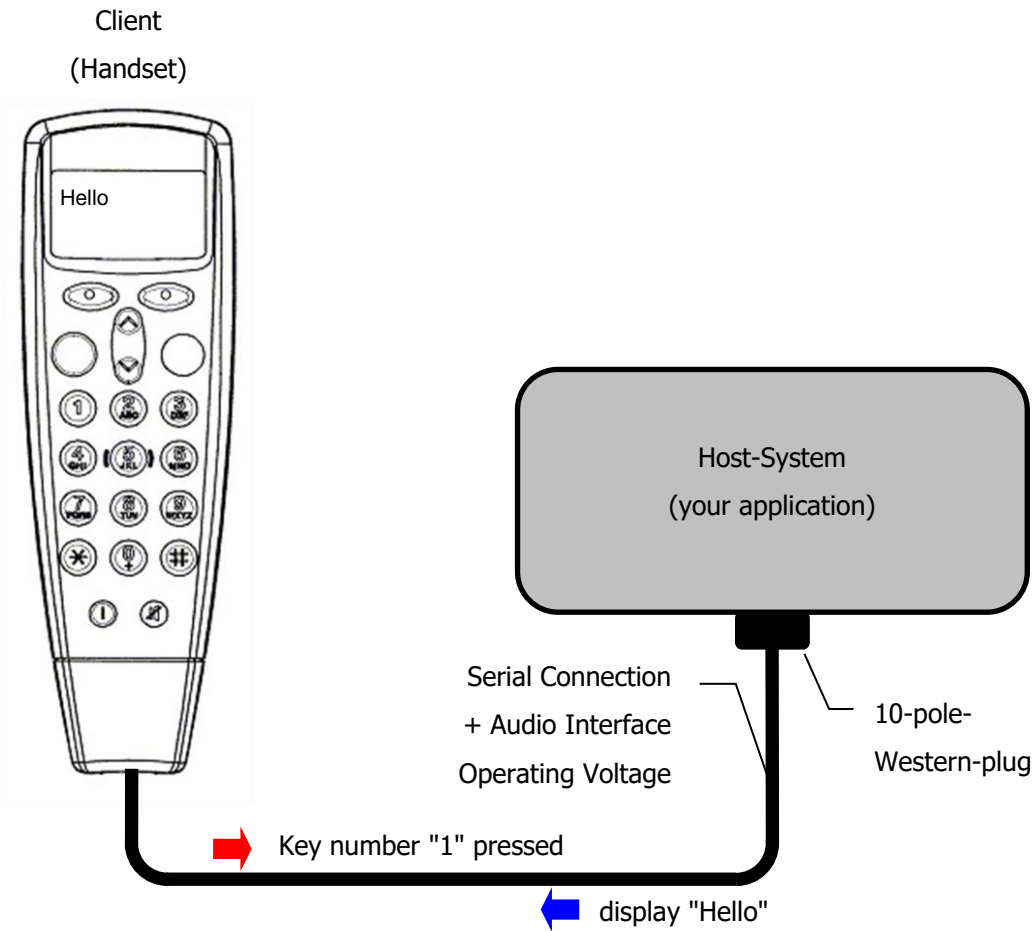
Figure 1: Communication between client and host system

Figure 2: Handset HA87

The Handset consists of the following main parts:

- Microphone
- Loudspeaker (earpiece)
- 21-key keyboard
- PTT button
- 96x54 dot matrix LCD-Display
- Backlight
- Spiral cable with 10-pole-Western-plug
- RS232 Serial Interface
- Hook

2.1 Hardware Definitions

The block diagram of the Handset is as followed:
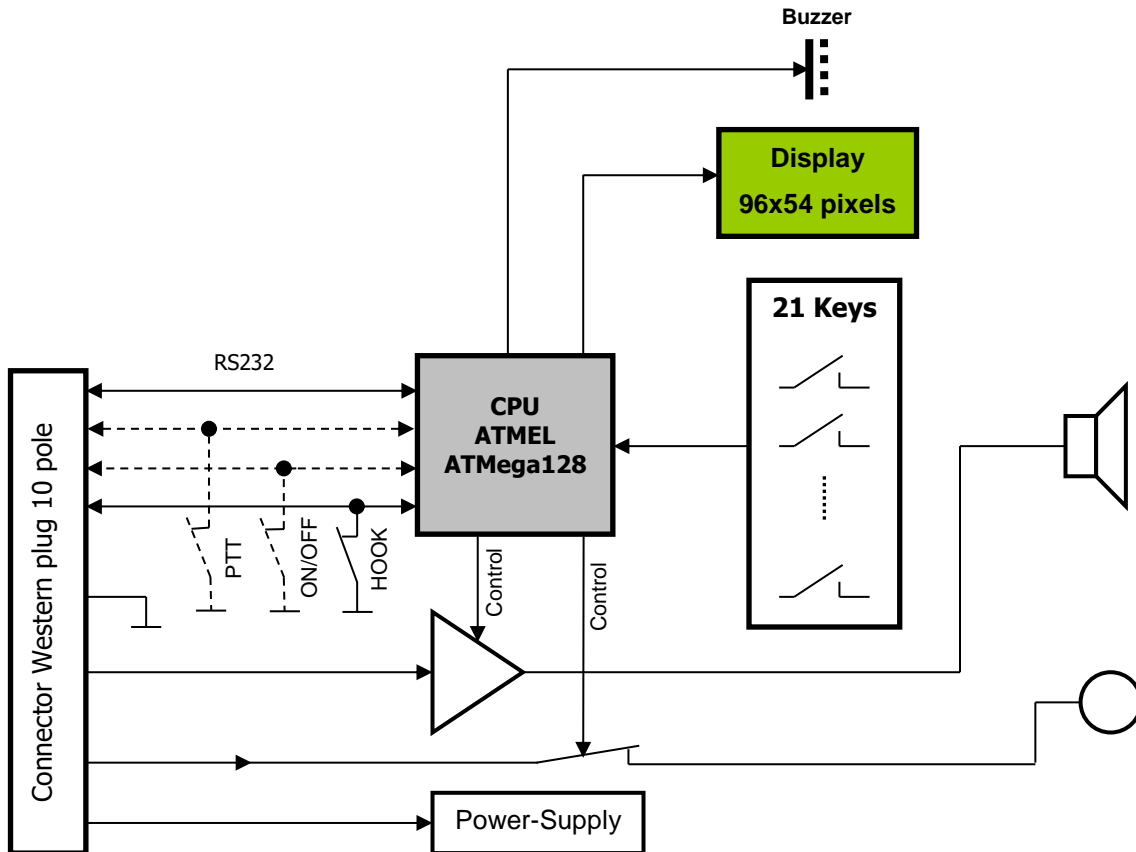
Figure 3: Block diagram

Depending on the model, PTT or ON/OFF are on PIN 1, see also table 3.

The 10-pole-Western-Plug has the following Pin-Out and forms the interface to your host-system:

PIN	Wire Description
1	PTT or ON/OFF
2	MIC -
3	MIC +
4	LS −
5	LS +
6	Hook (low active)
7	TxD
8	RxD
9	GND
10	VBATT (+9 … +15V)

Table 4: Wire description of the western plug

The settings for the serial connection are as shown below:

Baudrate:	115200 Baud default
	76800, 57600, 38400, 28800, 19200, 14400, 9600 optional
Parity:	None
Databits:	8
Stoppbits:	1
Handshake:	None

3 Terminal Protocol

As mentioned earlier there is a special protocol used for the communication between the client and the host. This byte-oriented protocol consists of the two parts

- Handset as Output
  The Handset sends "unsolicited" messages to the host after a key-event has occurred.

- Handset as Input
  The host can control the Handset via defined commands.

which are described in detail in this document.

3.1 Presentation Rules

In the following, the commands are presented with as much precision as possible. First there is a general description of the command providing information about the command behaviour. A "Syntax" paragraph describes the way to use the command. A "Parameter(s)" paragraph describes the required parameter value(s) where applicable. A "Returned Value(s)" paragraph provides possible responses of a command, where applicable. Finally, an "Example" paragraph provides an example of the command. Schemas are provided where necessary.

3.2 Command line

The byte-oriented protocol uses some sort of a frame. Commands always start with the <ESC>-Character ($0x1B_h$) and end with <CR> ($0x0D_h$) **or** <CR><LF> ($0x0D_h$ followed by $0x0A_h$). In this document both <CR> and <LF> are used.

<ESC>	Command	<CR>	<LF>
0x1B	Two or more Command-Bytes	0x0D	0x0A
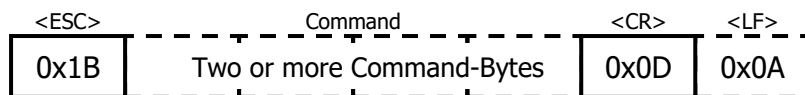
Figure 4: Schematic command

If a command is not correct or not understood, a question mark ('?') + <CR><LF> termination will be returned. That means, if a command was *formally* correct there is no response. But be aware that it does not mean the command was correct! Where possible, the question mark is returned upon invalid commands.

The following basic rules are applied:

- Control Characters (not printable characters) are placed within <>-brackets and are marked blue. For example, <ESC> equals the ESC-character which is $0x1B_h$. It is just one character! It is not ASCII 'E' 'S' 'C'!

- Variable values are placed within ()-brackets. These Values have to be ASCII-coded.

- Do not use spaces in commands unless it is quoted!

- <CR> represents "Carriage Return". The value of <CR> is $0x0D_h$.

- <LF> represents "Line Feed". The value of <LF> is $0x0A_h$.

Example: Set cursor position

<ESC>&H2;5<CR><LF>

This example consists of the following parts:

<ESC>	ESCAPE-Character 0x1B
&	Character '&'
H	Character 'H'
2	Variable Value for (Row)
;	Character ';'
5	Variable Value for (Column)
<CR>	Carriage Return
<LF>	Line Feed

The real byte stream for this example is:

<ESC>	&	H	2	;	5	<CR>	<LF>
0x1B	0x26	0x48	0x32	0x3B	0x35	0x0D	0x0A

This example will set the cursor to row 2, column 5. It does not turn on the cursor!

3.3 Handset as Output

The handset acts as output to notify the host-system about key-events when a key was pressed. Furthermore, some commands return a response.

3.3.1 The Keypad

The handset (client) acts as output to notify the host-system about key-events (pressed keys). Depending on the handset there are 20 or 21 keys available. The optional key is the REC/SOS-Key.

For the keys of the handset the following definitions are used:
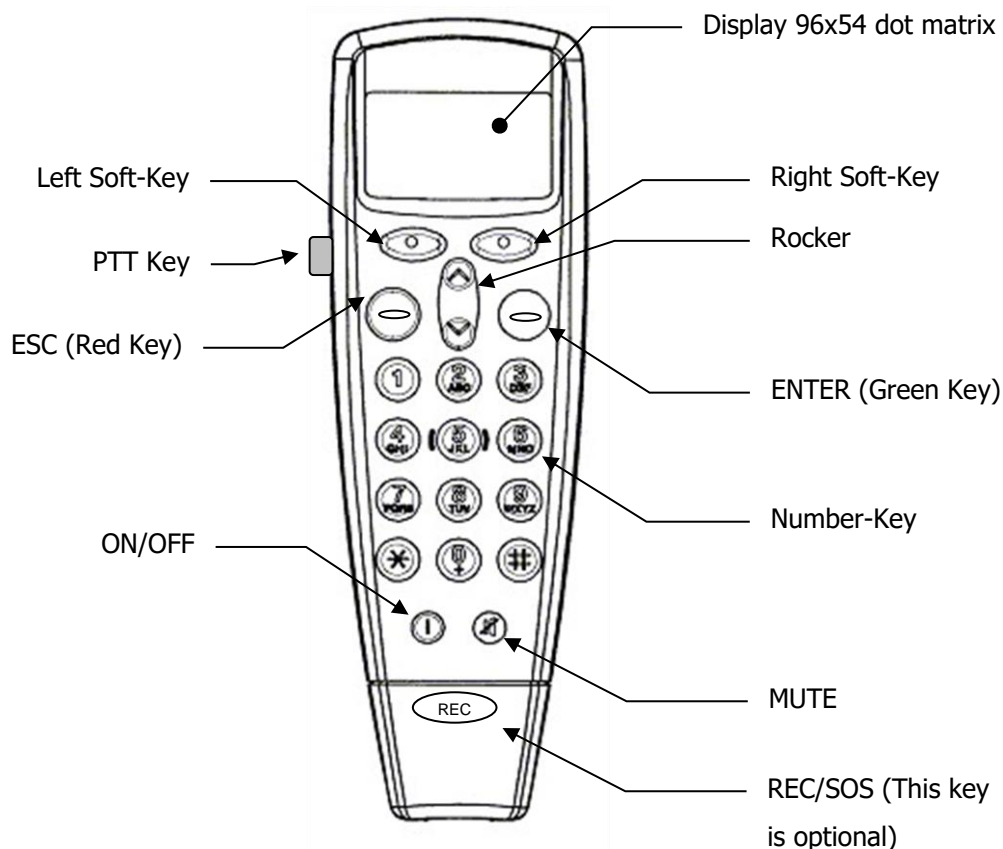
Figure 5: Keypad

The Hook-Switch is placed on the backside in the hole.

Remarks:
- The printing of the keypad can differ, but the definitions remain as defined above!
- Some buttons/keys might be missing on some handsets!

3.3.2 Key Events

There are 5 different Key Events possible:

- Start-Key-Event (SKE)
  This event occurs, when a key is pressed the first time.

- Long-Key-Event (LKE)
  After an adjustable period of time (*Time1*) the still pressed key will result in a the Long-Key-Event. Switches like the Hook-Switch or the PTT-Switch do not have a Long-Key-Event.

- Repeated-Key-Event (RKE)
  After an also adjustable period of time (*Time2*) a still pressed key will result in a Repeated-Key-Event. Switches like the Hook-Switch or the PTT-Switch do not have a Repeated-Key-Event.

- End-Key-Event (EKE)
  When a key is released an End-Key-Event occurs.

- False-Key-Event (FKE)
  When a key-error is detected, a False-Key-Event occurs. For example, pressing two keys at the same time can be a False-Key-Event.

3.3.3 Key-Event-Messages

For each Key-Event a Key-Event-Message is sent via the serial link to notify the host-system about the Key-Event.

In general, the structure of such a message is as follows:

- **Syntax**

  <ESC>K(Key)(Event)<CR><LF>   (6 Bytes)

- **Parameter(s)**

  Key: The Key-Code of the key, which has caused the event. Refer to the table in section 3.3.4 for the corresponding Key-Codes.

  Event: The event that has occurred according to the definitions following next.

According to section 3.3.2 for Key-Events the following ASCII-characters are assigned to (Event):

- Start-Key-Event   s
- Long-Key-Event   l   (low case "L")
- Repeated-Key-Event   r
- End-Key-Event   e

Therefore, the following Key-Event-Messages are sent when a Key-Event occurs:

- Start-Key-Event-Message (SKEM):   <ESC>K(Key)s<CR><LF>
- Long-Key-Event-Message (LKEM):   <ESC>K(Key)l<CR><LF>
- Repeated-Key-Event-Message (RKEM):   <ESC>K(Key)r<CR><LF>
- End-Key-Event-Message (EKEM):   <ESC>K(Key)e<CR><LF>
- False-Key-Event-Message (FKEM):   <ESC>KFx<CR><LF>

Because it is not possible to assign a Key when 2 keys are pressed at the same time, the False-Key-Event is an exception from the rule. This case can be detected by examining the message for the "F" after the <ESC> and "K". Any message containing the "F" has to be considered as invalid!

Example: For Key 1 the listed sequences will be sent when the Key is pressed and kept pressed for a while:

Event	ASCII-Stream	Hex-Stream (Real Bytestream)
SKE	<ESC>K1**s**<CR><LF>	0x1B, 0x4B, 0x31, 0x73, 0x0D, 0x0A
LKE	<ESC>K1**l**<CR><LF>	0x1B, 0x4B, 0x31, 0x6C, 0x0D, 0x0A
RKE	<ESC>K1**r**<CR><LF>	0x1B, 0x4B, 0x31, 0x72, 0x0D, 0x0A
RKE	<ESC>K1**r**<CR><LF>	0x1B, 0x4B, 0x31, 0x72, 0x0D, 0x0A
EKE	<ESC>K1**e**<CR><LF>	0x1B, 0x4B, 0x31, 0x65, 0x0D, 0x0A

Table 5: Response from the host system when pressing key "1"

3.3.4 Key-Codes

Please see the following table for the Key-Codes and Key-Messages (ASCII-coded):

Key	Code	SKEM	LKEM	RKEM	EKEM
Left-Soft-Key	L	KLs	KLl	KLr	KLe
Right-Soft-Key	R	KRs	KRl	KRr	KRe
ESC (red key)	E	KEs	KEl	KEr	KEe
ENTER (green key)	A	KAs	KAl	KAr	KAe
Rocker UP	U	KUs	KUl	KUr	KUe
Rocker DOWN	D	KDs	KDl	KDr	KDe
Key "1"	1	K1s	K1l	K1r	K1e
Key "2"	2	K2s	K2l	K2r	K2e
Key "3"	3	K3s	K3l	K3r	K3e
Key "4"	4	K4s	K4l	K4r	K4e
Key "5"	5	K5s	K5l	K5r	K5e
Key "6"	6	K6s	K6l	K6r	K6e
Key "7"	7	K7s	K7l	K7r	K7e
Key "8"	8	K8s	K8l	K8r	K8e
Key "9"	9	K9s	K9l	K9r	K9e
Key "0"	0	K0s	K0l	K0r	K0e
Key "*"	*	K*s	K*l	K*r	K*e
Key "#"	#	K#s	K#l	K#r	K#e
Key "ON/OFF"	O	KOs	KOl	KOr	KOe
Key "MUTE"	M	KMs	KMl	KMr	KMe
Key "REC/SOS"	N	KNs	KNl	KNr	KNe
Switch "HOOK"	H	KHs	n.a.	n.a.	KHe
Switch "PTT"	P	KPs	n.a.	n.a.	KPe

Table 6: Key-codes

Notice:

- The rest position of the HOOK-Switch is ON-HOOK! Removing the handset from the rest will cause the Start-Key-Event.
- The IT-command allows the user to switch off the LKEM and/or RKEM.
- **Example(s)**
  The following timing-diagram shows the keycodes which are sent for the **"Key 1"**.
  When the key is pressed, the Start-Key-Event occurs and the corresponding message is sent. If the key is kept pressed, the Long-Key-Event occurs after the time *Time1*. If the key is still kept pressed, the Repeated-Key-Event occurs and will cause the Repeated-Key-Event-Message (RKEM) to be sent. From now on there will be always the RKEM sent when the time *Time2* has expired and the key is still pressed. Finally, when the key is released, the End-Key-Event will occur and the End-Key-Event-Message will be sent.
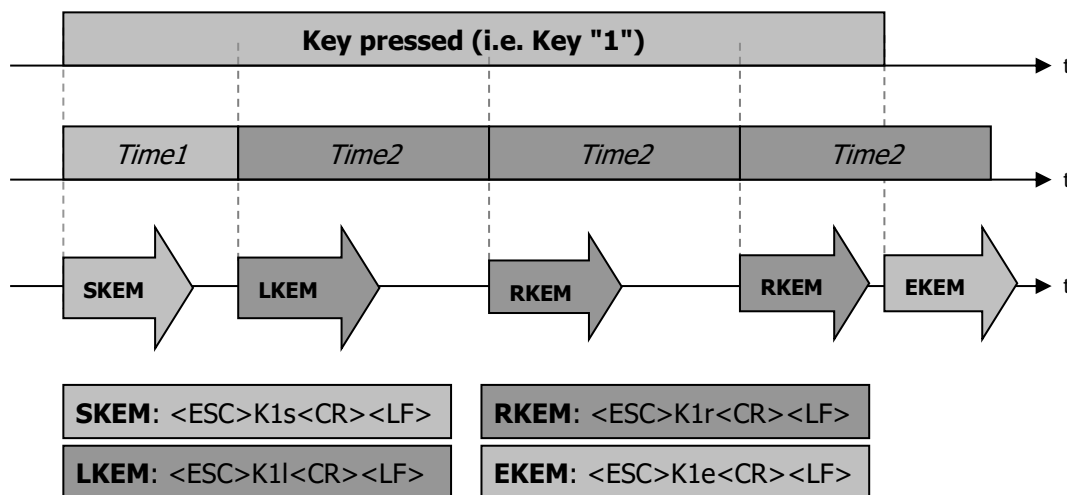
Figure 6: Timing diagram key 1 pressed

In the following timing diagram, a second key "Key 2" is pressed while the "Key 1" is pressed. The period of time both keys are pressed, is considered to be invalid. Please study the diagram on the next page for the timing and messages sent when two keys are pressed at the same time.

Notice, that no End-Key-Event-Message (EKEM) for "Key 1" will be sent! Any key message containing a "F" has to be considered as invalid. The next valid key message will be a Start-Key-Event-Message (SKEM).

SKEM 1: <ESC>K1s<CR><LF>   **SKEM 2**: ESC>K2s<CR><LF>

LKEM 1: <ESC>K1l<CR><LF>   **LKEM 2**: <ESC>K2l<CR><LF>

FKEM s: <ESC>KFs<CR><LF>   **EKEM 2**: ESC>K2e<CR><LF>

FKEM l: <ESC>KFl<CR><LF>
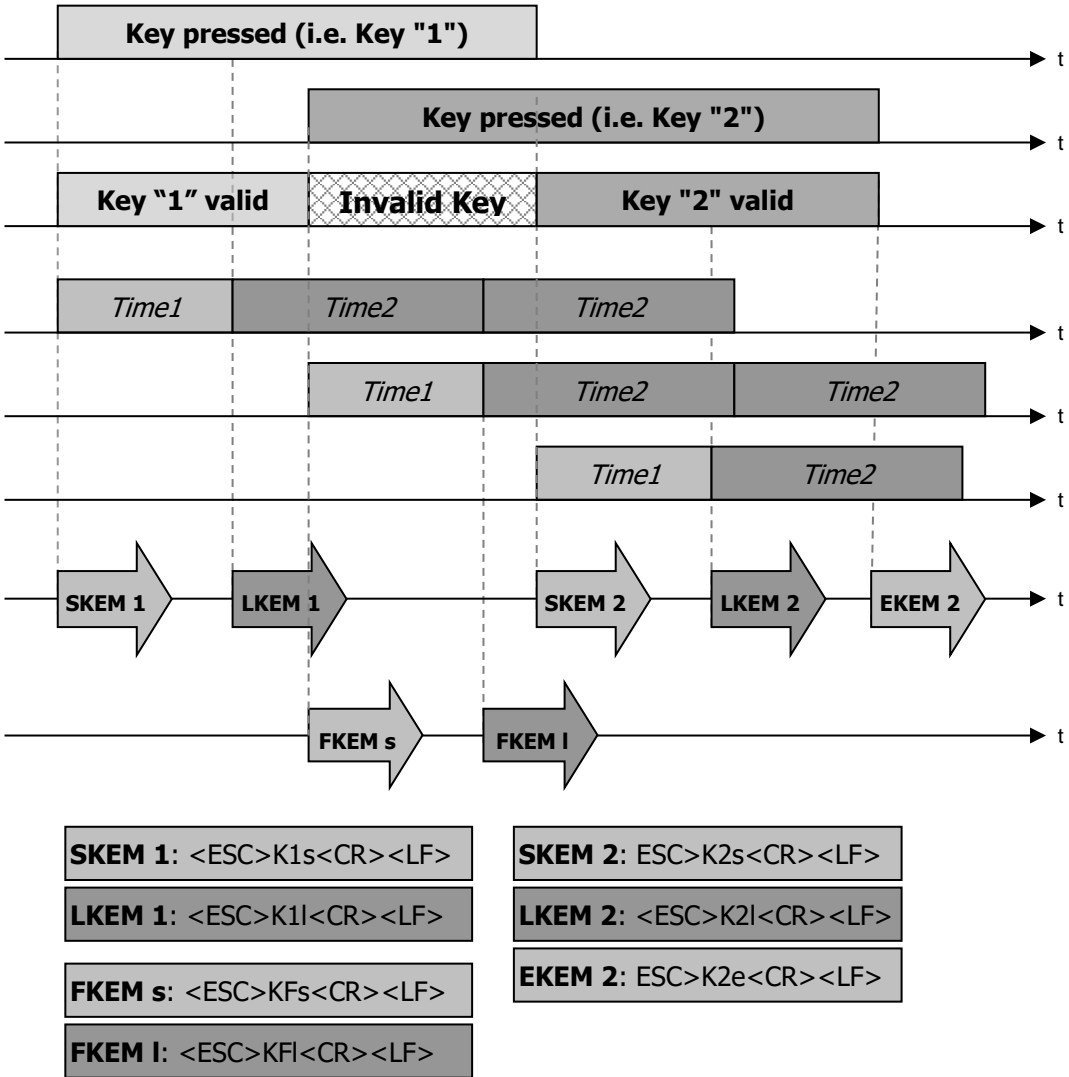
Figure 7: Timing diagram two buttons pressed

3.3.5 Power-On

When the Handset is connected to the Power-Supply, the sequence

<ESC>INIT<CR><LF>

is sent via the serial link with the selected baudrate. At delivery this baudrate is 115200. This sequence can be used to detect the moment, when the Handset is connected to the host-system.

Important:

When the Handset is connected to the power supply, the internal bootloader will send a version string with 115200 Baud regardless the selected Baud rate of the Terminal-Software. This has production reasons and can not be changed! The host-system has to check for the INIT-string to detect the correct initialization of the Terminal-Software.

3.4 Display

Before the control-commands are described there is some background about the structure of the display necessary. The following sections will provide these basics.

3.4.1 Display Organization

There are currently 3 different splittings of the display implemented. Some of them can have different display-modes. Furthermore, there are also several graphical modes available, which allow to use the display (or parts of it) for graphical purposes.

The display-splittings #1 to #3 are intended for the displaying of ASCII-text , while the graphical modes are intended to display anything you want – which can also include ASCII-Text.

Splitting #1	Splitting #2	Splitting #3
ICON		**ICON**
		Text 1*16 Characters
Text 4*16 Characters	**Text 5*16 Characters**	**8-big-size-chars.**
		not used
SoftKeyL / **Rocker** / **SoftKeyR**	**SoftKeyL** / **Rocker** / **SoftKeyR**	**SoftKeyL** / **Rocker** / **SoftKeyR**

- **DISP_MODE_04_16**
- **DISP_MODE_04_16_SMS**

- **DISP_MODE_05_16**
- **DISP_MODE_05_16_SMS**

- **DISP_MODE_BIG_SIZE**
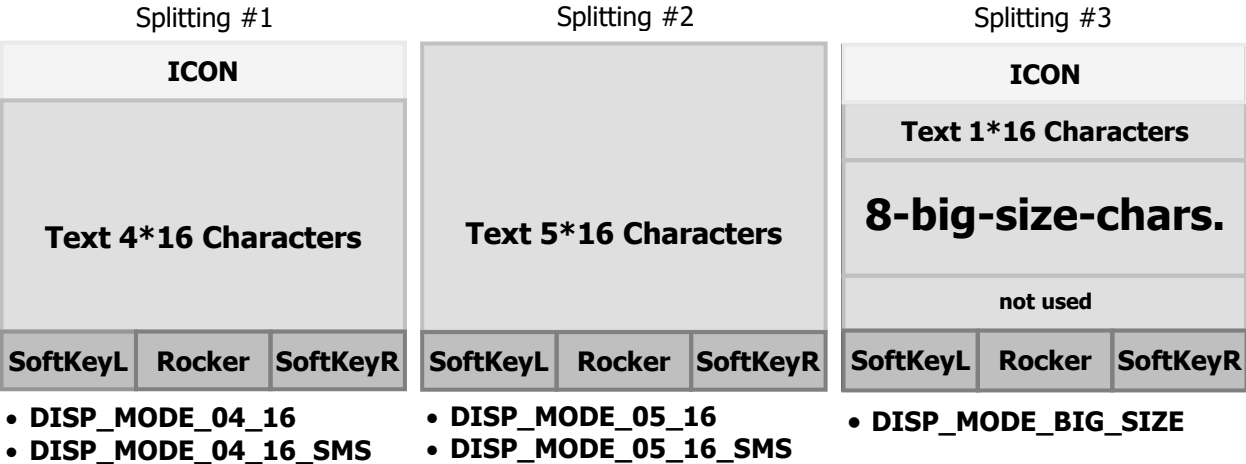
Figure 8: Display organization

Graphic Modes

Row 0
Row 1
Row 2
Row 3
Row 4
Row 5
Row 6

- **DISP_MODE_GR_4ROWS**
- **DISP_MODE_GR_5ROWS**
- **DISP_MODE_GR_6ROWS**
- **DISP_MODE_GR_7**
- **DISP_MODE_GR_8**
- **DISP_MODE_GR_9**
- **DISP_MODE_GR_10**
- **DISP_MODE_GR_11**
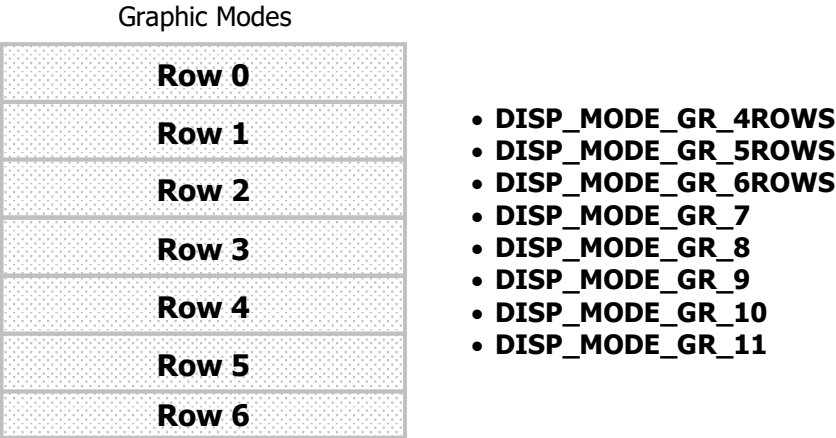
Figure 9: Graphic modes

The display can have the following components depending on the Display-Mode selected:

- **Icon-Bar (ICON)**
- **Text-Box (Text)**
- **Soft-Key-Bar (SoftKeyL – Rocker – SoftKeyR)**

When a graphic mode is selected it can be imagined as a transparency which are put on top of the current display. The figure below shows the principle of the graphic modes.
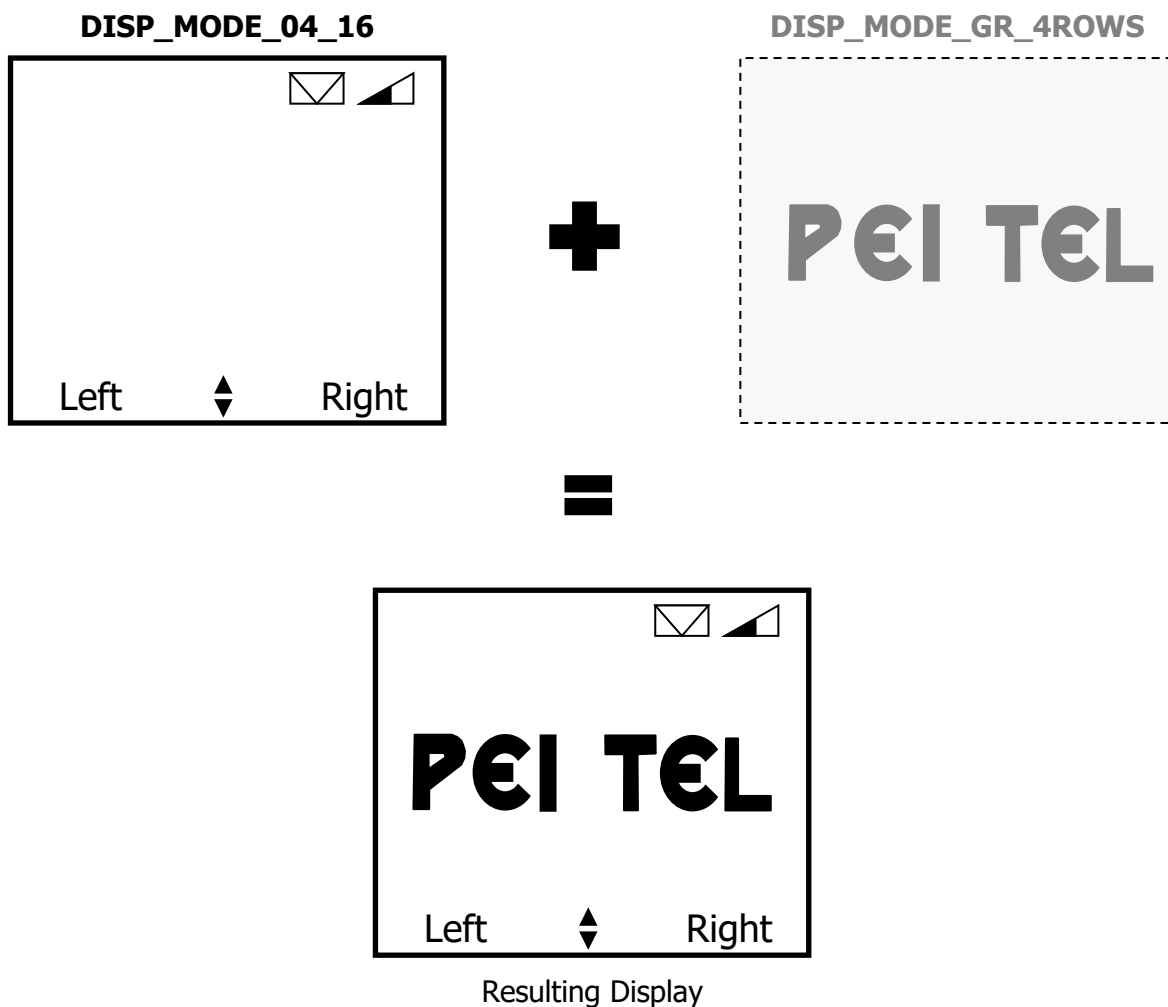
Figure 10: Graphic mode – imagined as transparencies

For example, Arabian characters could be displayed with the graphical modes. Because of the behaviour like a transparency the Icon-Bar and/or Softkey-Bar can still be visible while the Text-Box could show a graphic. The graphical modes are very useful – therefore, an own section deals with these modes.

3.4.2 Character-Sets

For displaying text there are two character-sets implemented:

- SMS
- TB

A character-set following the recommendation of the GSM-Standard 3.38 is used in SMS-modes. SMS may be processed almost directly due to encoding. In the other modes except the graphical modes the TB-character-set is used. Please see the appendix for detailed tables of the two character-sets.

3.4.3 Icon-Bar

There are already a lot of Icons implemented. To display these Icons the Icon-Bar of the display is divided into 5 Panels, where the Icons will be displayed. An Icon can be displayed in its defined position (Panel) only.

Signal-Strength	Missed-Calls	Roaming	SMS	Volume

Figure 11: Position(s) of certain Icons in the ICON-Bar

The Icons will be displayed in the display-modes DISP_MODE_04_16, DISP_MODE_04_16_SMS and in graphical modes not covering the first row (line) only. Icons may always be set. When switching the display to one of the modes mentioned above, the values last set are displayed.

For example, you have set in Display-Mode DISP_MODE_04_16 the Icon "Signal-Strength". Then you switch to the Display-Mode DISP_MODE_05_16 for displaying another text. When you switch back to DISP_MODE_04_16 the Icon "Signal-Strength" will be displayed again. You don't have to set it again.

The layout of the icons is shown in the sections describing the corresponding command(s).

3.4.4 Soft-Key-Bar

The Soft-Key-Bar is located in in lower part of the display and is devided into the following three Panels:

- SoftKeyL
- SoftKeyR
- Rocker

SoftKeyL	Rocker	SoftKeyR
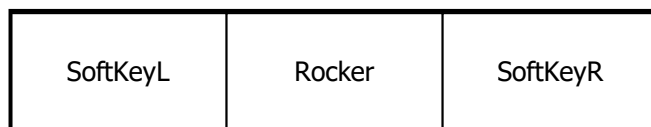
Figure 12: Soft-Key-Bar

While SoftKeyL and SoftkeyR are Text-Panels, the Rocker-Panel is for an Icon in non-graphical-modes. The character-set of the Soft-Key-Bar is TB. In some graphical modes the SoftKeyL-Panel and Soft-KeyR-Panel can be also used for graphical purposes while the Rocker remains in the display. The Soft-Keys and the Rocker are the Keys right beneath the display.

3.5 Handset as Input (Commands)

There are several commands defined to control the handset. They have to be sent to the Handset via the serial link with the selected communication settings, which are at delivery 115200,N,8,1. Commands always start with the <ESC> character and end with <CR> (0x0D$_h$) **or** <CR><LF> (0x0D$_h$ followed by 0x0A$_h$). In this document the combination <CR> and <LF> is used.

Upon invalid commands "? <CR><LF>" is returned.

3.5.1 Cursor ON/OFF

This command switches the cursor on or off. The cursor is supported in the display-modes DISP_MODE_04_16, DISP_MODE_04_16_SMS, DISP_MODE_05_16 and DISP_MODE_05_16_SMS only.

- **Syntax**

  <ESC>&C(Mode)<CR><LF>

- **Parameter(s)**

  Mode: 0   Switch off the cursor
        1   Turn on the cursor

- **Example(s)**

  <ESC>&C0<CR><LF>

3.5.2 Clear Display

This command clears the display in non-graphical modes and sets the cursor to the home-position of the selected display-mode.

- **Syntax**

  <ESC>&D<CR><LF>

- **Parameter(s)**
  None

- **Example(s)**

  <ESC>&D<CR><LF>

3.5.3 Set Cursor Position

With this command you can set the cursor to a certain position. Text always starts at the cursor-position. This command does not switch on the cursor! To switch on/off the cursor separate commands are used. When the display-mode is changed in non-graphical-modes, the cursor is set to the home-position of the selected display-mode. If the text written to the display is longer than the remaining space, then the text will be cut to fit in the display.

- **Syntax**

  <ESC>&H(Row);(Column)<CR><LF>

- **Parameter(s)**

  Row: The row the cursor is set to. The range depends on the selected display-mode. For example, if a display-mode with 5 rows (lines) is selected, then the range is from 0 to 4.

  Column: The Column the cursor is set to. The range is 0 to 15. A value > 15 results in discarding the entire command. The Parameter is ASCII-coded, that means Column 15 contains of the two Bytes "1" which is 0x31 and "5" which is 0x35!

- **Example(s)**

  <ESC>&H3;15<CR><LF>

  This example sets the cursor to row 3 (line 3) and column 15.

3.5.4 Get Cursor Position

With this command you can get the current cursor position.

- **Syntax**

  <ESC>&H?<CR><LF>

- **Parameter(s)**

- **Returned Value(s)**

  <ESC>&H: (Row);(Column)<CR><LF>

- **Example(s)**

  <ESC>&H?<CR><LF>   Request current cursor position

  Returned response could be (current cursor position is row 3, column 15):

  <ESC>&H: 3;15<CR><LF>

3.5.5 Clear Line

This command clears the line from the current cursor position to the end of the line. To clear an entire line, the cursor has to be set to the start of the line first.

- **Syntax**

  <ESC>&K<CR><LF>

- **Parameter(s)**
  None

- **Example(s)**

  <ESC>[3;0H<CR><LF> Set Cursor to 3;0 (Line 3, Column 0)

  <ESC>&K<CR><LF>   Clear line 3

3.5.6 Restart Handset

This command performs a software restart of the Handset. The bootloader string will not be sent.

- **Syntax**

  <ESC>&O0<CR><LF>

- **Parameter(s)**
  None

- **Example(s)**

  <ESC>&O0<CR><LF>

3.5.7 Get Serial Number

With this command you can get the serial number of the Handset, if one is programmed.

- **Syntax**

  <ESC>&S?<CR><LF>

- **Parameter(s)**

- **Returned Value(s)**

  <ESC>&S: (serial number)<CR><LF>

  serial number: serial number, if one is programmed

        ERROR, if serial number is not programmed

- **Example(s)**

  <ESC>&S?<CR><LF>   Request serial number

  Returned response could be (Serial number is 1, production date is 12.02.07)

  <ESC>&S: 0000000001/12.02.07<CR><LF>

3.5.8 Get Version

This command is used to request the string of the SW-version of the Handset.

- **Syntax**

  <ESC>&V?<CR><LF>

- **Parameter(s)**
  None

- **Returned Value(s)**
  The current SW-Version in the format:

<ESC>&V: (Version)<CR><LF>

- **Example(s)**

   <ESC>&V?<CR><LF>   Request version

   Returned response could be:
    <ESC>&V: HA400 V.01.00 25.09.2006<CR><LF>

3.5.9 Get HOOK-State

This command is used to get the current state of the HOOK-switch. With this switch it can be detected whether the handset is On-Hook or Off-Hook. The host-system could then perform corresponding actions like switching the audio-paths.

- **Syntax**

   <ESC>KH?<CR><LF>

- **Parameter(s)**
   None

- **Returned Value(s)**
   The current state of the HOOK-switch:

   <ESC>KH: (State)<CR><LF>

   State: H   Handset is On Hook
             h   Handset is Off Hook

- **Example(s)**

   <ESC>KH?<CR><LF>   Request HOOK-State

   Returned response could be (Handset is On Hook):

    <ESC>KH: H<CR><LF>

3.5.10 Get PTT-State

This command is used to request the current state of the PTT-switch.

- **Syntax**

  <ESC>KP?<CR><LF>

- **Parameter(s)**
  None

- **Returned Value(s)**
  The current state of the PTT-switch:

  <ESC>KP: *(State)<CR><LF>*

  State: P   PTT-switch is pressed
          p   PTT-switch is released

- **Example(s)**

  <ESC>KP?<CR><LF>   Request PTT-State

  Returned response could be (PTT is released):

  <ESC>KP: p<CR><LF>

3.5.11 Set Contrast

With this command it is possible to set the contrast of the display. The value of the contrast is stored in EEPROM permanently.

- **Syntax**

  <ESC>IA(Value)<CR><LF>

- **Parameter(s)**
  Value: Value for the contrast in the range 1 (bright) to 20 (dark).

- **Example(s)**

  <ESC>IA10<CR><LF>   Set contrast to a middle value

3.5.12 Get Contrast

This command is used to retrieve the current contrast value.

- **Syntax**

  <ESC>IA?<CR><LF>

- **Parameter(s)**
  None

- **Returned Value(s)**
  The current contrast value in the format:

  <ESC>IA: (Value)<CR><LF>

  Attention: The returned value of the contrast is ASCII-coded. For values greater than 9 there are two bytes returned for the current setting.

- **Example(s)**

  <ESC>IA?<CR><LF>   Request current contrast setting

  Returned response could be (current contrast setting is 10 in this case):

  <ESC>IA: 10<CR><LF>

3.5.13 Increase Contrast

With this command it is possible to increase the contrast by 1 step. When the max value (20) has already been reached, a "? <CR><LF>" is returned. The new value will also be stored in EEPROM.

- **Syntax**

  <ESC>IA+<CR><LF>

- **Parameter(s)**
  None

- **Example(s)**

  <ESC>IA+<CR><LF>   Increase contrast

3.5.14 Decrease Contrast

With this command it is possible to decrease the contrast by 1 step. When the min value (0) has already been reached, a "? <CR><LF>" is returned. The new value will also be stored in EEPROM.

- **Syntax**

  <ESC>IA-<CR><LF>

- **Parameter(s)**
  None

- **Example(s)**

  <ESC>IA-<CR><LF>   Decrease contrast

3.5.15 Set Buzzer

With this command it is possible to switch on the buzzer for a certain time in steps of 10 ms, if a buzzer is available.

- **Syntax**

  <ESC>IB(Value)<CR><LF>

- **Parameter(s)**
  Value: Value for the buzzer-time in the range 0 (short) to 255 (long).

- **Example(s)**

  <ESC>IB10<CR><LF>   Switch on buzzer for 100 ms

3.5.16 Select Display-Mode

This command selects the display-mode and turns on/off the display. In a non-graphical-mode the display-mode has to be selected **_prior_** dealing with the display (for example writing text to it). In the graphical modes the graphic to be displayed has to be transferred first to the Handset and then the mode has to be selected **_afterwards_**. The reason for that behaviour is, that it may take some time to transfer a graphic. So the display will be updated when the whole graphic has arrived in the Handset. This avoids seeing just a part of the graphic, then the next part etc.

- **Syntax**

  <ESC>IDM(Mode)<CR><LF>

- **Parameter(s)**

Mode:	D	Turn off display
	E	Turn on display
	0	DISP_MODE_04_16
	1	DISP_MODE_04_16_SMS
	2	DISP_MODE_05_16
	3	DISP_MODE_05_16_SMS
	4	DISP_MODE_BIG_SIZE
	5	DISP_MODE_GR_4ROWS
	6	DISP_MODE_GR_5ROWS
	7	DISP_MODE_GR_6ROWS
	8	DISP_MODE_GR_7
	9	DISP_MODE_GR_8
	10	DISP_MODE_GR_9
	11	DISP_MODE_GR_10
	12	DISP_MODE_GR_11

- **Example(s)**

  <ESC>IDM0<CR><LF>   Select display-mode DISP_MODE_04_16

3.5.17 Get Display-Mode

This command is used to retrieve current display Mode.

- **Syntax**

  <ESC>IDM?<CR><LF>

- **Parameter(s)**
  None

- **Returned Value(s)**
  Current display mode according to the list in [section 3.5.16](#) in the format:

  <ESC>IDM: (Mode)<CR><LF>

- **Example(s)**

  <ESC>IDM?<CR><LF>   Request current display mode

  Returned response could be (current display mode is 10, which is DISP_MODE_GR_9):

  <ESC>IDM: 10<CR><LF>

3.5.18 Set Backlight-Mode/-Time

This command is used to configure the backlight of the display in the Handset. There are 5 basic modes available:

- Normal-Mode (automatic)   Mode 0
- Normal-Mode (permanent)   Mode 1
- PWM-Mode (automatic)   Mode 2
- PWM-Mode (permanent)   Mode 3
- Permanently OFF   Mode 4

In PWM-Modes, the brightness is adjustable by the IN-command. In Normal-Modes the brightness can not be adjusted. The brightness is then as high as possible.

Automatic means that the backlight is switched on when a key is pressed. After an adjustable period of time, the backlight is switched off automatically. With every key-press the backlight-timer is restarted. It is also possible that the backlight of the keypad remains ON all the time and just the display-backlight is turned off after the period of time has expired. This can be achieved by setting a flag in Normal-Mode (automatic).

Permanent means, that the backlight remains on all the time.

At delivery, the Backlight-Mode is PWM-Mode (Automatic) – Mode 2 and Time is 30 sec.

- **Syntax**

  <ESC>IE(Mode)<CR><LF>　　　　　Sets Mode

  <ESC>IE(Time)<CR><LF>　　　　　Sets Time

- **Parameter(s)**

  Mode:　　Backlight-Mode – the following values are supported:

  - 0　　　　Backlight is switched off permanently (Mode 4)

  - E　　　　Backlight is switched on permanently (Mode1)

  - A　　　　Selects Normal-Mode (automatic) and turns on backlight
    for the stored value of Time n (Mode 0)

  - S　　　　Selects Normal-Mode (automatic). The Backlight is not
    turned on immediately (Mode 0)

  - P　　　　Backlight is switched on permanently in PWM-Mode.
    The brightness can be set with the IN-command (Mode 3).

  - Q　　　　Selects PWM-Mode (automatic). Brightness can be set
    with the IN-command (Mode 2)

  - R　　　　Restarts the backlight with the selected parameters (Mode
    and Time). For both Normal- and PWM-Modes.

  - e　　　　Sets the flag for the permanent backlight of the keypad. Can be set
    in Normal-Mode (automatic) only (Mode 0). In PWM-Modes a "?" is
    returned, when this command is sent. Is not a Mode!

  - r　　　　Clears the flag for the permanent backlight of the keypad
    and restarts the backlight. Can be used in Normal-Mode (automatic)
    only (Mode 0). In PWM-Modes a "?" is returned, when this command
    is sent. Is not a Mode!

  - s　　　　Same as r, but backlight is not restarted.

  - Time　　Sets the duration of the backlight to a user-defined
    value in steps of 1 s. Value is stored in EEPROM permanently.
    Range is 1 to 250 sec.

- **Example(s)**

  <ESC>IE0<CR><LF>　　　　　　　Switch OFF backlight permanently

  <ESC>IE10<CR><LF>　　　　　　　Set Time to 10 sec

3.5.19 Get Backlight-Mode/-Time

Command to retrieve the current backlight mode and time.

- **Syntax**

  <ESC>IE?<CR><LF>

- **Parameter(s)**

- **Returned Value(s)**

  <ESC>IE: (Mode);(Time)<CR><LF>

Mode:	0	Normal-Mode (automatic)
	1	Normal-Mode (permanent)
	2	PWM-Mode (automatic)
	3	PWM-Mode (permanent)
	4	OFF

  Time: Time in sec

- **Example(s)**

  <ESC>IE?<CR><LF>   Request backlight settings

  Returned response could be (Mode 2, 15 seconds):

  <ESC>IE: 2;15<CR><LF>

3.5.20 Set Icon "Signal-Strength"

With this command it is possible to show the icon for "Signal-Strength". The icon for the signal strength is located in the Signal-Strength-Panel of the Icon-Bar and looks like shown below.

Figure 13: Icon "Signal Strength"

- **Syntax**

  <ESC>IF(State)<CR><LF>

- **Parameter(s)**

  State: 0   Switch off icon "Signal-Strength"

             1-6   Switch on icon "Signal-Strength" with corresponding value

                  1:      Display icon without signal-bars

                  6:      Display icon with 5 signal-bars

- **Example(s)**

  <ESC>IF3<CR><LF>   Switch on icon "Signal-Strength" with 2 bars

3.5.21 Set Icon "Volume Handsfree-Mode"

There are 2 volume-icons plus an icon for muting available. There are the following rules applied:

- Just one can be displayed at the time.

- It will be displayed in the Volume-Panel of the Icon-Bar.

- The icon "Mute" overrules the 2 volume-icons!

- A volume-icon can not be displayed if the icon "Mute" is active. When switching on an volume-icon and icon "Mute" was (or is) active, then icon "Mute" will be displayed. Nevertheless, the value of the icon can be updated.

- When switching off the icon "Mute", the last displayed volume-icon will be displayed again with its current value – if there was previously one displayed.

Private-Mode   Hands-free-Mode   Mute

Figure 14: Volume icons

- **Syntax**

  <ESC>IJ(Value)<CR><LF>

- **Parameter(s)**

Value:	0	Switch off all volume-icons (inclusive icon "Mute" – but if icon "Mute" was active it will be displayed again, when a volume-icon is switched on!)
	1-10	Activate icon "Volume Handsfree-Mode" with corresponding value. The icon will be displayed only if icon "Mute" is not active! If icon "Mute" is active, icon "Mute" overrules the volume-icons and will be displayed.

  1:    Display icon with min value (triangle empty)
  10:   Display icon with max value (triangle full)

- **Example(s)**

  <ESC>IJ5<CR><LF>   Switch on icon "Volume Handsfree-Mode"
  with a middle value (only if icon "Mute" is not active).
  If icon "Mute" is (was) active, then the icon "Mute" will
  be displayed – but we can update the value anyway.

3.5.22 Clear Soft-Keys

With this command it is possible to clear both SoftKeyL-Panel and SoftKeyR-Panel at once.

- **Syntax**

  <ESC>IK0<CR><LF>

- **Parameter(s)**
  None

- **Example(s)**

  <ESC>IK0<CR><LF>   Clear both Soft-Key-Panels

3.5.23 Set Left-Soft-Key

With this command it is possible to write some text in the Panel "SoftKeyL" of the Soft-Key-Bar. The text is limited to 7 characters.

- **Syntax**

    <ESC>IK1(Text)<CR><LF>

- **Parameter(s)**
    Text: The text to be placed in the SoftKeyL-Panel (0 – 7 characters)

- **Example(s)**

    <ESC>IK1Left<CR><LF>   Write "Left" to SofKeyL-Panel

3.5.24 Set Right-Soft-Key

With this command it is possible to write some text in the Panel "SoftKeyR" of the Soft-Key-Bar. The text is limited to 7 characters.

- **Syntax**

    <ESC>IK2(Text)<CR><LF>

- **Parameter(s)**
    Text: The text to be placed in the SoftKeyR-Panel (0 – 7 characters)

- **Example(s)**

    <ESC>IK2Right<CR><LF>   Write "Right" to SofKeyR-Panel

    <ESC>IK2<CR><LF>   Clear SoftKeyR-Panel

3.5.25 Set Icon "Volume Private-Mode"

The Icon "Volume Private-Mode" is member of the volume-icons. For general rules see section 3.5.21.

- **Syntax**

  <ESC>IL(Value)<CR><LF>

- **Parameter(s)**

Value:	0	Switch off all volume-icons (inclusive icon "Mute" – but if icon "Mute" was active it will be displayed again, when a volume-icon is switched on!)
	1-10	Activate icon "Volume Private Mode" with corresponding value. The icon will be displayed only if icon "Mute" is not active! If icon "Mute" is active, icon "Mute" overrules the volume-icons and will be displayed.

1:	Display icon with min value (triangle empty)
10:	Display icon with max value (triangle full)

- **Example(s)**

  <ESC>IL5<CR><LF>   Switch on icon "Volume Private-Mode" with a middle value (only if icon "Mute" is not active)

  If icon "Mute" is (was) active, then the icon "Mute" will be displayed – but we can update the value anyway.

3.5.26 Set Icon "Mute"

The Icon "Mute" is member of the volume-icons. For general rules see section 3.5.21. Whether the microphone-path is muted or not depends on the hardware of the Handset.

- **Syntax**

  <ESC>IM(State)<CR><LF>

- **Parameter(s)**

State:	0	Switch off icon "Mute" – if a volume-icon was activated previously it will now be shown
	1	Activate and display icon "Mute". A volume-icon will not be displayed anymore – but it remains active!

- **Example(s)**

  <ESC>IM1<CR><LF>   Switch on icon "Mute".

3.5.27 Set Brightness (PWM)

The brightness of the backlight is controlled by a PWM-signal. The duty-cycle of the PWM can be set by this command. The range is from 0 to 100%. The higher the value, the higher the brightness. For PWM-Modes only (Mode 2 and Mode 3). The value of the PWM is stored in EEPROM permanently.

- **Syntax**

  <ESC>IN(Value)<CR><LF>

- **Parameter(s)**
  Value: Value for the PWM in the range 0% (dark) to 100% (bright).

- **Example(s)**

  <ESC>IN50<CR><LF>   Set brightness to a middle value

3.5.28 Get Brightness (PWM)

This command is used to retrieve the current brightness value for PWM-Modes.

- **Syntax**

  <ESC>IN?<CR><LF>

- **Parameter(s)**
  None

- **Returned Value(s)**
  The current brightness value in the format:

  <ESC>IN: (Value)<CR><LF>

  Attention: The returned value of the brightness is ASCII-coded. For values greater than 9 there are two bytes returned for the current setting.

- **Example(s)**

  <ESC>IN?<CR><LF>   Request current brightness setting

  Returned response could be (current brightness setting is 50):

  <ESC>IN: 50<CR><LF>

3.5.29 Increase Brightness (PWM)

With this command it is possible to increase the brightness by 1 step. When the max value (100) has already been reached, a "? <CR><LF>" is returned. The new value will also be stored in EEPROM.

- **Syntax**

    <ESC>IN+<CR><LF>

- **Parameter(s)**
  None

- **Example(s)**

    <ESC>IN+<CR><LF>   Increase brightness

3.5.30 Decrease Brightness (PWM)

With this command it is possible to decrease the brightness by 1 step. When the min value (0) has already been reached, a "? <CR><LF>" is returned. The new value will also be stored in EEPROM.

- **Syntax**

    <ESC>IN-<CR><LF>

- **Parameter(s)**
  None

- **Example(s)**

    <ESC>IN-<CR><LF>   Decrease brightness

3.5.31 Set Icon "Missed Calls"

The Icon "Missed Calls" is located in the Missed-Calls-Panel of the Icon-Bar and is a number.

- **Syntax**

    <ESC>IP(Value)<CR><LF>

- **Parameter(s)**
  Value: 0   Switch off icon "Missed Calls"
               1-19   Switch on icon "Missed Calls" with corresponding Value

- **Example(s)**

    <ESC>IP10<CR><LF>   Switch on icon "Missed Calls" with Value 10

3.5.32 Set Icon "Roaming"

The Icon "Roaming" is located in the middle of the Icon-Bar and looks like shown below.

Figure 15: Roaming icon

- **Syntax**

  <ESC>IR(State)<CR><LF>

- **Parameter(s)**
  State: 0 Switch off icon "Roaming"
  1 Switch on icon "Roaming"

- **Example(s)**

  <ESC>IR1<CR><LF> Switch on icon "Roaming"

3.5.33 Set Icon "Read-SMS"

With this command it is possible to show the icon for "Read SMS". The Icons "SMS" are located in the SMS-Panel of the Icon-Bar. An unread SMS has a higher priority than a read SMS. If the Icons "Unread SMS" and "Read SMS" are switched on, then only the Icon for unread SMS is displayed. After switching off the "Unread SMS", the Icon for "Read SMS" will be displayed. Icons can be displayed in display-modes containing the Icon-Bar only. When an icon is displayed it will be displayed as long as it is switched off – even if the display-mode is changed.

Figure 16: "Read SMS" icon

- **Syntax**

  <ESC>IS(State)<CR><LF>

- **Parameter(s)**
  State: 0 Switch off icon "Read SMS"
  1 Switch on icon "Read SMS"

- **Example(s)**

  <ESC>IS1<CR><LF> Switch on icon "Read SMS"

3.5.34 Set Key-Press-Times

This command is used to set the times *Time1* and *Time2* for the different Key-Press-Events SKE, LKE, RKE and EKE. The Key-Press-functionality has already been described in chapter 3.3. These times are permanently stored in an EEPROM.

- **Syntax**

  <ESC>IT(Time1);(Time2)<CR><LF>

- **Parameter(s)**

  Time1:   *Time1* in steps of 100 ms. Valid values are in the range between 4 and 50 which corresponds to 0.4 s and 5.0 s. The default value is 12 (1.2 s).

  Time2:   *Time2* in steps of 100ms. Valid values are in the range between 4 and 50 which corresponds to 0.4 s and 5.0 s. The default value is 12 (1.2 s).

  If *Time2* is set to 0, then the Repeated-Key-Event-Message (RKEM) is switched off. If *Time1* and *Time2* are set to 0, Long-Key-Event-Message (LKEM) and Repeated-Key-Event-Message (RKEM) are switched off. Only Start-Key-Event-Message (SKEM) and End-Key-Event-Message (EKEM).

- **Example(s)**

  <ESC>IT0;0<CR><LF>

  This example sets the Time1 and Time2 to 0 and switches off LKEM and RKEM.

3.5.35 Get Key-Press-Times

Command to retrieve the current key-time settings.

- **Syntax**

  <ESC>IT?<CR><LF>

- **Parameter(s)**

- **Returned Value(s)**

  <ESC>IT: (Time1);(Time2)<CR><LF>

  Time1:   Time, when Long-Key-Event (LKE) will occur, in steps of 100 ms

  Time2:   Time, when Repeated-Key-Event (RKE) will occur after LKE, in steps of 100 ms

- **Example(s)**

    <ESC>IT?<CR><LF>   Request key-press-times settings

    Returned response could be (LKEM and RKEM switched off):

    <ESC>IT: 0;0<CR><LF>

3.5.36 Set Icon "Unread-SMS"

With this command it is possible to show the icon for "Unread SMS". The icon "Unread SMS" can be displayed in a flashing-mode, too. When the flashing-mode is selected, the icon "Read SMS" should be deactivated first (if it was activated) -otherwise it could mix-up with the icon "Read SMS".
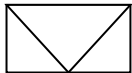
Figure 17: Icon "Unread SMS"

- **Syntax**

    <ESC>IU(State)<CR><LF>

- **Parameter(s)**

    State: 0   Switch off icon "Unread SMS"
              1   Switch on icon "Unread SMS"
              2   Switch on icon "Unread SMS" in flashing-mode

- **Example(s)**

    <ESC>IU2<CR><LF>   Switch on icon "Unread SMS" in flashing-mode

3.5.37 Set Volume

The internal potentiometer is used to set the attenuation factor of the handset. This only effects the volume of the internal speaker. The value for the volume will be saved permanently in an EEPROM.

The potentiometer has 8 logarithmic steps.

- **Syntax**

  <ESC>IV(Value)<CR><LF>

- **Parameter(s)**

  Value: Value for the desired attenuation according to the table for potentiometer settings:

Value	Attenuation [dB]
0	35
1	30
2	25
3	20
4	15
5	10
6	5
7	0

Value is limited to 7. The default value is 3.

Table 7: Volume Table

- **Example(s)**

  <ESC>IV4<CR><LF>   Increase Volume of speaker in Handset

3.5.38 Get Volume

This command is used to retrieve the current volume value.

- **Syntax**

  <ESC>IV?<CR><LF>

- **Parameter(s)**
  None

- **Returned Value(s)**

  The current volume value in the format:

  <ESC>IV: (Value)<CR><LF>

  Attention: The returned value of the volume is ASCII-coded.

- **Example(s)**

  <ESC>IV?<CR><LF>   Request current volume setting

  Returned response could be (current volume setting is 4):

  <ESC>IV: 4<CR><LF>

3.5.39 Increase Volume

With this command it is possible to increase the volume by 1 step. When the max value (7) has already been reached, a "? <CR><LF>" is returned. The new value will also be stored in EEPROM.

- **Syntax**

  <ESC>IV+<CR><LF>

- **Parameter(s)**
  None

- **Example(s)**

  <ESC>IV+<CR><LF>   Increase volume

3.5.40 Decrease Volume

With this command it is possible to decrease the volume by 1 step. When the min value (0) has already been reached, a "? <CR><LF>" is returned. The new value will also be stored in EEPROM.

- **Syntax**

  <ESC>IV-<CR><LF>

- **Parameter(s)**
  None

- **Example(s)**

  <ESC>IV-<CR><LF>   Decrease volume

3.5.41 Set Icon "Rocker"

The icons "Rocker" are located in Rocker-Panel of the Soft-Key-Bar. The icon "Rocker" is intended to represent the Rocker-Key. There are 3 icons available.

Rocker up   Rocker up and down   Rocker down
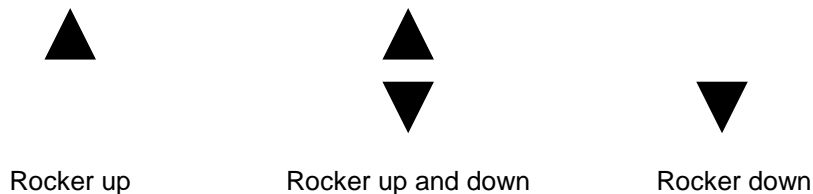
Figure 18: Rocker icons

- **Syntax**

  <ESC>IW(State)<CR><LF>

- **Parameter(s)**

  State: 0   Switch off icon "Rocker"
             1   Switch on icon "Rocker down"
             2   Switch on icon "Rocker up"
             3   Switch on icon "Rocker up and down"

- **Example(s)**

  <ESC>IW3<CR><LF>   Switch on icon "Rocker up and down"

3.5.42 Set Baudrate

This command changes the baudrate of the serial interface on the fly. That means, that all communication is stopped, the serial interface is closed and opened again with the new baudrate. The selected baudrate will be saved permanently in EEPROM. At delivery, the baudrate is 115200 Baud.

- **Syntax**

  <ESC>IX(Baud)<CR><LF>

- **Parameter(s)**

  Baud: Value for the desired Baudrate according to the table:

Baud	Baudrate
9600	9600 Baud
14400	14400 Baud
19200	19200 Baud
28800	28800 Baud
38400	38400 Baud
57600	57600 Baud
76800	76800 Baud
115200	115200 Baud

Table 8: Baudrate

- **Example(s)**

  <ESC>IX9600<CR><LF>   Change baudrate to 9600

3.5.43 Get Baudrate

This command is used to retrieve the current baudrate.

- **Syntax**

  <ESC>IX?<CR><LF>

- **Parameter(s)**
  None

- **Returned Value(s)**
  The current baudrate value in the format:

  ***<ESC>IX: (Baud)<CR>*<LF>**

  Attention: The returned value of the baudrate is ASCII-coded.

- **Example(s)**

  <ESC>IX?<CR><LF>   Request current baudrate

  Returned response could be (current baudrate is 9600):

  <ESC>IX: 9600<CR><LF>

3.5.44 Text in Invers-Style

A text-line can be displayed invers. Only one line can be shown in invers-style at the time. The textline will look like this:

Hello

Figure 19: Inverted text

- **Syntax**

  <ESC>IZ(Line)<CR><LF>

- **Parameter(s)**

  Line: 0   Switch off invers-style

            1 – 5 Show corresponding line in invers-style (1 corresponds to row 0). In display-modes

    o DISP_MODE_04_16
    o DISP_MODE_04_16_SMS

  there are 4 lines available, in display-modes

    o DISP_MODE_05_16
    o DISP_MODE_04_16_SMS

  there are 5 lines available.

- **Example(s)**

  <ESC>&D<CR><LF>   Clear Display/Set cursor to home pos.
  Hello<CR><LF>   Writes "Hello" to the display

  <ESC>IZ1<CR><LF>   Displays line 1 in invers style

3.5.45 Write Text

Text can be written always to the display in non-graphical modes. It is the only command which does not start with the <ESC>-character. The text starts from the current cursor-position. Text has to be terminated with <CR><LF>. But a <CR><LF> does not cause the text to be written to the next line. To write text to the next line, the cursor has to be set to the start position of that line first!

- **Syntax**

  Plain text<CR><LF>

- **Parameter(s)**
  None

- **Example(s)**

    <ESC>IDM2<CR><LF> Select display-mode DISP_MODE_05_16
    <ESC>[3;0H<CR><LF> Set Cursor to 3;0 (Line 3, Column 0)

    Hello<CR><LF>   Writes the text "Hello" to line 3

3.5.46 DISP_MODE_BIG_SIZE

This mode is a fixed combination of a "normal" display mode and a graphic mode. It uses the rows (lines) 1 to 3. Row 2 and 3 are used for displaying up to 8 big-size-characters (double-sized), Row 1 has normal size. With the 9th character the first character is automatically shifted to line 1 and so forth. So altogether there are 16 (row 1, normal size) plus 8 (row 2 and 3, big-size) characters possible. When there are more than 24 characters to be displayed, then line 1 is not shifted anymore and just one more character is added when it is written to the display.

The Icon-Bar and the Soft-Key-Bar are still available. They can be used - but they don't have to be used.

	Row 0 – Icon-Bar
03	Row 1 – normal size
32835160	Row 2 and Row 3 – big size
	Row 4 – not available
Left ↕ Right	Row 5 – Soft-Key-Bar
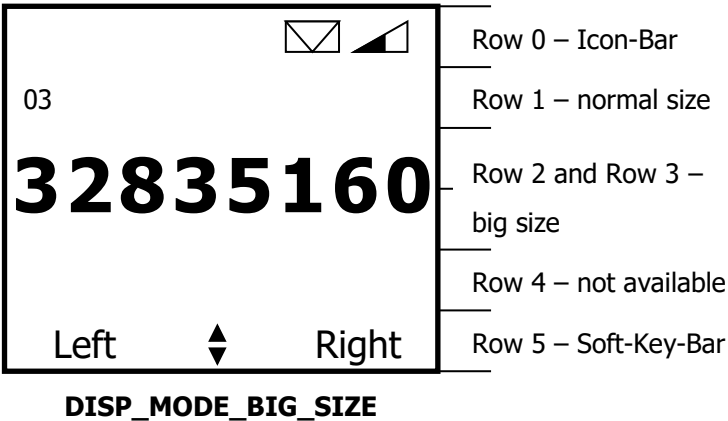
DISP_MODE_BIG_SIZE

Figure 20: Display mode big size

The figure above shows an example of the display-mode DISP_MODE_BIG_SIZE. Here the telephone-number 0332835160 is shown in the display.

Example:

The following steps have to be performed to get this screen (it is shown very detailed so that you can get an idea of how the Terminal works):

- Switch to a display mode allowing the Soft-Key-Bar and the Icon-Bar to be displayed, for example DISP_MODE_04_16

  <ESC>IDM0<CR><LF>

- Set text for SoftKeyL

  <ESC>IK1Left<CR><LF>

- Set text for SoftKeyR

  <ESC>IK2Right<CR><LF>

- Set icon "Rocker up and down"

  <ESC>IW3<CR><LF>

- Set Icon "Read SMS"

  <ESC>IU1<CR><LF>

- Set Icon "Volume" with value 4

  <ESC>IL4<CR><LF>

The result will be the basic display without the telephone-number. Now we use the display-mode DISP_MODE_BIG_SIZE

- Clear display (just to be save)

  <ESC>&D<CR><LF>

- Switch to display-mode DISP_MODE_BIG_SIZE

  <ESC>IDM4<CR><LF>

- Send telephone-number to the display

  0332835160<CR><LF>

Notice: The steps can sometimes differ, because you have to take in account the point of start. This example shows the steps after power-up.

3.6 Graphic Modes

In this section it will be discussed how the graphic-modes work. As mentioned earlier, there are several graphic modes implemented which allow the host-system to use the display **or** parts of it for graphical purposes.

In general, a graphical mode can be imagined as a transparency which is put on top of the current display. The following figure shows the principle of the graphic-modes.

In the figure on the next page a combination is used. The "main"-display uses the display-mode DISP_MODE_04_16 and with the graphic-mode DISP_MODE_GR_4ROWS the graphic shaded grey (here the PEITEL-logo) is put on top of the "main"-display, so that the resulting display will look like shown on the right-hand side.
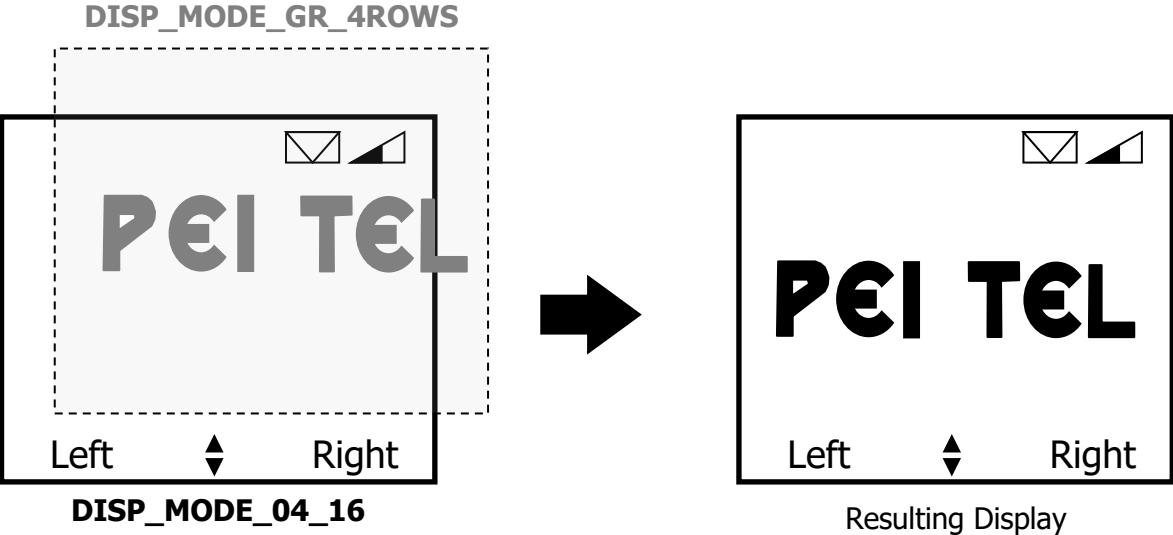
Figure 21: Graphic modes imagined as transparencies

3.6.1 Display Structure Graphic Mode

Now we have a more detailed look at the display structure. A row consists of 96x8 pixels. Every single pixel can be addressed by the user. Depending on the graphic mode up to 7 rows can be displayed (please see also 3.4.1 Display Organization). For technical reasons the 7th row has just 96x5 pixels. Thus, 96x53 pixels can be displayed altogether (see figure).
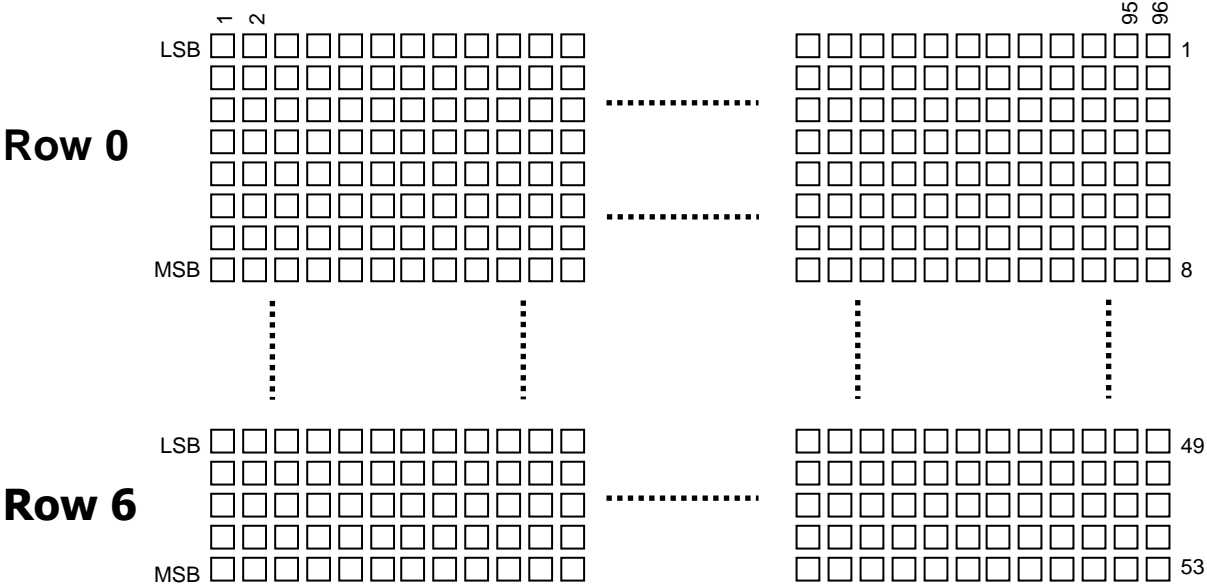
Figure 22: Display structure in graphic mode

3.6.2 Coding Scheme Graphic Mode

Having a look at one row will reveal the general coding scheme for the graphical modes. For each column the LSB (Least Significant Bit) is the top-dot, the MSB (Most Significant Bit) is the lowest dot as shown. If a dot of a column should be displayed, the corresponding bit has to be set to 1:
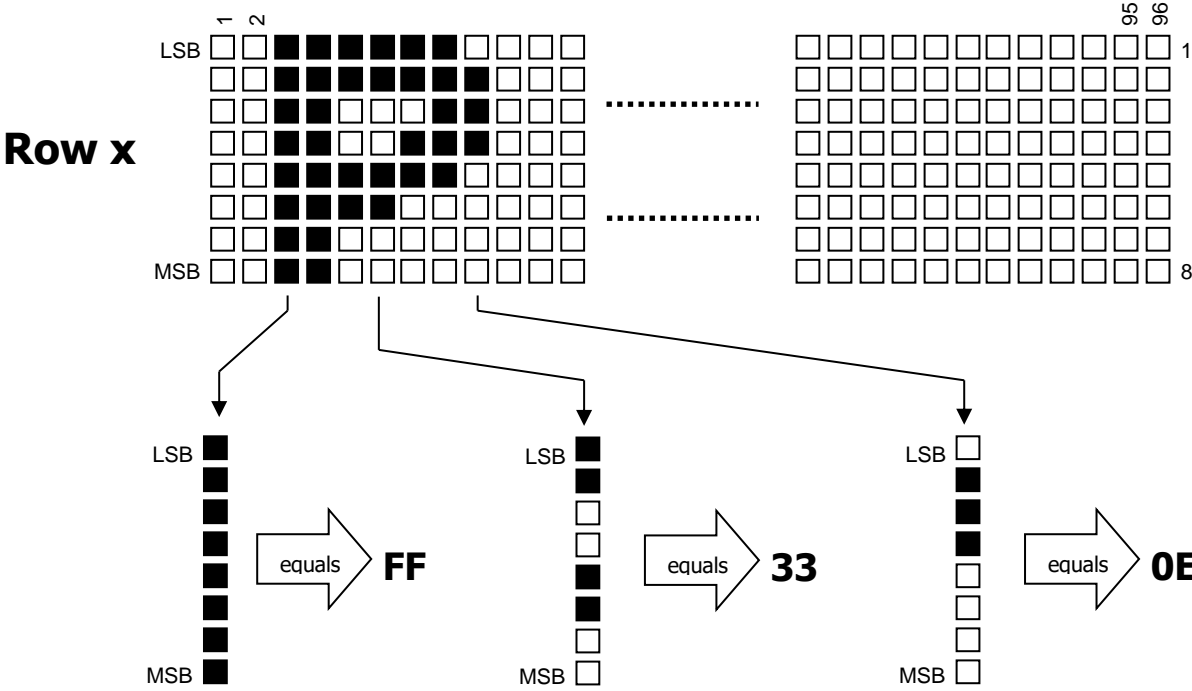
Figure 23: Coding scheme graphic mode

Therefore, the coding for the first 10 columns would be:

0000**FF**FF33**33**1B1F**0E**00

Following points have to be considered:

- A column is coded with a hexadecimal value. Because of the ASCII-oriented protocol the hexadecimal value consists actually of two bytes. For example, FF is 0x46 (F) followed by another 0x46 (F)!

- The values are case sensitive! For example, FF does not equal ff!

- For an entire display of graphics, a lot of bytes would have to be transferred. An entire row would consist of at least 2 x 96 Bytes of data. To speed things up, a compression algorithm can be used as shown in the next section.

3.6.3 Compression Algorithm

To fill an entire display with a graphic would require to transfer a lot of bytes. Even at a baudrate of 115200 Baud this would take quite a while. Therefore, a compression algorithm has been defined.

With the compression algorithm consecutive, equal values can be written as *n times the value*. To differentiate between "normal" data and "compressed" data, small letters are used. The small letter "a" means 2 times the value followed, the letter "b" means 3 times the value followed and so forth. The "highest" compression small letter is "y", which means 26 times the value followed. That is also the reason, why the data values are case sensitive.

Compression table:

2	3	4	5	6	7	8	9	10	11	12	13	14
a	b	c	d	e	f	g	h	i	j	k	l	m

15	16	17	18	19	20	21	22	23	24	25	26
n	o	p	q	r	s	t	u	v	w	x	y

Table 9: Compression table

Example:

The data stream with 52 bytes

00 00 00 00 FF FF FF FF FF 33 33 33 33 33 33 33 33 1B 2C 00 00 00 00 00 00 00

can be compressed to

c00 dFF g33 1B 2C f00

which has only 16 bytes.

00 00 00 00	FF FF FF FF FF	33 33 33 33 33 33 33 33	1B	2C	00 00 00 00 00 00 00
4 times "00"	5 times "FF"	8 times "33"	No compress.		7 times "00"
c00	dFF	g33	1B	2C	f00
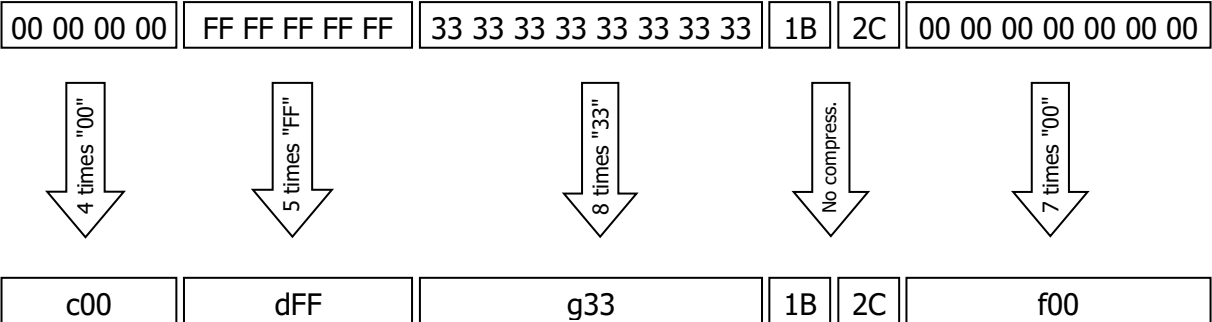
The spaces between the values are just for visual reasons. The real data stream does not have these spaces!

The real data stream would be:

c	0	0	d	F	F	g	3	3	1	B	2	C	f	0	0
0x63	0x30	0x30	0x64	0x46	0x46	0x67	0x33	0x33	0x31	0x42	0x32	0x43	0x66	0x30	0x30

3.6.4 Graphic Display Modes

A short description of the 8 graphical modes is provided in this section.

ROW 0
ROW 1
ROW 2
ROW 3
ROW 4
ROW 5
ROW 6
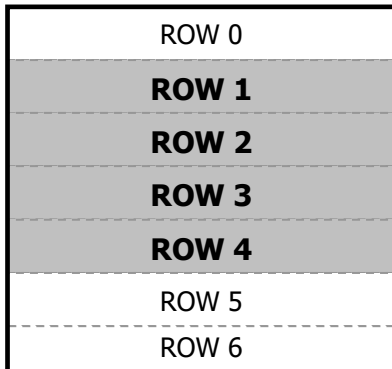
DISP_MODE_GR_4ROWS:

Row 1 to Row 4 can be used for graphics. Icon-Bar (Row 0) and Soft-Key-Bar (Row 5) remain visible.

Command: <ESC>IDM5<CR><LF>

ROW 0
ROW 1
ROW 2
ROW 3
ROW 4
ROW 5
ROW 6
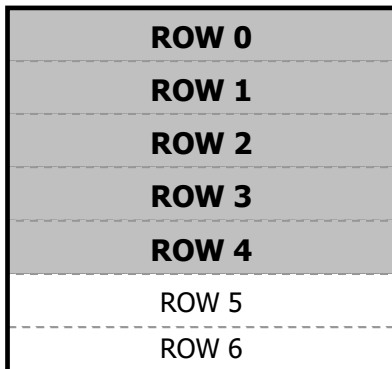
DISP_MODE_GR_5ROWS:

Row 0 to Row 4 can be used for graphics. Soft-Key-Bar (Row 5) is still visible. Icon-Bar is not visible in this mode.

Command: <ESC>IDM6<CR><LF>

ROW 0
ROW 1
ROW 2
ROW 3
ROW 4
ROW 5
ROW 6
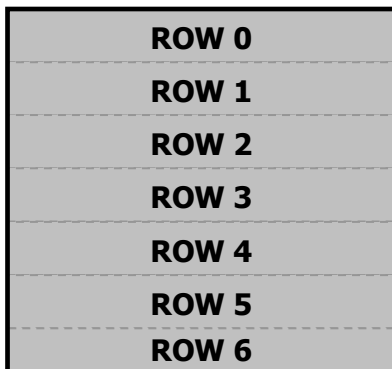
DISP_MODE_GR_6ROWS:

Entire display can be used for graphics. Neither Soft-Key-Bar nor Icon-Bar are visible in this mode.

Command: <ESC>IDM7<CR><LF>

ROW 0
ROW 1
ROW 2
ROW 3
ROW 4
ROW 5
ROW 6
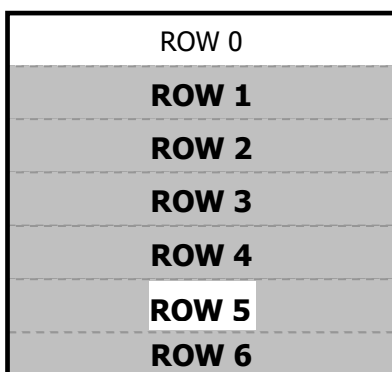
DISP_MODE_GR_7:

Row 1 to Row 6 can be used for graphics. Icon-Bar (Row 0) is visible. The Rocker-Panel of the Soft-Key-Bar (Row 5) is visible.

Command: <ESC>IDM8<CR><LF>
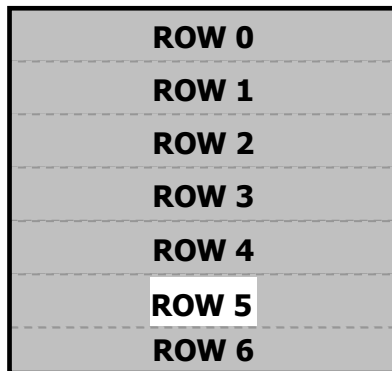
DISP_MODE_GR_8:

Row 0 to Row 6 can be used for graphics. Icon-Bar (Row 0) is not visible. The Rocker-Panel of the Soft-Key-Bar (Row 5) is visible.

Command: <ESC>IDM9<CR><LF>
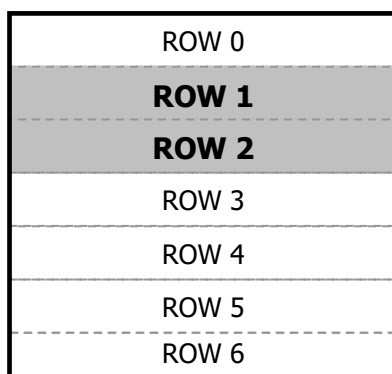
DISP_MODE_GR_9:

Row 1 and Row 2 can be used for graphics.

Command: <ESC>IDM10<CR><LF>
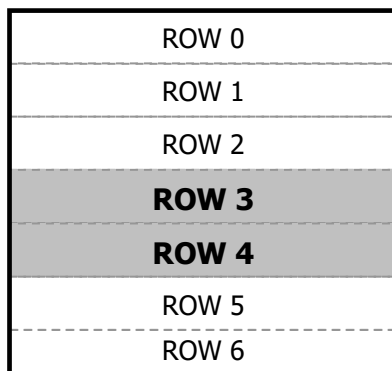
DISP_MODE_GR_10:

Row 3 and Row 4 can be used for graphics.

Command: <ESC>IDM11<CR><LF>
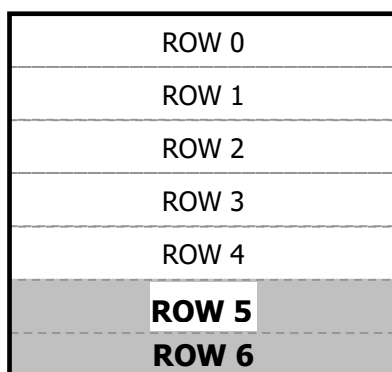
DISP_MODE_GR_11:

Row 5 and Row 6 can be used for graphics. The Rocker-Panel of the Soft-Key-Bar (Row 5) is visible.

Command: <ESC>IDM12<CR><LF>

Figure 24: Graphic display modes

Attention:

When data is written to a row, this data is buffered. The next time that row is displayed, the buffered data will be displayed. This can cause unwanted effects, when display-modes with many rows are selected.

Now that we have knowledge about the principles of the graphic mode, the command for using the graphical modes can be introduced.

3.6.5 Send Graphic Data

With this command the uncompressed or compressed data can be sent to the Handset. **Please remember to select the corresponding graphical mode afterwards to update the display!** The command "Clear Display" does not work for graphic modes.

- **Syntax**

  <ESC>G(Row),(Data)z<CR><LF>

- **Parameter(s)**

Row:	r0-r6	The row where the data shall be displayed
Data:	xxx	The uncompressed or compressed data ASCII-coded.
	00	To clear an entire row.

- **Example(s)**

  Example 1:

  <ESC>Gr0,00z<CR><LF>   Clear row 0

  <ESC>IDM7<CR><LF>   Select display-mode to update display

  Example 2:

  <ESC>Gr0,a00aFFa331B1F0Ez<CR><LF>

  <ESC>IDM9<CR><LF>   Select display-mode to update display

  Example 2 displays the previously defined graphic (special character "P", section 2.6.2) at row 0 of the display.

Hints:

It is recommended to transfer all rows of the graphic first, then update the display. In this way, the whole graphic will be visible at once. Otherwise, there is the possibility of seeing the graphic "growing"!

Select only the graphic mode(s) you need. For example, if you would like to put a graphic in row 1 and row 2, then select DISP_MODE_GR_9 and not one of the other.

4 Appendix

4.1 Character-Set SMS

Character-Set SMS: 0-127

Dez	Hex	Char
0	00	**1)**
1	01	**£**
2	02	**$**
3	03	**¥**
4	04	**È**
5	05	**É**
6	06	**Ù**
7	07	**Ì**
8	08	**Ò**
9	09	**Ç**
10	0A	**LF**
11	0B	**Ø**
12	0C	**ø**
13	0D	**CR**
14	0E	**Å**
15	0F	**å**
16	10	**Δ**
17	11	**_**
18	12	**.**
19	13	**.**
20	14	**.**
21	15	**.**
22	16	**.**
23	17	**.**
24	18	**.**
25	19	**.**
26	1A	**.**
27	1B	**2)**
28	1C	**Æ**
29	1D	**æ**
30	1E	**ß**
31	1F	**É**

Dez	Hex	Char
32	20	**SP**
33	21	**!**
34	22	**"**
35	23	**#**
36	24	**¤**
37	25	**%**
38	26	**&**
39	27	**`**
40	28	**(**
41	29	**)**
42	2A	*****
43	2B	**+**
44	2C	**,**
45	2D	**-**
46	2E	**.**
47	2F	**/**
48	30	**0**
49	31	**1**
50	32	**2**
51	33	**3**
52	34	**4**
53	35	**5**
54	36	**6**
55	37	**7**
56	38	**8**
57	39	**9**
58	3A	**:**
59	3B	**;**
60	3C	**<**
61	3D	**=**
62	3E	**>**
63	3F	**?**

Dez	Hex	Char
64	40	**¡**
65	41	**A**
66	42	**B**
67	43	**C**
68	44	**D**
69	45	**E**
70	46	**F**
71	47	**G**
72	48	**H**
73	49	**I**
74	4A	**J**
75	4B	**K**
76	4C	**L**
77	4D	**M**
78	4E	**N**
79	4F	**O**
80	50	**P**
81	51	**Q**
82	52	**R**
83	53	**S**
84	54	**T**
85	55	**U**
86	56	**V**
87	57	**W**
88	58	**X**
89	59	**Y**
90	5A	**Z**
91	5B	**Ä**
92	5C	**Ö**
93	5D	**Ñ**
94	5E	**Ü**
95	5F	**§**

Dez	Hex	Char
96	60	**¿**
97	61	**a**
98	62	**b**
99	63	**c**
100	64	**d**
101	65	**e**
102	66	**f**
103	67	**g**
104	68	**h**
105	69	**i**
106	6A	**j**
107	6B	**k**
108	6C	**l**
109	6D	**m**
110	6E	**n**
111	6F	**o**
112	70	**p**
113	71	**q**
114	72	**r**
115	73	**s**
116	74	**t**
117	75	**u**
118	76	**v**
119	77	**w**
120	78	**x**
121	79	**y**
122	7A	**z**
123	7B	**ä**
124	7C	**ö**
125	7D	**ñ**
126	7E	**ü**
127	7F	**à**

1) The character '@' was moved to 128; this allows the use of strings terminated by zero.
2) This code value is reserved for the extension to another extension table. On receipt of this code, a receiving entity shall display a space until another extension table is defined.

Character-Set SMS: 128-255

Dez	Hex	Char	Dez	Hex	Char	Dez	Hex	Char	Dez	Hex	Char
128	80	@	160	A0	**1)**	192	C0	**1)**	224	E0	**1)**
129	81	**1)**	161	A1	**1)**	193	C1	**1)**	225	E1	**1)**
130	82	**1)**	162	A2	**1)**	194	C2	**1)**	226	E2	**1)**
131	83	**1)**	163	A3	**1)**	195	C3	**1)**	227	E3	**1)**
132	84	**1)**	164	A4	**1)**	196	C4	**1)**	228	E4	**1)**
133	85	**1)**	165	A5	**1)**	197	C5	**Ä**	229	E5	**ä**
134	86	**1)**	166	A6	**1)**	198	C6	**1)**	230	E6	**1)**
135	87	**1)**	167	A7	**1)**	199	C7	**1)**	231	E7	**1)**
136	88	**1)**	168	A8	**1)**	200	C8	**1)**	232	E8	**1)**
137	89	**1)**	169	A9	**1)**	201	C9	**1)**	233	E9	**1)**
138	8A	**1)**	170	AA	**1)**	202	CA	**1)**	234	EA	**1)**
139	8B	**1)**	171	AB	**1)**	203	CB	**1)**	235	EB	**1)**
140	8C	**1)**	172	AC	**1)**	204	CC	**1)**	236	EC	**1)**
141	8D	**1)**	173	AD	**1)**	205	CD	**1)**	237	ED	**1)**
142	8E	**1)**	174	AE	**1)**	206	CE	**1)**	238	EE	**1)**
143	8F	**Ä**	175	AF	**1)**	207	CF	**1)**	239	EF	**1)**
144	90	**1)**	176	B0	**1)**	208	D0	**1)**	240	F0	**1)**
145	91	**1)**	177	B1	**1)**	209	D1	**1)**	241	F1	**1)**
146	92	**1)**	178	B2	**1)**	210	D2	**1)**	242	F2	**1)**
147	93	**1)**	179	B3	**1)**	211	D3	**1)**	243	F3	**1)**
148	94	**1)**	180	B4	**1)**	212	D4	**1)**	244	F4	**1)**
149	95	**1)**	181	B5	**1)**	213	D5	**1)**	245	F5	**1)**
150	96	**1)**	182	B6	**1)**	214	D6	**1)**	246	F6	**1)**
151	97	**1)**	183	B7	**1)**	215	D7	**Ö**	247	F7	**ö**
152	98	**1)**	184	B8	**1)**	216	D8	**1)**	248	F8	**1)**
153	99	**1)**	185	B9	**1)**	217	D9	**1)**	249	F9	**1)**
154	9A	**Ö**	186	BA	**1)**	218	DA	**1)**	250	FA	**1)**
155	9B	**Ü**	187	BB	**1)**	219	DB	**1)**	251	FB	**1)**
156	9C	**1)**	188	BC	**1)**	220	DC	**1)**	252	FC	**1)**
157	9D	**1)**	189	BD	**1)**	221	DD	**Ü**	253	FD	**ü**
158	9E	**1)**	190	BE	**1)**	222	DE	**1)**	254	FE	**1)**
159	9F	**1)**	191	BF	**1)**	223	DF	**1)**	255	FF	**1)**

1) This value will be displayed as a "╲".

4.2 Character-Set TB

Character-Set TB: 0-127

Dez	Hex	Char
0	00	**1)**
1	01	**1)**
2	02	**1)**
3	03	**1)**
4	04	**1)**
5	05	**1)**
6	06	**1)**
7	07	**1)**
8	08	**1)**
9	09	**1)**
10	0A	**LF**
11	0B	**1)**
12	0C	**1)**
13	0D	**CR**
14	0E	**1)**
15	0F	**1)**
16	10	**1)**
17	11	**1)**
18	12	**1)**
19	13	**1)**
20	14	**1)**
21	15	**1)**
22	16	**1)**
23	17	**1)**
24	18	**1)**
25	19	**1)**
26	1A	**1)**
27	1B	**1)**
28	1C	**1)**
29	1D	**1)**
30	1E	**1)**
31	1F	**1)**

Dez	Hex	Char
32	20	**SP**
33	21	**!**
34	22	**"**
35	23	**#**
36	24	¤
37	25	**%**
38	26	**&**
39	27	`
40	28	**(**
41	29	**)**
42	2A	*****
43	2B	**+**
44	2C	**,**
45	2D	**-**
46	2E	**.**
47	2F	**/**
48	30	**0**
49	31	**1**
50	32	**2**
51	33	**3**
52	34	**4**
53	35	**5**
54	36	**6**
55	37	**7**
56	38	**8**
57	39	**9**
58	3A	**:**
59	3B	**;**
60	3C	**<**
61	3D	**=**
62	3E	**>**
63	3F	**?**

Dez	Hex	Char
64	40	@
65	41	**A**
66	42	**B**
67	43	**C**
68	44	**D**
69	45	**E**
70	46	**F**
71	47	**G**
72	48	**H**
73	49	**I**
74	4A	**J**
75	4B	**K**
76	4C	**L**
77	4D	**M**
78	4E	**N**
79	4F	**O**
80	50	**P**
81	51	**Q**
82	52	**R**
83	53	**S**
84	54	**T**
85	55	**U**
86	56	**V**
87	57	**W**
88	58	**X**
89	59	**Y**
90	5A	**Z**
91	5B	**[**
92	5C	****
93	5D	**]**
94	5E	‾
95	5F	──

Dez	Hex	Char
96	60	‖
97	61	**a**
98	62	**b**
99	63	**c**
100	64	**d**
101	65	**e**
102	66	**f**
103	67	**g**
104	68	**h**
105	69	**i**
106	6A	**j**
107	6B	**k**
108	6C	**l**
109	6D	**m**
110	6E	**n**
111	6F	**o**
112	70	**p**
113	71	**q**
114	72	**r**
115	73	**s**
116	74	**t**
117	75	**u**
118	76	**v**
119	77	**w**
120	78	**x**
121	79	**y**
122	7A	**z**
123	7B	**{**
124	7C	**|**
125	7D	**}**
126	7E	‾
127	7F	**1)**

1) This not printable character will be displayed as a SPACE.

2) This value will be displayed as " ＼".

Character-Set TB: 128-255

Dez	Hex	Char
128	80	**2)**
129	81	**2)**
130	82	**2)**
131	83	**2)**
132	84	**2)**
133	85	**2)**
134	86	**2)**
135	87	**2)**
136	88	**2)**
137	89	**2)**
138	8A	**2)**
139	8B	**2)**
140	8C	**2)**
141	8D	**2)**
142	8E	**Ä**
143	8F	**2)**
144	90	**2)**
145	91	**2)**
146	92	**`**
147	93	**2)**
148	94	**2)**
149	95	**2)**
150	96	**2)**
151	97	**2)**
152	98	**2)**
153	99	**Ö**
154	9A	**Ü**
155	9B	**2)**
156	9C	**2)**
157	9D	**2)**
158	9E	**2)**
159	9F	**2)**

Dez	Hex	Char
160	A0	**2)**
161	A1	**2)**
162	A2	**2)**
163	A3	**£**
164	A4	**2)**
165	A5	**2)**
166	A6	**2)**
167	A7	**2)**
168	A8	**¿**
169	A9	**2)**
170	AA	**2)**
171	AB	**2)**
172	AC	**2)**
173	AD	**2)**
174	AE	**2)**
175	AF	**2)**
176	B0	**2)**
177	B1	**2)**
178	B2	**2)**
179	B3	**2)**
180	B4	**2)**
181	B5	**2)**
182	B6	**2)**
183	B7	**2)**
184	B8	**2)**
185	B9	**2)**
186	BA	**2)**
187	BB	**2)**
188	BC	**2)**
189	BD	**2)**
190	BE	**2)**
191	BF	**¿**

Dez	Hex	Char
192	C0	**À**
193	C1	**Á**
194	C2	**Â**
195	C3	**Ã**
196	C4	**Ä**
197	C5	**Å**
198	C6	**Æ**
199	C7	**Ç**
200	C8	**È**
201	C9	**É**
202	CA	**Ê**
203	CB	**Ë**
204	CC	**Ì**
205	CD	**Í**
206	CE	**Î**
207	CF	**Ï**
208	D0	**2)**
209	D1	**Ñ**
210	D2	**Ò**
211	D3	**Ó**
212	D4	**Ô**
213	D5	**Õ**
214	D6	**Ö**
215	D7	**2)**
216	D8	**Ø**
217	D9	**Ù**
218	DA	**Ú**
219	DB	**Û**
220	DC	**Ü**
221	DD	**2)**
222	DE	**2)**
223	DF	**2)**

Dez	Hex	Char
224	E0	**2)**
225	E1	**2)**
226	E2	**2)**
227	E3	**2)**
228	E4	**ä**
229	E5	**2)**
230	E6	**2)**
231	E7	**2)**
232	E8	**2)**
233	E9	**2)**
234	EA	**2)**
235	EB	**2)**
236	EC	**2)**
237	ED	**2)**
238	EE	**2)**
239	EF	**2)**
240	F0	**2)**
241	F1	**2)**
242	F2	**2)**
243	F3	**2)**
244	F4	**2)**
245	F5	**2)**
246	F6	**ö**
247	F7	**2)**
248	F8	**2)**
249	F9	**2)**
250	FA	**2)**
251	FB	**2)**
252	FC	**ü**
253	FD	**2)**
254	FE	**2)**
255	FF	**2)**

2) This value will be displayed as " ↘".

4.3 Abbreviations

SKE   Start-Key-Event
SKEM   Start-Key-Event-Message
LKE   Long-Key-Event
LKEM   Long-Key-Event-Message
RKE   Repeated-Key-Event
RKEM   Repeated-Key-Message
EKE   End-Key-Event
EKEM   End-Key-Event-Message
FKE   False-Key-Event
FKEM   False-Key-Event-Message

PTT   Push-To-Talk
PWM   Pulse Width Modulation

<ESC>   ESC-Character ($0x1B_h$) – one single byte
<CR>   Carriage Return-Character ($0x0D_h$) – one single byte
<LF>   Line Feed-Character ($0x0A_h$) – one single byte

END OF DOCUMENT