Contents

1 WHAT IS SMS-BASE? ................................................................................................................. 10
2 INSTALLATION OF SMS-BASE ............................................................................................. 14
3 USING THE SMS-BASE MENUS ............................................................................................ 21
4 USING A PHONE MODEM .......................................................................................................... 70
5 CHANGING THE PASSWORD ..................................................................................................... 74
6 BACKING UP DATA ................................................................................................................... 76
7 DIAGNOSING ERRORS ............................................................................................................... 77
A DIRECTORIES AND FILES IN SMS-BASE ........................................................................ 80
B SMS-BASE SPECIFICATION .................................................................................................. 83
C SETTING AND TESTING THE PHONE MODEM ..................................................................... 85
D USING SMS-BASE VIA A SRIOSACO UNIT ........................................................................ 94
E SPA BUS AND DATA COMMUNICATION PROTOCOL ................................................... 97
F USING SMS-BASE WITH WINDOWS 3.1 ............................................................................. 99
G USING THE TOOL ERASEALL .............................................................................................. 100

© 2001 ABB Substation Automation Oy, All rights reserved
## CONTENTS

<table>
<thead>
<tr>
<th>I</th>
<th>WHAT IS SMS-BASE ? ...........................................................................................................</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>THE MAIN FUNCTIONS PERFORMED BY THE PRODUCT ..................................................................</td>
<td>10</td>
</tr>
<tr>
<td>1.2</td>
<td>SYSTEMS WITH SMS-BASE .................................................................................................</td>
<td>11</td>
</tr>
<tr>
<td>1.3</td>
<td>SMS-BASE DATA FLOW .......................................................................................................</td>
<td>12</td>
</tr>
<tr>
<td>1.4</td>
<td>CONTROLLING PROGRAM OPERATION ....................................................................................</td>
<td>13</td>
</tr>
<tr>
<td>II</td>
<td>INSTALLATION OF SMS-BASE ............................................................................................</td>
<td>14</td>
</tr>
<tr>
<td>2.1</td>
<td>THE PARTS OF THE PRODUCT ..............................................................................................</td>
<td>14</td>
</tr>
<tr>
<td>2.2</td>
<td>INSTALLING THE PROGRAM ..............................................................................................</td>
<td>14</td>
</tr>
<tr>
<td>2.2.1</td>
<td>How to get started after installation ..........................................................................</td>
<td>18</td>
</tr>
<tr>
<td>2.3</td>
<td>CONNECTING THE MODEM .................................................................................................</td>
<td>19</td>
</tr>
<tr>
<td>2.4</td>
<td>CODE PAGE SETTING ........................................................................................................</td>
<td>19</td>
</tr>
<tr>
<td>III</td>
<td>USING THE SMS-BASE MENUS ............................................................................................</td>
<td>21</td>
</tr>
<tr>
<td>3.1</td>
<td>SMS-BASE SELECT MENU ..................................................................................................</td>
<td>21</td>
</tr>
<tr>
<td>3.2</td>
<td>SMS-BASE UTILITIES MENU .............................................................................................</td>
<td>21</td>
</tr>
<tr>
<td>3.2.1</td>
<td>Alter additional configurations ....................................................................................</td>
<td>21</td>
</tr>
<tr>
<td>3.2.1.1</td>
<td>Changing transformer settings and relay inputs .........................................................</td>
<td>22</td>
</tr>
<tr>
<td>3.2.1.2</td>
<td>SPACOM slave address ....................................................................................................</td>
<td>22</td>
</tr>
<tr>
<td>3.2.2</td>
<td>Scan and list modules ....................................................................................................</td>
<td>23</td>
</tr>
<tr>
<td>3.2.3</td>
<td>Alter application structure ..........................................................................................</td>
<td>23</td>
</tr>
<tr>
<td>3.2.3.1</td>
<td>Adding system elements to the application structure ....................................................</td>
<td>23</td>
</tr>
<tr>
<td>3.2.3.2</td>
<td>Working with the menu ..................................................................................................</td>
<td>24</td>
</tr>
<tr>
<td>3.2.3.3</td>
<td>Designations ..................................................................................................................</td>
<td>24</td>
</tr>
<tr>
<td>3.2.3.4</td>
<td>Editing alternatives available for selection ..................................................................</td>
<td>25</td>
</tr>
<tr>
<td>3.2.3.5</td>
<td>Editing text ....................................................................................................................</td>
<td>25</td>
</tr>
<tr>
<td>3.2.3.6</td>
<td>Communication settings ..............................................................................................</td>
<td>26</td>
</tr>
<tr>
<td>3.2.3.7</td>
<td>Adding new elements to the application structure .......................................................</td>
<td>26</td>
</tr>
<tr>
<td>3.2.3.7.1</td>
<td>How to manually compose/add/delete relays and modules ........................................</td>
<td>29</td>
</tr>
<tr>
<td>3.2.3.8</td>
<td>Entering modem and telephone numbers .....................................................................</td>
<td>30</td>
</tr>
<tr>
<td>3.2.3.8.1</td>
<td>Using the communication parameters phone-number list ............................................</td>
<td>30</td>
</tr>
<tr>
<td>3.2.3.8.2</td>
<td>Phone-number as a part of the application structure ....................................................</td>
<td>31</td>
</tr>
<tr>
<td>3.2.3.8.3</td>
<td>Giving the phone-number within the modem window ...................................................</td>
<td>32</td>
</tr>
<tr>
<td>3.2.3.9</td>
<td>Entering a non-SPA relay ...........................................................................................</td>
<td>33</td>
</tr>
<tr>
<td>3.2.3.10</td>
<td>Entering a relay without settings ...............................................................................</td>
<td>34</td>
</tr>
<tr>
<td>3.2.3.11</td>
<td>Supplementary system elements for historical data or sets of settings ........................</td>
<td>35</td>
</tr>
<tr>
<td>3.2.3.12</td>
<td>Deleting a selection ....................................................................................................</td>
<td>36</td>
</tr>
<tr>
<td>3.2.3.13</td>
<td>Creating a new application structure ............................................................................</td>
<td>37</td>
</tr>
<tr>
<td>3.2.3.13.1</td>
<td>Using ‘SETUP’/’Appl. Struct. Path’ ..............................................................................</td>
<td>38</td>
</tr>
</tbody>
</table>
3.2.3.14  AS Handling ............................................................................................................ 42
3.2.3.14.1  Changing to another application structure .......................................................... 43
3.2.3.14.2  Creating application structures ............................................................................. 44
3.2.3.14.3  Copying application structures .............................................................................. 45
3.2.3.14.4  Deleting application structures ............................................................................. 46
3.2.3.14.5  Editing application structure comments ................................................................. 46
3.2.3.14.6  AS Handling help function ..................................................................................... 47
3.2.3.14.7  Scanning for application structures ......................................................................... 47
3.2.3.14.8  Packing application structures ............................................................................... 48
3.2.3.14.9  Unpacking application structures ......................................................................... 49
3.2.3.15  Copying units within the application structure ........................................................... 50
3.2.3.16  Show transformer settings and relay inputs ............................................................... 52
3.2.3.17  Working with data on floppy disks ........................................................................... 52
3.2.4  Checking the application structure ................................................................................ 53
3.2.4.1  Using the ‘Check application structure’ menu ............................................................. 53
3.2.4.2  What does ‘Check application structure’ do? ............................................................... 54
3.2.4.3  How does the ‘Check application structure’ check multiple slave addresses?............ 55
3.2.5  SPA- and Terminal emulators ....................................................................................... 57
3.2.5.1  SPA-Terminal emulator ............................................................................................... 57
3.2.5.2  Terminal emulator ...................................................................................................... 58
3.2.5.3  Examples for usage of SPA- & Terminal emulators ..................................................... 60
3.2.5.3.1  SPA-Terminal emulator examples ............................................................................. 60
3.2.5.3.2  Terminal emulator examples .................................................................................... 62
3.2.6  SMS-BASE colour settings .......................................................................................... 66
3.2.7  View readme ............................................................................................................... 66
3.2.7.1  Introduction ............................................................................................................... 66
3.2.7.2  Connecting the modem ............................................................................................. 66
3.3  SMS-BASE ABOUT MENU ............................................................................................ 68
3.4  SMS-BASE SETUP MENU ............................................................................................ 68
3.4.1  Display adapter ............................................................................................................ 68
3.4.2  Printer port .................................................................................................................. 69
3.4.3  Printer form lines ......................................................................................................... 69
3.4.4  Appl. struct. path .......................................................................................................... 69
3.4.5  Save setup ................................................................................................................... 69
4  USING A PHONE MODEM ..................................................................................................... 70
4.1  CALLING .......................................................................................................................... 70
4.2  HANGING UP .................................................................................................................. 72
4.3  WARNING OF WRONG CONNECTION OR AN UNFINISHED CALL ......................... 73
5  CHANGING THE PASSWORD ............................................................................................. 74
6  BACKING UP DATA .............................................................................................................. 76
7  DIAGNOSING ERRORS .......................................................................................................... 77
7.1  ERROR MESSAGES ......................................................................................................... 77
I The SMS 010 concept

SMS 010 software as a whole consists of parametrisation, reporting and disturbance evaluation software. The separate software products are combined according to the needed functionality. Furthermore is communication equipment needed according to the system built-up. REPORT is also requiring a SACO 100 M or a SRIO 500/1000 M data communicator.

The complete SMS 010 software package provides setting, reporting and disturbance evaluation facilities. The separate software products are combined according to the needs.

Fig. IA SMS 010 software structure

SMS-BASE is the basic software that is always needed to be able to run a SMS 010 application. It is a universal program for relays, disturbance recorders and annunciators. Other SMS 010 software modules are added to this program according to the requirements of the application. Depending on the types of protection in use, special software module/s or programs are needed, e.g. SM/SPA_, SM/REL 100, REPORT, DR-COM, REVAL, etc.

SMS 010 enables the user to call a relay via a leased line, the public telephone system or, if he is in the station, via a direct optical fibre link to read relay data, events, disturbances and change settings. SMS 010 can also be made to communicate with a PC by connecting a cable to its serial port. For example, when using a SPAC 500/300 relay, it is possible to link SPAC to the PC by connecting the SPAC control module’s RS-232 front panel interface to the PC’s RS-232 port using the cable supplied with the SM/SPA_ package. The data read from the SPAC can then be examined in SMS 010.

The current power system load values, recorded values and relay settings can be viewed and the settings can also be changed.

The communication between the PC and the relay is based on the SPA protocol.

With the aid of the PC, it is also possible to create a file with relay settings which only takes a few minutes to download to the protection.

The SMS 010 Station Monitoring System is designed with the relay engineer in mind to provide all the information and tools he needs to perform the tasks more efficiently.
II SMS-BASE v. 2.1

The SMS-BASE version 2.1 software replaces earlier versions. The changes from v. 2.0 to v. 2.1 are few.

Handling of the year 2000 is corrected where is was needed, concerns printing and some presentation of file dates.

When printing and measuring transducers are involved, a question box that asks if the user wants to also print the measuring transducers is added.

The start-up picture was also slightly upgraded.

Repeating of important remarks still worthwhile mentioning from version 2.0:

The transformer settings were increased by In2 and I02. This means that SM/SPA_ v. 2.0 and higher contain some relays, e.g. SPAA 341 with the SPCJ 4D28 module that utilises these added transformers. If the application structure, (AS), is made by SMS-BASE versions 1.0 or 1.1 aren’t these transformers included, but the transformers are automatically added, when the module is used, (e.g. by ‘Edit parameters and monitor data’ or ‘editing the transformer settings’).

The maximum dataparts or modules in a ‘Unit’ were increased to 150. This means that the AS directory structure has been extended. Default is now four characters for each directory in the application structure, e.g. ..\O001\S005\O003\.. . Due to this an update into versions 2.0 or higher of the programs REPORT and DR-COM is required when using SMS-BASE v. 2.0 or higher. Application structures created by SMS-BASE versions 1.0 or 1.1 can be used as before, also with REPORT and DR-COM v. 2.0.

The printing of parameters is now when selecting ‘Print parameters’ for a module enabling the user to decide whether all parameters, (measured values, state values, setting parameters), or only setting parameters should be printed.

III Summary of the SMS-BASE User’s Guide

This User’s Guide instructs the user in the normal use of the software product SMS-BASE. SMS-BASE provides the possibility of viewing the data and setting the parameters of units of the PYRAMID series using a personal computer (PC).

From a remote location, the PC communicates with the unit either via a leased line or the public telephone network, or within the same substation by a direct connection.

A typical system using SMS-BASE is presented at the beginning of the User’s Guide. The software installation procedure is also described, which includes an example of software configured for a specific application.

Examples of how to view relay data and set relay parameters are presented in the relevant 'SM/__' user guides. Exercises enable a quick start of entering a real protection configuration and taking SMS-BASE to practical use.

Finally, the use of telephone modems is explained. The technical specification and operating requirements of SMS-BASE, communication links, a detailed list of the software files and the differences when used in conjunction with a SRIO unit are dealt with in the appendices.
IV Application

This User’s Guide is applicable to SMS-BASE versions 2.0 and 2.1.

This User’s Guide describes the normal application of the software product SMS-BASE.

Reference should be made to the specific documentation for details of communication hardware and its connection. Those mentioned in this guide are only by way of example.

V Conditions for use

The information contained in this document is subject to change without notice and should not be construed as a commitment by ABB Transmit Oy, Relays & Network Control division, which also may introduce improvements and/or changes in the product(s) and/or programs described in this document at any time. ABB Transmit Oy, Relays & Network Control division assumes no responsibility for any errors that may appear in this document or in the described product(s), except as may be expressly agreed in specific documents signed by ABB Transmit Oy, Relays & Network Control division in connection with the supply of the product(s).

In no event should ABB Transmit Oy, Relays & Network Control division be liable for special, incidental or consequential damages arising from the use of this document or of the software and hardware products described in this document.

VI Copyright

This document and the enclosed software must not be reproduced or copied in part or as a whole in any form or by any means without the prior written permission of ABB Transmit Oy, Relays & Network Control division except in the following cases and for the following purposes:

- The floppy disks may be copied once for back-up purposes only.

The contents of this document and enclosed products must not be imparted to a third party nor be used for any unauthorised purpose. Contravention will be prosecuted.

Copyright © ABB Transmit Oy, Relays & Network Control division 1997

VII Trademarks

The following trademarks are used in this guide:

<table>
<thead>
<tr>
<th>TRADEMARK</th>
<th>REGISTERED BY</th>
</tr>
</thead>
<tbody>
<tr>
<td>SMS-BASE</td>
<td>ABB Transmit Oy, Relays &amp; Network Control Division, Finland</td>
</tr>
<tr>
<td>REPORT</td>
<td>ABB Transmit Oy, Relays &amp; Network Control Division, Finland</td>
</tr>
<tr>
<td>DR-COM</td>
<td>ABB Transmit Oy, Relays &amp; Network Control Division, Finland</td>
</tr>
<tr>
<td>SM/SPA</td>
<td>ABB Transmit Oy, Relays &amp; Network Control Division, Finland</td>
</tr>
<tr>
<td>SM/SACO</td>
<td>ABB Transmit Oy, Relays &amp; Network Control Division, Finland</td>
</tr>
<tr>
<td>SM/SPCR</td>
<td>ABB Transmit Oy, Relays &amp; Network Control Division, Finland</td>
</tr>
<tr>
<td>IBM</td>
<td>International Business Machines Corporation</td>
</tr>
<tr>
<td>EGA</td>
<td>International Business Machines Corporation</td>
</tr>
<tr>
<td>VGA</td>
<td>International Business Machines Corporation</td>
</tr>
</tbody>
</table>

Additional products referred to in this guide are registered trademarks of their respective manufacturers.
VIII     Conventions in this guide

The following conventions are used in this guide:

<table>
<thead>
<tr>
<th>A key on the keyboard</th>
<th>&lt;A&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>Special keys</td>
<td>&lt;up-arrow&gt;, &lt;down-arrow&gt;, &lt;left-arrow&gt;, &lt;right-arrow&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;Home&gt;, &lt;End&gt;, &lt;PgUp&gt;, &lt;PgDn&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;Ins&gt;, &lt;Del&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;NumLock&gt;, &lt;CapsLock&gt;</td>
</tr>
<tr>
<td>Keys pressed at the same time</td>
<td>&lt;Alt&gt;, &lt;Ctrl&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;PrintScrn&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;Alt&gt;+&lt;F2&gt;</td>
</tr>
</tbody>
</table>

Texts quoted from the screen are written in 'Quotation' marks. If the position of the text on the screen is important, the whole or part of the screen is shown. Information from the screen is shown in a frame where it is not in quotation marks.

A part of the screen (a window) is represented as:

```
RAW
SPA
SRIO
```

The selected option in a window is in bold print.

A quotation of the above information would read:

'SPA'

The response by the user to a request is set in double quotation marks, e.g.
"Station 1" or "009" or "xcopy C:\SMS\<Appl. Struct. Name> /s/e a;".

An interactive dialogue between the user (U) and the computer (C) is sometimes listed in the same way as a script of a play. The following is an example of an interactive dialogue:

```
C - C:
U - C:\xcopy *,* e:
    C -
    U -
```
### IX Designations and definitions

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Relay</td>
<td>PYRAMID Relays</td>
</tr>
<tr>
<td>SMS 010</td>
<td>A system for setting of parameters, measuring, monitoring, event reporting and disturbance evaluation</td>
</tr>
<tr>
<td>SMS-BASE</td>
<td>The basic software needed to run SMS 010 applications</td>
</tr>
<tr>
<td>Application structure</td>
<td>The configuration of stations, protected units, relays and parts of relays and data storage for the relay parts for a particular application</td>
</tr>
<tr>
<td>SPA</td>
<td>Data bus and protocol for communication between a 'Master' and a number of 'Slaves'</td>
</tr>
<tr>
<td>Master</td>
<td>A unit requesting information</td>
</tr>
<tr>
<td>Slave</td>
<td>A unit responding to the requests from the 'Master'</td>
</tr>
<tr>
<td>Modem</td>
<td>Two-way converter at a signal interface</td>
</tr>
<tr>
<td>Phone modem</td>
<td>Modem between RS-232 and telexine corresponding to the CCITT specification</td>
</tr>
<tr>
<td>Fibre modem</td>
<td>Modem for RS-232 between an electrical and a optical fibre communication link</td>
</tr>
<tr>
<td>Dataset</td>
<td>A block of data transferred by a relay to a PC</td>
</tr>
<tr>
<td>Filename</td>
<td>8 characters or less followed by an extension</td>
</tr>
<tr>
<td>Extension</td>
<td>3 characters or less defining the type of file</td>
</tr>
<tr>
<td>REC</td>
<td>'Ext' for acquired 'received values'</td>
</tr>
<tr>
<td>INS</td>
<td>'Ext' for acquired 'installed values'</td>
</tr>
<tr>
<td>VAL</td>
<td>'Ext' for observed 'validated values'</td>
</tr>
<tr>
<td>Window</td>
<td>A part of the screen</td>
</tr>
<tr>
<td>Default value</td>
<td>Value used when no other value specified</td>
</tr>
<tr>
<td>Password</td>
<td>Personal code entered by a user to permit him to modify relay settings</td>
</tr>
<tr>
<td>Chapter</td>
<td>Principal part of this guide</td>
</tr>
<tr>
<td>Section</td>
<td>Part of a chapter</td>
</tr>
</tbody>
</table>
1 WHAT IS SMS-BASE?

SMS-BASE is the platform needed for almost all SMS 010 programs. SMS-BASE provides the user with the tools needed for creating and handling of application structures, setting communications, terminal emulator functions... Briefly said: SMS-BASE is the corner-stone of SMS 010.

SMS-BASE is essential when treating the modules, sending and receiving data to/from the modules, editing parameter settings etc.

Working in SMS-BASE is based on an application structure that is built-up according to the actual power system. ‘UTILITIES’/ ‘Alter application structure’ is the main tool for configuring the application structure, while ‘SELECT’ is the alternative providing with the tools and other SMS 010 programs installed for using the application structures, (communicating with the modules, setting new values, event handling, disturbance recording and evaluation etc.)

1.1 The main functions performed by the product

SMS-BASE is used mainly for configuring an SMS 010 system by providing the user with tools for determining the application structure (AS) and arranging the system's stations.

Another important function performed by SMS-BASE is to transfer the information stored in PYRAMID Relays to the screen of a PC and enable it to be processed from the keyboard via a communication channel. The PC also provides facility for storing data and producing reports.

SMS-BASE's data input and output functions (Fig. 1.1.A) introduce the concept of storing data on a relay-by-relay basis according to the structure of the power system. Data can also be monitored, edited and processed for reports. The relay input data consist of parameter settings and the output data of the values of current settings, power system load values and recorded signal status.

<table>
<thead>
<tr>
<th>INPUT DATA</th>
<th>OUTPUT DATA</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Name of station, bay etc.</td>
<td>- Read</td>
</tr>
<tr>
<td>- Communication port settings</td>
<td>- Stored signal status</td>
</tr>
<tr>
<td>- Telephone number</td>
<td>- Load values</td>
</tr>
<tr>
<td>- Relay settings</td>
<td>- Settings</td>
</tr>
</tbody>
</table>

Fig. 1.1.A SMS-BASE data input and output functions
1.2 Systems with SMS-BASE

The SMS-BASE software can run on a PC which is connected either directly to a PYRAMID relay (Fig. 1.2.A) or via telephone modems to a group of PYRAMID relays (Fig. 1.2.B). The actual configuration of the system can vary, but in all cases there is an SPA loop. The co-ordination with a station control system connected to the same loop is presented in the appendices.

Fig. 1.2.A An example of an external MMI system using SMS-BASE

Fig. 1.2.B An example of a SMS 010 system using SMS-BASE
1.3 SMS-BASE data flow

In a SMS-BASE system, data is stored in the relay and in the personal computer. The SMS-BASE commands facilitate the transfer of data between the relay and the personal computer. SMS-BASE includes functions for the selection of the correct data memory in the computer for each relay as well as functions to minimise the risk of mistaken selection.

In SMS-BASE, an 'application structure' is defined as the representation of the relays in the real power system structure.

There are two different types of data in most PYRAMID relays, 1) monitored variables such as load values and recorded data and 2) setting parameters.

SMS-BASE stores monitored variables in a file of the type 'VALUES.INS'. The contents of the data file can be viewed by activating the function 'Monitor measured and recorded data'.

During the setting procedure, parameter settings can be viewed in the part of the screen labelled 'Present values'. Initially the setting files 'VALUE.INS' and 'VALUES.VAL' contain the same values. The settings in 'VALUES.VAL', however, can be changed. The contents of the files will then be different until 'VALUES.VAL' has been transferred to the relay buffer and returned to overwrite the contents of 'VALUES.INS'. While the setting procedure is in progress, windows showing the data of both files are displayed side by side for comparison. If the user is satisfied with the new settings, the 'Edit buffer' contains a set of new settings validated by the user.

The new settings in the file 'VALUES.VAL' are downloaded to become the effective relay settings upon entering the corresponding command at the PC.

<table>
<thead>
<tr>
<th>SPACOM relays</th>
<th>COMMUNICATION</th>
<th>SMS-BASE</th>
</tr>
</thead>
<tbody>
<tr>
<td>DATA STORE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Monitored variables</td>
<td>--&gt;-----------&gt;--</td>
<td>FILES</td>
</tr>
<tr>
<td>Parameter settings</td>
<td>--&gt;-----------&gt;--</td>
<td>VALUES.INS</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>FILES</td>
</tr>
<tr>
<td></td>
<td></td>
<td>VALUES.INS</td>
</tr>
<tr>
<td></td>
<td></td>
<td>VALUES.VAL</td>
</tr>
<tr>
<td></td>
<td></td>
<td>VALUES.VAL</td>
</tr>
</tbody>
</table>

Fig. 1.3.A SMS-BASE data memories and data flow

SMS-BASE refuses to accept any inadmissible settings the user attempts to enter.
1.4 Controlling program operation

The user controls the operation of the SMS-BASE program by selecting the desired items from the main menu and the sub-menus (Fig. 1.4.A). The application structure is edited and the communication parameters are set by the option 'UTILITIES/Alter Application structure'. Communication problems are traced with the aid of 'UTILITIES/Terminal emulator' or SPA-Terminal emulator.

![Diagram of SMS-BASE v. 2.0 menu structure](attachment:image.png)

**Fig. 1.4.A** SMS-BASE program control in the SELECT' and 'UTILITIES’ menus.

**PART** | **FUNCTION**
--- | ---
Main menu | For selecting one of the main system operations
SELECT | Select relay and view and change settings
- Edit parameters and monitor data | Editor used to view and change parameter settings
- Send and receive parameters | Downloads parameters to the relay
- Receive parameters | Requests present parameter settings from the relay
- Resetting of output relays and registers | Latched relays and registers can be resetted
- Print parameters | Prints the contents of VALUES.VAL, -.INS or -.REC
- Modem handling | Calls the station modem
- Password handling | Check password and station selection

UTILITIES | For changing transducer data and the relay address
- Alter additional configuration | Scan the SPA nodes and list the relays connected
- Scan and list modules | Add, delete and edit the application structure
- Alter application structure | Emulates a terminal on a PC
- Terminal emulator | For checking and drawing the application structure
- Check application structure | For defining the SMS-BASE screen colours
- Colour setting | View connecting cables data
- View readme |
2 INSTALLATION OF SMS-BASE

Before SMS-BASE can be used for the first time, the software has to be installed on the hard disk of the PC. This guide assumes that the user is familiar with PC’s.

You will always have to start a new installation with the SMS-BASE 1 disk. After a successful installation of the SMS-BASE disk 1 the other SMS 010 software disks can be installed in any order. Notice that except from SMS-BASE, and perhaps other programs, you will also need to install at least some relay package, SM/... Otherwise you will not have the relay/module software required for making your application structure. Please, consult the corresponding manual for installation guidance, if needed.

The standard installation for SMS-BASE is presented in detail and some alternatives are briefly explained.

2.1 The parts of the product

The SMS-BASE product is supplied in a folder containing this manual and a number of floppy disks. The contents of the disks included in the SMS-BASE package are presented in detail in Appendix A. Apart from the SMS-BASE program on disk 1, the SMS-BASE disk 2 includes a example application structure for exercises.

2.2 Installing the program

Installation of the SMS-BASE software system involves copying files from the floppy disks to the hard disk 'C:'. The installation procedure presented below assumes that the files are being loaded from the 'A:' drive, but any other 3½” drive may be used.

The program files can also be installed to a directory on another drive than C:

IMPORTANT to know before installation of SMS 010 products!

If the harddisk of the PC is defect the installation program might perform an uncompleted installation. The DOS command CHKDSK, (CHKDSK/F can repair minor faults), can be used to clarify if the harddisk is defect. SCANDISK is also possible to use, if available.
In Brief:

The normal sequence when installing the program for the first time is:

1. Insert floppy disk 1 in drive A:
2. Change the command prompt to A:>
3. Determine the main memory and hard disk space using 'RESOURCE'.
4. Compare the result with the requirement for SMS-BASE given in Appendix B.
5. Run the installation program 'INSTALL' on floppy disk 1.
6. Press 'ENTER' so that SMS-BASE installs the path 'C:\SMS\BASE'.
7. If SMS-BASE is already installed in this path, you can choose to update or reinstall it.
8. Select either A:> or C:\\DATA-EX as the path for your application structure data or define your own.
9. Select the type of your graphics adapter.
10. The files are now installed on your hard disk.
11. Install the example application structure, if you do not already have one on your hard disk.
12. Change the command prompt back to C:.

If SMS-BASE already exists on your hard disk, the installation program checks its version and providing it is older, starts to update it to the new version. Should the existing version be the same as the version on the disks, the message 'SMS-BASE program is up-to-date!' is displayed.
The detailed presentation of the installation sequence which follows assumes that the command prompt is set to show the current directory, i.e. "PROMPT=$PSG" is set in your AUTOEXEC.BAT file. The steps of the brief list of operations above are used as headings in the following detailed explanation.

1. Insert floppy disk 1 in drive A:

2. Change the command prompt to A:>
   
   C - C:\>
   U - C:\a:  followed by <Enter>
   C - A:\>

3. Determine the main memory and hard disk space using 'RESOURCE'.
   
   C - A:\>
   U - A:\pkunzip must resource.exe  followed by <Enter>
   U - A:\resource  followed by <Enter>

The following screen appears:

```
ABB RELAYS SMS RESOURCE
RESOURCES OF THIS PC
- MS-DOS version 5.00
- Basic primary work memory available: 538 kbytes
- Available space at hard disk C: 7.3 Mbyte

Info: The figure for the available basic primary work memory does not include the size of this program
```

Fig. 2.2.A  'Resource' shows how much memory is available.

4. Compare the result with the requirement for SMS-BASE given in Appendix B. Compare the available space to that required according to Appendix B. There is no sense in continuing with the installation unless there is sufficient main and hard disk memory space!

5. Run the installation program 'INSTALL' for floppy disk 1.
   
   C - A:\>
   U - A:\install  followed by <Enter>

The following screens are then displayed:

6. First you select the SMS-BASE home directory: (Maximum installation path length is altogether 21 characters).

   ![Path for your installation?](C:\SMS\)

7. If SMS-BASE is already installed in this path, the following question appears:

   ![Do you want to update the program at](C:\SMS\BASE\)

   Yes  No

8. Now choose the path for the example application structure:

   ![Select path for your setting data structure?](C:\SMS\DATA-EX\)

   Custom path
9. It is now necessary to select the type of graphic adapter in use. (Select ‘Autodetect’ if you don’t know).
This setting can also be changed from the SMS-BASE program:

```
Select type of your graphics adapter ?=
Hercules
Toshiba T310(e)
A菲 Olivetti
Nokia 3TT
Modified Hercules (Wyse B&W)
EGA
VGA
Autodetect
```

10. The files are now installed on your hard disk:

```
Installing/Updating program...
```

Installation of SMS-BASE on your hard disk is now complete:

```
New program installed !
Press any key to continue
```

11. The second disk includes an example application structure which is a help when using SMS-BASE for the
first time. Insert SMS-BASE disk 2 and install the application structure. It can be done in a continuous
installation or as a separate installation started by the INSTALL command at the floppy disk station.
Select the path for your application structure.

There are no other limits than the free harddisk space for the amount of installed application structures,
which means that it is possible to install the enclosed application structure examples for other programs,
as well. Select ‘Custom path’ and give the new application name. Notice that it is always the last installed
application that will be shown, when starting the SMS-BASE program. Changing between different
application structures is done within the SMS-BASE program, ‘SETUP’/Appl. struct. path’ where a new
or existing appl. struct. path can be given. Confirm the change by selecting ‘SETUP’/Save setup’.
Another possibility is ‘UTILITIES’/Alter application structure’/S = AS handling’.

12. Change the command prompt back to C:
C - A:\>
U - A:\e: followed by <Enter>
C - C:\>
U - Remove the last floppy disk from drive A:

How to start the program is shown at the very beginning of Chapter 3.

Providing you used the default paths and depending on what relays are installed, the hard disk will have the
following directory tree after installation has been completed:

```
C:\SMS\BASE\            Home directory of the system
  SUPPORT\              SMS-BASE auxiliary programs
  MODULES\              Descriptions
    OTHER\              Descriptions of non-SPA modules and relays
      MODEM\           Description of modem
    C:\SMS\DATA-EX\      Application structure data
```
2.2.1 How to get started after installation

This section is a brief start-up for SMS-BASE user's. This section contains references to information about the most important steps, like creating a new application structure, setting of comm. parameters etc.

If you have installed another application structure, (from REPORT or DR-COM disks), you will notice that the examples in this manual differs a bit from your application. This is of minor importance, the principle for working in the application structure is the same for all application structures.

- Start of the SMS-BASE program
  Start of the SMS-BASE program is done from the C:\SMS\BASE directory, (if default installation), by typing SMSBASE.

- Creating of application structure, adding of relays and slave addresses
  Creating of a new application structure is done by the 'SETUP'/Appl.struct.path' commands described in section 3.2.3.13. Save the new name with 'SETUP'/Save setup’. Recommendation: Use C:\SMS or similar as home directory for your applications. Then you will always know where to find your applications. Before it is possible to utilise the new application, you will have to add items to the five levels of the application structure. Hence next step will be to choose 'UTILITIES'/Alter application structure’, see section 3.2.3.7.

  When you add relays at the 'Unit' level you should also select each module in the relay and give it the correct slave address. Avoid using identical slave addresses for different modules, which might cause communication difficulties.

- Changing between different applications
  Section 3.4.4. 'SETUP'/Appl. struct. path’ enables changing between different applications. Save the change with 'SETUP'/Save setup’.

- Create/change communication parameters
  Section 3.2.3.6 'UTILITIES'/Alter application structure’ explains how to create the essential communication parameters with the <C> button. Recommendation: Create the communication parameters at the STATION level. i.e. at the 'Obj'/Bay' selection window. The comm. param. will then be valid for all units within the concerned station. Save the parameters with <Enter>, before that you will NOT have any working communication, (comm.parameters don’t exist by default).

- When to use SPA or SRIO protocol ?
  The SPA protocol is mostly used for direct communication to the relays. The SRIIO protocol is to be selected, when using SRIIO 500 M, SRIIO 1000 M or SACO 100 M. These data communication units use the SACO 100 M protocol, (however called SRIIO protocol in SMS-BASE), for communication to the PC comm.port. The REPORT program needs these bus-masters, both SPA or SRIIO protocol may be used for DR-COM. The standard comm. param. settings are for each protocol listed in the appendix D.2. The SPCR 8C19/8C27 disturbance recorder modules can when directly connected via cable to the PC use a SPA protocol, (with 7 or 8 data-bits).

  The settings, (unit lists etc.), for SRIIO/SACO data communication units are shortly described in the REPORT manual. The SRIIO/SACO data communicators must be configured for the relay modules to poll for alarms & events.
• **SPA- and Terminal emulator**

The Terminal emulators can be used for testing of communication and tracing of communication difficulties, but also for reading and setting of module parameters. (Not stored in the SMS-BASE application).

Sections 3.2.5.1 and 3.2.5.3.1 describes the usage of the SPA-Terminal, ( used for SPA type relays), while sections 3.2.5.2 and 3.2.5.3.2 describe the Terminal emulator, which is mostly used for (“on-line”) communication to the SRIO/SACO data communication units.

• **Setting of measuring transformers and relay inputs**

The relay inputs, current and voltage measuring transformers can be set from the 'UTILITIES'/ Alter additional configuration'/ Transducer data’, see section 3.2.1.1. The settings are valid for a whole relay, not only a single module.

• **Color setting**

Section 3.2.6 ‘UTILITIES/Color setting’ enables the SMS-BASE operator to customise the used colors in SMS-BASE. Save the new settings and exit the program. The new colors settings are taken in to use by next start-up.

• **Exit the program**

At any level, function or program, except from when the program is questioning the operator for an answer, (select an alternative), the <Escape> can be used for exiting. At the SMS-BASE main screen level it will ask if the operator intends to exit the program. <Enter> or <Y> will finish the running of SMS-BASE program and return to the DOS command line.

2.3 **Connecting the modem**

Refer to Appendix C for the connection of the telephone modem to the PC.

Appendix C also gives information on connecting an optical fibre modem to a PC and to a telephone modem. Refer to the corresponding documents for details of connecting relays to modems in the substation.

2.4 **Code page setting**

The code page defines the translation of ASCII codes into characters for presentation on the screen, respectively for printing by the printer. Refer to the instructions for your PC operating system for further information.

SMS-BASE uses semi-graphical and Greek characters that are available on code pages 437, 860, 863 and 865.

Code page 437 is used as the default setting.
The following code pages should not be used, otherwise the displays will be incorrect:

<table>
<thead>
<tr>
<th>CODE PAGE NUMBER</th>
<th>TYPE</th>
<th>COMMENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>850</td>
<td>International</td>
<td>No Greek characters</td>
</tr>
<tr>
<td>852</td>
<td>Slavic</td>
<td>No Greek characters, no frames</td>
</tr>
</tbody>
</table>

The procedure for checking and changing the code page number is as follows:

1. To check the currently active code page number:

   C - C:\>
   U - C:\>chcp followed by <Enter>
   C - Active code page: 437
   C - C:\>

   Always check the active code page number before setting a new one.

   The command "mode" + <Enter> returns details of the code page settings for all the PC devices.

2. To change active code page number to 437 enter:

   - "chcp 437".
3 USING THE SMS-BASE MENUS

This chapter describes the procedure for all the SMS-BASE menu selecting operations. The contents of the menu described in Chapter 3.1 are defined by the user using the tool 'Alter application structure' which is selected from the 'UTILITIES' menu.

```
C - C:\>
U - C:\>cd SMS\BASE followed by <Enter> changes directory to ..\BASE\nC - C:\SMS\BASE>
U - C:\SMS\BASE\SMSBASE followed by <Enter> starts SMS-BASE
```

Fig. 3.A Starting the SMS-BASE program.

3.1 SMS-BASE SELECT MENU

The 'SELECT' menu is the channel to set the parameters for the modules, read data from the modules, send new settings to the modules, run additional programs like REPORT and DR-COM etc. In all these cases is the application structure very essential, because the communication is based on the communication settings and the configuration of it. You can move in the application structure by pressing the arrow buttons, <Home>, <End>, <PgUp> and <PgDn> keys. An item is selected by pressing <Enter> and when the <Esc> key is pressed moving back to previous level in the structure. By the way, only one application structure can be in use at the same time.

3.2 SMS-BASE UTILITIES MENU

The 'UTILITIES' item in the main menu includes many useful SMS-BASE tools for modifying and configuring of application structures. Some other functions are found as well. Selecting ‘UTILITIES’ opens the following sub-menu:

```
Alter additional configurations
Scan and list modules
Alter application structure
Check application structure
Terminal emulator
SPA-terminal emulator
Colour settings
View readme
```

Fig. 3.2.A These menu items are the found under the ‘UTILITIES’ menu

3.2.1 Alter additional configurations

When using SPACOM relays, the module data can be viewed in either 'per unit' notation or as 'primary values'. Where the data has been selected to be displayed in primary values, it is important to set the primary, the secondary and the module input values according to c.t.'s, v.t.'s and relay inputs in use. The corresponding settings are made by selecting this item from the 'UTILITIES' menu.
3.2.1.1 Changing transformer settings and relay inputs

Since the transformers, current and voltage transformers, are used for calculating values in SMS-BASE it is also sometimes necessary to change these settings. The default values, Fig. 3.2.1.A, are always installed to the application structure when adding units, which means that the only task for the user is to edit these settings, when needed. The file containing these settings is named PACKAGE.PCK and every unit in the application structure has its own transformer settings.

Editing mode in Fig. 3.2.1.A is achieved by marking the value to change and pressing <Enter> and giving the whole value, (parts of the old value cannot be reused).

The transformer settings can be set different for each relay, but the setting is valid for all the modules found in this relay. SMS-BASE and SM/SPA_ versions earlier than 2.0 were not equipped with In2 and I02 settings. This means that an application structure created by a SMS-BASE version 1.0 or 1.1 should be updated when using e.g. SPAA 341 from SM/SPA_ v. 2.0.

Fortunately for the user is SMS-BASE 2.0 automatically updating the transformer settings. When e.g. editing a module, ('Edit parameters and monitor data'), is the PACKAGE.PCK on the ‘Unit’ level updated. (The settings for In2 and I02 are copied from In and I0, but only if they are missing). Updating is naturally also done by editing the transducer file, ‘Modify’/’Transducer data’, which was described earlier in this section.

3.2.1.2 SPACOM slave address

The 'SPACOM address' (SPA NodeNo) in each of the relay modules can be changed by selecting the corresponding module from the 'Modify/SPACOM address' menu. Upon selecting the desired relay module, a window is opened which displays the current address and makes provision for entering a new address for the module. Note that when giving a new address in 'SPACOM address' this new address field remains until a successful change in the module is done. The 'New address' is transferred to the module both at 'Send and receive parameters' and 'Receive parameters'. At the same time is the slave address of the module in the application structure updated.

Note ! If communicating via a SACO/SRIO data communicator will this operation result in loosing communication with the module when the new slave address isn’t included in the UNIT list. Accordingly, there will be an interrupt in the communication between SMS-BASE and the module in question until the UNIT list is fixed.
3.2.2 Scan and list modules

This utility scans all the possible SPA addresses, 1 - 999, within selected station and lists which relay modules SMS-BASE currently is able to communicate with. The communication parameters have to be set correct, otherwise can the modules not be detected. The advantage with this function is that modules without display, can be traced for address, if unknown. (E.g. SPOC 1XX has no display, but has default slave address according to serial number.) The list can be saved in a file or transferred directly to the printer. The function can be interrupted by <Esc>.

![Scanning slaves](image)

Fig. 3.2.2.A Example of a list when scanning the system for responding slaves, addresses in left table

3.2.3 Alter application structure

SMS-BASE has to be aware of which units/(relays) and modules/(data items) the structure includes and how to communicate with them. This information is entered via the menu item ‘Alter application structure’.

‘Alter application structure’ is mainly used during commissioning of the system to enter the basic description of the power system and later if the structure is to be modified and also for changing communication parameters.

‘Alter application structure’ is the main tool for configuring and setting up the application structure and provides the user with many application structure related functions. Section 3.2.3 informs thoroughly how to work with application structures and the functions offered.

3.2.3.1 Adding system elements to the application structure

The data of relays and other devices is stored in a five-level structure. The default names of these levels are ‘Organisation’, ‘Station’, ‘Object/Bay’, ‘Unit’ and ‘Module/Part’. The user can define a logic structure consisting of several ‘Organisations’ each with several ‘Stations’, several ‘Objects/Bays’ etc.

Each level in the application structure corresponds to a level in the directory structure of the PC.

Select ‘Alter application structure’ in the ‘UTILITIES’ menu to work with the application. If you have installed SMS-BASE for the first time without including the example application structure, you will have to create a new one.

The example application structure provided with the program is used as the starting point for the exercises in this chapter. It has five levels.
3.2.3.2 Working with the menu

Selecting 'Alter Application Structure' from the 'UTILITIES' menu starts the following screen (providing the SMS-BASE example application structure was installed):

![Screen Screenshot]

Fig. 3.2.3.2.A The first screen of 'Alter application structure'

The designation of the ‘Organisation’, Fig 3.2.3.2.A, is presented in a box to the upper left of the of the screen. The paths and directories where the SMS-BASE program and the application structure data are located are given at the top of the screen on the right. The active keys and their functions are displayed along the bottom of the screen.

The procedure for working with 'Alter application structure' is as follows:

**Selection**
- Select using the <arrow> keys. Press <Enter> to confirm your choice or <Esc> to undo.
  Confirmed selections are shown in the lower part of the screen.

**Change text**
- Select the text and press <E> to open an editing field.

**Row editor**
- Write or edit the text using the normal editing keys: <side arrow>, <Home>, <End>, <Ins>, <Del>.

**Change structure**
- Make selections as far down the structure as possible. Mark the alternative after the one to be added.
  Press <A> to add a system element.
  Press <D> to delete a system element.

**Change setting**
- Use <spacebar> to select from the list.
  Press <Enter> to confirm your choice.

If there are more than 16 alternatives on a level of the structure, the user can see the exceeding ones by scrolling.
3.2.3.3 Designations

Any text not exceeding 46 characters can be entered for each of the alternatives on the levels 'Organisation', 'Station', 'Object/Bay', 'Unit' and 'Module/Part'.

Note that the text must be written exactly as it appears on the lowest 'Module' level of the example structure. The unit will normally be added from the 'Quick add' window, in which case all the modules belonging to the unit are filled in automatically. All units installed during the installation procedure are displayed in the 'Quick add' window at the unit level. If a unit is added using the installation program at a later stage, it is automatically included in the list in the 'Quick add' window.

Modems and phone numbers are entered according to special rules presented elsewhere in this document, however, the syntax for telephone numbers should be decided before putting SMS-BASE into general use.

3.2.3.4 Editing alternatives available for selection

Once installed, the alternatives available for selection in the application structure are not usually changed. It might become necessary if new relays are included, existing relays are modified regarding the modules, the name of some part of the system is changed or the method of communication changes. The following sections aims at giving insight into these matters.

3.2.3.5 Editing text

Assuming the following change has to be made to texts on two levels:

<table>
<thead>
<tr>
<th>BEFORE</th>
<th>AFTER</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vikom</td>
<td>Vikom 1</td>
</tr>
<tr>
<td>H22 Grundvik</td>
<td>H22 Grundvik 1</td>
</tr>
<tr>
<td>H23 Sundom</td>
<td>H23 Sundom 1</td>
</tr>
</tbody>
</table>

The procedure is as follows:

- Select 'Vikom'.
- Press <E> for editing. The window below appears:

```
[Edit the menu item]

Vikom
```

- Move the cursor to the end of the text by pressing <End>.
  Write number ‘1’ after ‘Vikom’.
  ‘Vikom 1’
  Terminate the entry by pressing <Enter>.
The selected text changes.

- Mark 'Vikom 1' and continue down to the 'Bay' level by pressing <Enter>. Change the texts 'H22 Grundvik' and 'H23 Sundom' the same way.
- Now the screen shows:

![Screen shot](image)

Fig. 3.2.3.5.A The application screen after editing the texts

Note that if the modules are added manually, the syntax of the texts at the beginning of the level must be correct. Normally the units to be added are listed in the 'Quick add' window at the 'Unit' level and the corresponding modules are added automatically.

### 3.2.3.6 Communication settings

This section handles the setting of communications in an application structure. Normally it is enough with one set of communication parameters\(^1\) located on the station level. Modem phone-numbers\(^2\) can be used several ways, but this is described more in detail in section 3.2.3.8. Finally, the module slave address\(^3\) has to be set, while it is tremendously important for a successful communication.

The type of communication settings varies according to the level of the application structure:

<table>
<thead>
<tr>
<th>TYPE OF DATA</th>
<th>RELATED TO</th>
</tr>
</thead>
<tbody>
<tr>
<td>(^1) Serial port settings</td>
<td>One setting per station</td>
</tr>
<tr>
<td></td>
<td>entered at the 'Select Object/Bay' level</td>
</tr>
<tr>
<td>(^2) Phone number for a modem</td>
<td>One or many numbers per station</td>
</tr>
<tr>
<td></td>
<td>entered at the 'Select Object/Bay' level</td>
</tr>
<tr>
<td>(^3) SPA node number</td>
<td>One number per module</td>
</tr>
<tr>
<td></td>
<td>entered at the 'Select Module' level</td>
</tr>
</tbody>
</table>

1. Communication parameters for the serial port

Press <C> to change or enter the communication settings for the serial port. This alternative is available on several levels in the structure, but the recommendation is to have the communication parameters on the 'Station' level, ('Select Obj/Bay'). Besides, e.g. REPORT and DR-COM are using the settings in the communication parameter file located at the 'Station' level. <P> for printing is activated at the same places as <C> and prints the contents of the SPIN.CNF file. Printing can be done only for valid communication parameter files, (SPIN.CNF).

The procedure is as follows:

- Press <C> at 'Select Object/Bay' level
- The screen changes to:

```
<table>
<thead>
<tr>
<th>Select Organization</th>
<th>SMSBASE directory:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Asm No.</td>
<td>Select Station</td>
</tr>
<tr>
<td>Vaasa</td>
<td>Select Object/Bay</td>
</tr>
<tr>
<td>Vaster</td>
<td>Serie 100 SPA COM protections</td>
</tr>
<tr>
<td>Serie 100 SPA COM protections</td>
<td></td>
</tr>
<tr>
<td>Serie 300 SPA COM protections</td>
<td></td>
</tr>
<tr>
<td>Serie 500 SPA COM protections</td>
<td></td>
</tr>
<tr>
<td>Serie 800 SPA COM protections</td>
<td></td>
</tr>
<tr>
<td>Serie 900 SPA COM protections</td>
<td></td>
</tr>
<tr>
<td>Serie SACO Annunciator unit</td>
<td></td>
</tr>
<tr>
<td>MODRM phno: 8 012 987 - 654</td>
<td></td>
</tr>
</tbody>
</table>

Edit communication parameters
```

![Edit communication parameters](image)

**Fig 3.2.3.6.A** Setting the communication parameters for selected ‘Station’ Vaasa

The current settings are showed to the left, while the settings can be changed/edited to the right, (Fig. 3.2.3.6.A.). (Typical SRIO settings to the right, while typical SPA settings to the left). The new values are selected using the up and down <arrow> keys and pressing <Space bar>, while the ‘Serial port’ option ‘Other’ has to be edited by <E> button. The telephone number ‘Phone no’ field edit window is opened by pressing <E> and the phone list is displayed, (section 3.2.3.8.1). Press <Esc> to quit the function and keep the old settings or <Enter> to quit the function and confirm the new settings.

<table>
<thead>
<tr>
<th>DEFAULT VALUE</th>
<th>ALTERNATIVE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>‘Phone no’: 3161717</td>
<td>Number 30 digits long</td>
</tr>
<tr>
<td>‘Connection’: Direct</td>
<td>Direct, Phone modem, Fiber</td>
</tr>
<tr>
<td>‘Serial port’: COM1</td>
<td>COM2, COM3, COM4, (COM3 &amp; 4 = Other)</td>
</tr>
<tr>
<td>‘Serial port’ SPA</td>
<td>SRIO, RAW</td>
</tr>
<tr>
<td>‘Baud rate’: 9600</td>
<td>300, 1200, 2400, 4800, 9600, 19200, 38400</td>
</tr>
<tr>
<td>‘Parity’: EVEN</td>
<td>NONE, ODD</td>
</tr>
<tr>
<td>‘Data bits’: 7</td>
<td>8</td>
</tr>
<tr>
<td>‘Echo’ IGNORE</td>
<td>IGNORE, CHECK</td>
</tr>
<tr>
<td>‘Hand shake’ NONE</td>
<td>RTS-idle, RTS/CTS, XON/XOFF</td>
</tr>
<tr>
<td>‘Stop bits’ 1</td>
<td>2</td>
</tr>
</tbody>
</table>

**Table 3.2.3.6.B** The options available for communication parameter file SPIN.CNF

The ‘SRIO’ protocol is used in conjunction with the SRIO 500M or 1000M or SACO 100M unit. The normal setting is ‘SPA’. RAW is reserved for future use.

‘Serial port’ will by toggling the <Space bar> toggle between COM1, COM2 and Other. The option ‘Other’ have to be edited by the <E> button and the addresses for the port given, displayed below:

```
<table>
<thead>
<tr>
<th>Other</th>
<th>I/O address</th>
<th>IRQ-level</th>
</tr>
</thead>
<tbody>
<tr>
<td>COM1</td>
<td>3PB</td>
<td>4</td>
</tr>
<tr>
<td>COM2</td>
<td>1PB</td>
<td>3</td>
</tr>
<tr>
<td>COM3</td>
<td>3PB</td>
<td>4</td>
</tr>
<tr>
<td>COM4</td>
<td>2PB</td>
<td>3</td>
</tr>
</tbody>
</table>
```

The ‘IRQ level’ and ‘hexadecimal I/O address’, has to be given manually, by <E> as in Edit when ‘Other’ is selected. An error message appears and the value will not be accepted, when an attempt is made to enter a hexadecimal I/O address ending with anything else than ‘0’ or ‘8’. The ‘COM1’ or ‘COM2’ are automatically given ‘I/O address’ and ‘IRQ level’
The telephone number entered in the field ‘Phone no’ is used by SMS 010 programs such as REPORT, DR-COM, which use automatic dialling, providing that the setting ‘Connection’ is set to ‘Phone modem’. The telephone number entered will be used.

**Notice !**

The default settings by SMS-BASE need no changes, when communication is done directly via cable to a SPACOM relay, which means that the user only have to confirm the settings by pressing <Enter>.

The default settings for SPA and SACO/SRIO data communicator are:

<table>
<thead>
<tr>
<th>Communication parameter</th>
<th>SPA default value</th>
<th>SRIO default value</th>
</tr>
</thead>
<tbody>
<tr>
<td>‘Phone no:’</td>
<td>161707</td>
<td>161707</td>
</tr>
<tr>
<td>‘Connection:’</td>
<td>Direct</td>
<td>Direct</td>
</tr>
<tr>
<td>‘Serial port:’</td>
<td>COM1</td>
<td>COM1</td>
</tr>
<tr>
<td>‘Protocol:’</td>
<td>SPA</td>
<td>SRIO</td>
</tr>
<tr>
<td>‘Baud rate:’</td>
<td>9600</td>
<td>1200</td>
</tr>
<tr>
<td>‘Parity:’</td>
<td>EVEN</td>
<td>NONE</td>
</tr>
<tr>
<td>‘Data bits:’</td>
<td>7</td>
<td>8</td>
</tr>
<tr>
<td>‘Echo:’</td>
<td>IGNORE</td>
<td>IGNORE</td>
</tr>
<tr>
<td>‘Hand shake:’</td>
<td>NONE</td>
<td>NONE</td>
</tr>
<tr>
<td>‘Stop bits:’</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 3.2.3.6.C Default settings for SPA and SACO/SRIO data communicator

**Note !** Check with the SACO/SRIO data communicator, when used, for actual settings ! Also refer to Appendix D in this User’s Guide.

When using phone-modems the other communication settings are still needed, because after the modem connection is established the communication between the PC and the relays/data communicators is performed according to these settings.

2 **Modem phone numbers**

The telephone number for a modem can also be entered as text at the 'Object/Bay' level. The procedure for changing the number is then as follows:

- Mark the selection ’MODEM PhNo:...’.
- Open the editing field by pressing <E>.
- Edit the telephone number as if it were a text.

Refer to section 3.2.3.8 for details regarding implementation of phone-numbers.
3. Module slave numbers

The SPA node number, (slave address), is changed at the lowest level in the application structure for each module as follows:

- Mark says:
  'SPCJ 4D29 Overcurrent and earth-fault module'

Press <Enter> and following screen appears:

```
Setting SPA.COM slave address
The module address is: 001
New address: 2.34
Or to set the address? Y
```

- Enter the 'New address' and confirm by pressing <Enter>.
- The question 'OK to set the address?' appears. Press <Enter> to replace the old address by the new one. Undo with <Esc> or any key other than <Y>.

- Follow the same procedure when wanting to check the SPA node number, (slave address), but exit with <Esc>, without changing anything.

(All slaves within a loop must have unique addresses).

This procedure only changes the address in the application structure in the PC. The existing address of the corresponding physical module is not changed! The procedure for transferring the changes to the physical module is given in section 3.2.1.2, but then has the current slave address to be correct, both in the application structure as well as for the one used in the physical module, otherwise no contact be established.

### 3.2.3.7 Adding new elements to the application structure

New elements are added to the application structure at the time of commissioning SMS-BASE, but also later when the structure of the power system is to be changed.

This section covers how to add a new station to the application structure. Use the corresponding parts of the following procedure for making minor changes to the structure. Note that the telephone number for a modem and the serial communication settings should only be entered once for each station and that the SPA node number should be entered at the lowest level for each part of a unit or relay.

**In Brief:**

Procedure for adding a ‘station’, ‘Object/Bay’ and a ‘Unit’, (‘Unit’ = Relay)

1. Add the ‘Station’.
2. Add the ‘Object/Bay’ item.
3. Set the serial communication parameters for the station, (once for each station)
4. Add the units of each object or bay.
5. Enter the SPA node number, (slave address), for the modules.
6. Enter the modem settings and the telephone number, when needed.
The following is in the example below to illustrate the operations by the user:

<table>
<thead>
<tr>
<th>Station</th>
<th>Dolben</th>
</tr>
</thead>
<tbody>
<tr>
<td>Obj/Bay</td>
<td>Cubicle 34 H 12  Grundvik 2</td>
</tr>
<tr>
<td>Unit</td>
<td>SPAC 535 C1  feeder protection</td>
</tr>
</tbody>
</table>

1. Add the station

- Start with the initial view at the station level in after selecting ’Alter application structure’:

  Select Station

  Vikom 1

- Press <A> for ‘Add’ to open the window for entering the station text and confirming it by pressing <Enter>:

  Create a new menu item

  Dolben

2. Add the ‘Object/Bay’ element

- Select the new station ‘Dolben’.
- The ‘Object/Bay’ level is now displayed and is empty:

  Select Object/Bay

- Press <A> to add a new object or bay.
- This opens a window similar to that for adding a station.
- Enter the object or bay text.
- Repeat the operation, if several objects or bays have to be entered.

3. Set the communication parameters for the station

SMS-BASE v. 2.0 is efficiently encouraging the user to set the communication parameters whenever adding an ‘Obj/Bay’ element and no communication parameter file SPIN.CNF for the ‘Station’ has been created before.

Fig. 3.2.3.7.A  Adding a menu item to the ‘Select Obj/Bay’ level

Since no communication parameters have been set for this station is the user queried for whether to create the communication parameters at once or later on.
Enter the settings for the serial communication port for each station. It is recommended to only set communication parameters at the station level.

If a station has several SPA loops with different modems and telephone numbers, the relays in each loop should be treated as if they belong to different stations. Therefore add a station for every SPA loop. Enter an extra comment in the station text, for instance:

'ABB1234 NICE VIEW SPA-Loop 1'
'ABB1234 NICE VIEW SPA-Loop 2'

- The procedure for entering new settings for a serial communication port is the same way as for changing them (see Section 3.2.3.6). The selection text begins with 'Create communication parameters' until the parameters have been set,(saved) once.

How to set the communication parameters are described in section 3.2.3.6.

4. Add the Units of each Object/Bay

- Press <A> to add a new unit.
- Select the unit to be added, e.g. feeder protection SPAC 535 or distance relay RELZ 100, by pressing <Enter>.
Explanation of the options in figure 3.2.3.7.C

A Manual Entry
A manual entry enables the operator to compose own relays. See section 3.2.3.7.1.

MODEM
The modem is installed as a unit for on demand calling.

OTHER RELAY
Other relay is a relay not able to communicate with SMS 010. The operator gets 8 screens to use for relay information. See section 3.2.3.9.

DRCOM STATION
A DR-COM station that is needed in the application for setting DR-COM.

REPORT STATION
A REPORT station that is needed in the application for setting REPORT.

REL 561
The REL 561 relay.

SACO 16A3
The SACO 16A3 analogue input annunciator.

SPAA 121
The SPAA 121 relay.

etc.

- Press <Y>, and the modules for the unit are added automatically.

Fig. 3.2.3.7.D Application structure window for adding a SPAC 535 unit

![Application structure window for adding a SPAC 535 unit](image)

Fig. 3.2.3.7.E Application structure window for adding a RELZ 100 unit.

Before the units in Fig. 3.2.3.7.D and E are added is the user asked for if he wants to give another default address than 1 to the modules at the adding procedure. (Gives same address to all modules).

Setting SPACOM slave address

Old Default address: 001
New Default address: 46

Fig. 3.2.3.7.F Setting the slave address for all modules within the unit

Note ! The addresses will be the same for all the modules in the same unit, but the slave addresses can be given separately for each module on the module level. E.g. REL 511 have a lot of dataparts that use the same address and then is this function useful for setting the same address to all dataparts.

Warning
Please set the address for the new 'Mod/Part' under the new 'Unit' 'SPACOM 535 C1 Feeder Terminal'. (The 'Mod/Part's was added automatically.)
Press any key to continue...

Fig. 3.2.3.7.G Don’t forget the set the individual slave addresses !
4.1 Adding the modules for each unit

- Selecting one of the new units opens the following window:

```
<table>
<thead>
<tr>
<th>Select Module/Part of Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPCJ 6D3 Control module</td>
</tr>
<tr>
<td>SPCJ 4D29 Overcurrent and earth-fault module</td>
</tr>
<tr>
<td>SPCJ 3C4 Directional earth-fault module</td>
</tr>
</tbody>
</table>
```

- 'S = show relay transducers’ shown in the menu bar refers to a list with the input c.t. and v.t. data for the relay. Refer to Section 3.2.1 for the procedure for changing the setting.
- Note! Modules can also be added manually by pressing <A>.

5. Entering SPA node numbers, (slave address), for modules

- Every module should have an unique SPA node number.
- Select one of the new modules, e.g. 'SPCJ 4D29...'.
- Enter a node number in the window as described in section 3.2.3.6.
- Repeat the same operation for each module.
- Repeat the whole operation for each unit.

6. Entering the modem settings and telephone number

- Enter the modem and phone number for each station just at the 'Object/Bay' level.
- Refer to Section 3.2.3.8 below for further details on modems.

3.2.3.7.1 How to manually compose/add/delete relays and modules

Sometimes it becomes necessary to make changes to a standard relay or even compose an "own" relay. E.g. in following cases: SACO units are extendible and more units can be added, or SPAC 5xx gets a disturbance recorder installed, meanwhile an other module is removed.

The start is always the same: 'UTILITIES/Alter application structure', but then there are two ways for modifying relays:

**METHOD A)** Select or add a standard relay as shown in figure 3.2.3.7.A

**METHOD B)** Add 'A Manual Entry’. This option is found in the 'Select unit' window as shown in 3.2.3.7.A.

**NOTE!**
It is impossible to add any module if it hasn’t been installed to SMS 010 in connection with some SM/... relay package!

**METHOD A)**

This method is the easiest and fastest. Start by selecting the relay at the unit level to be modified or add the standard relay. The modifying is done at the 'Module/Part of Unit’. Adding of modules is performed by the <A> button, while removing is done by the <D> button.
CHAPTER 3

The adding is done as follows:
At the module level press the <A> button and the window in figure 3.2.3.7.1.A is shown.

Fig. 3.2.3.7.1.A    Creating a new menu item

Next step is to fill in the window shown in figure 3.2.3.7.1.A according to the example. It is of most importance to give the module name correctly.

Syntax: Module type+<Space>+Module identification+<Space>+Additional text

Examples of accepted writing:
SPCR 8C27 explanatory text
saco 16d2b explanatory text
sPCj 4d29 explanatory text

Fig. 3.2.3.7.1.B   The new menu item is given correctly!

Repeat the adding of modules until the operation is finished, (don’t forget to give the slave addresses).

In the SPCR 8C.. cases, these modules can also be added as an standalone units directly from the 'Select unit' window shown in figure 3.2.3.7.C.

METHOD B)

This method is using the same principles as in method A, but it starts at a higher level, the 'Unit' level, by adding 'A Manual Entry', see fig. 3.2.3.7.C. This text can be edited to fit the user by the <E> button. Continue to the next level by selecting this item and add the desired modules by typing them as explained in method A.

3.2.3.8    Entering modem and telephone numbers

There are three methods of giving phone-numbers in SMS-BASE

1. Using the communication parameters phone-number list, 3.2.3.8.1
2. Phone-number as a part of the application structure, 3.2.3.8.2
3. Giving the phone-number within manual modem window, 3.2.3.8.3

3.2.3.8.1    Using the communication parameters phone-number list

This section describes the use of the phone-list. The phone-number list is a list containing the phone-numbers defined by the user. The list is the same for any application structure started from the same SMS-BASE installation. The phone list enables the user to change between the phone-numbers. Explanation texts or comments can also be given.
The phone list is started within the ‘Alter application structure’/<C> communication parameters and by marking the phone-number field by the cursor and pressing the <E> button. This also normally means that the editing of the phone-number list is done from the ‘Station’ level, (‘Select Object/Bay’ level), where the valid communication parameters normally are.

<table>
<thead>
<tr>
<th>Phone no:</th>
<th>161707</th>
<th>161707</th>
</tr>
</thead>
<tbody>
<tr>
<td>Connection:</td>
<td>Phone-modem</td>
<td>Phone-modem</td>
</tr>
<tr>
<td>Serial port:</td>
<td>COM1</td>
<td>COM1</td>
</tr>
<tr>
<td>Protocol:</td>
<td>SPA</td>
<td>SPA</td>
</tr>
<tr>
<td>Baud rate:</td>
<td>9600</td>
<td>9600</td>
</tr>
<tr>
<td>Parity:</td>
<td>EVEN</td>
<td>EVEN</td>
</tr>
<tr>
<td>Data bits:</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>Echo:</td>
<td>IGNORE</td>
<td>IGNORE</td>
</tr>
<tr>
<td>Handshake:</td>
<td>NONE</td>
<td>NONE</td>
</tr>
<tr>
<td>Stopbits:</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Fig. 3.2.3.8.1.A  The phone-number list is opened by marking the ‘Phone no:’ field and pressing <E>

Starting the function for the first time, means that the list is empty

The phone-list contains of two fields; name and number. If the amount of numbers exceed the size of the box, can the rest be scrolled by the <arrow> buttons.

Fig. 3.2.3.8.1.B  The phone-list with function bar

1. Adding phone-numbers

Numbers can be added by the <A> button. The user can give number and names, but the name/-s can be omitted because they are only comments that aren’t used for communication. 30 characters are allowed in the phone number and also 30 for the names/comments.

Syntax for the phone-numbers:

The phone-number characters are transferred by the PC to the modem exactly as they are entered. The telephone number will be evaluated by the modem according to the “AT convention”. This means that the telephone number can include the following characters:

|“(" ")” “_” “_” “_” “_” These four characters are ignored by the modem but enhance readability.|
|“W” ‘Wait for dial tone’. In the example the modem waits for a dial tone after ‘0’.|
|“.” Wait for 2 s and continue with the remainder of the number. Use this with a bad telephone line or a slow exchange.|

Example: 8 w |012| 987 - 6543
CHAPTER 3

Add a new item

Number: 613163160
Name: Station Gluf

Fig. 3.2.3.8.1.C Adding the number to station Gluf

Are You Sure [Y/N] ? Y

Fig. 3.2.3.8.1.D Confirm the adding of the number to the list

2. Changing to another phone-number.

Open the phone-list and mark the wanted number by the cursor and press <Enter>

<table>
<thead>
<tr>
<th>Name</th>
<th>Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kantvik Matningsstation</td>
<td>77896</td>
</tr>
<tr>
<td>Station Glerum</td>
<td>0 W [032] 987 - 6754</td>
</tr>
<tr>
<td>Station Gluf</td>
<td>613163160</td>
</tr>
</tbody>
</table>

A=Add D=Delete E=Edit S=Sort ESC=Exit ENTER=Select

Fig. 3.2.3.8.1.E Selecting the phone-number to use in the serial communication parameter file SPIN.CNF

3. Editing a phone-number

If the number or names added have changed, were faulty or need modification, then is ‘E=Edit’ the option. Both number and name can be edited. Mark the number to be edited and press <E>. The number field is to be edited first, finish by <Enter> and edit the name field. Finish by <Enter>.

Edit the item

Number: 234456
Name: Station Gluf

Fig. 3.2.3.8.1.F Editing the number 234455

After the editing is done, ended by <Enter>, has the changes to be confirmed. The user can compare the old and new data before accepting.

Edit

Old:
Number: 234456
Name: Station Gluf

New:
Number: 234457
Name: Station Gluf

Are You Sure [Y/N] ? Y

Fig. 3.2.3.8.1.G Confirm the changes!

Page 36
4. Deleting a number

Phone-numbers in the list can also be deleted. Mark the number in question and press <D>. When deleting a phone-number that is confirmed and listed in the ‘Phone-no’ field in the communication parameter file, (SPIN-CNФ), the number will be removed from the list, but remains in the SPIN-CNФ file! This is due to the fact that ‘Phone no’ field in the communication parameter settings cannot be empty.

![Deleting the selected number and corresponding names.](Fig. 3.2.3.8.1.H)

5. Sorting the phone-numbers

For the convenience of the user can the numbers be sorted. Sorting is done alphabetically. The items has to be more than two, otherwise is sorting not supported.

![Sorting the items in the phone-list](Fig. 3.2.3.8.1.I)

6. Exiting the phone-list

After finishing the operations in the phone-list is exiting done by <Escape>. Before returning to the communication parameters menu, has the user to define whether he wants to save the changes to the phone-list or not.

![Are you sure?](Fig. 3.2.3.8.1.J)

Assume that a new number is given, (added), and selected by pressing <Enter>, but the user don’t want to save the list. This results in that the list remains unchanged, but the new number selected, (but not added), is used as phone-number in the communication parameter file SPIN-CNФ.

### 3.2.3.8.2 Phone-number as a part of the application structure

SMS-BASE processes modems as a station resource in the same way as relays. There should be only one modem and one telephone number per station element. Add more elements where a station has several modems, telephone numbers or SPA loops.

**Note!** A modem is only added to the application structure as a 'Unit', if it is going to be used for on-demand dialling to the station, e.g. REPORT automatically uses the telephone number entered into the 'Communication parameters’ window for the station.

Modems have to be entered according to the rules for 'Alter application structure' and some operations may appear illogical.
In Brief:

The procedure for adding a modem for on-demand dialling:

1. Enter 'MODEM PhNo: at the 'Object/Bay' level.
2. Select MODEM from the 'Quick add' window at the 'Unit' level.

The telephone number in the example in this Section is 'PhNo: 0 W (012) 987 - 6543'.

1. Entering 'MODEM PhNo' at the 'Object/Bay' level

- Insert the modem at the bottom of the 'Object/Bay' list.
- Move the cursor to the bottom of the list and press <A>.
- Enter a text of less than 33 characters for the telephone number and confirm by pressing <Enter>:

   ```
   Create a new menu item
   MODEM PhNo: 0 W (012) 987 - 6543
   ```

   Enter the telephone number exactly as written above. All the characters next to 'PhNo:' are considered to be part of the telephone number. The characters are transferred by the PC to the modem exactly as they are entered. The telephone number will be evaluated by the modem according to the "AT convention". This means that the telephone number can include the following characters:

<table>
<thead>
<tr>
<th>Character</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;(&quot;</td>
<td>Wait for dial tone. In the example the modem waits for a dial tone after '0'.</td>
</tr>
<tr>
<td>&quot;W&quot;</td>
<td>&quot;Wait for dial tone&quot;. In the example the modem waits for a dial tone after '0'.</td>
</tr>
<tr>
<td>&quot;,,&quot;</td>
<td>Wait for 2 s and continue with the remainder of the number. Use this with a bad telephone line or a slow exchange.</td>
</tr>
</tbody>
</table>

- A list of the following format now appears at the 'Object/Bay':

   ```
   Select Object/Bay
   MODM PhNo: 0 W (012) 987 - 6543
   ```

2. Selecting 'MODEM' at the 'Unit' level

- Select 'MODEM PhNo:...' at the 'Object/Bay' level.
- An empty list appears at the 'Unit' level.
- Press <A> and select 'MODEM' at this level.

   This level is omitted in normal operation, since no selection is required if the alternatives are less than two.

3.2.3.8.3 Giving the phone-number within the modem window

When utilising the application structure for communication, start from ‘SELECT’ SMS-BASE select menu is a function button 'F3 MODEM' displayed at the ‘Station’ level. By starting this can the user manually dial and hang-up the phone-modem. The number displayed is the one found in the serial communication parameter file SPIN.CNF, but the user can also give another modem phone-number. This number isn’t saved anywhere. These operations are thoroughly described in chapter 4.
3.2.3.9  **Entering a non-SPA relay**

This module is useful, for example, for documenting the settings for a relay that has no SPA communication. The relay type or manufacturer is of no consequence, because the data is always saved in the same format and cannot be transferred to any other location.

The data can be saved in a file or printed. The module is added to the application structure in the same way as all other units. When in the ‘Alter application structure’ utility, the unit ‘OTHER RELAY …’ is added exactly the same as, for example, ‘SPAC 535’.

**Fig. 3.2.3.9.A**  Adding a non-communicating relay

When selecting the module in the application structure, there are 8 empty screens for entering the relay data. The screen below shows an example for a solid-state overcurrent relay with the designation SPAJ 1B1 J3.

**Fig. 3.2.3.9.B**  Information about the SPAJ 1B1 J3 added by the user

3.2.3.10  **Entering a relay without settings**

This module called ‘SPAJ 140 DEMO’ is useful for training, since it does not permit the user to send changed settings to any module. The module is basically the same as ‘SPCJ 4D29’, but does NOT include the command ‘Send and receive parameters’ and ‘Resetting of output registers’. The procedure for adding this relay is the same way as for ‘SPAJ 140 C’ with the exception that SPAJ 140 DEMO is selected instead in the ‘Unit’ window.

**Fig. 3.2.3.10.**  Module SPAJ 140 DEMO

3.2.3.11  **Supplementary system elements for historical data or sets of settings**

The parameters for a module are kept in two files, one for the acquired variables, VALUES.INS, and one for the changed settings, VALUES.VAL. The existing files are overwritten at the conclusion of new data acquisition or setting operations. This will naturally make it impossible to keep the old values stored in the module in the application structure.

It is then possible to add more units to the application structure and make another ‘Receive parameters’ from the modules or set the module parameters from the beginning. Another possibility is to use the ‘M-Copy-Mark’ function for copying units in the application structure and adding them again to the application structure. In both cases can unnecessary modules be manually removed. M-Copy Mark is described in section 3.2.3.15.
CHAPTER 3

When making copies, which not are intended for “active” use, keep them outside the ‘REPORT and DR-COM stations.

In order to save old data or prepare a supplementary set of parameter settings, a separate 'Unit' element has to be created. In such instances, note in the text belonging to the element what it contains.

An interesting fault in 'Vikom 1', 'H22 Grundvik 1' and 'SPAC 535 C...' can be saved as follows:

- Add an extra unit at the 'Unit' level:

  Select Unit
  SPAC 535 D1 Feeder terminal
  SPAC 535 D2 Disturbance 95-09-29

  The following options appear below SPAC 535 C Disturbance 95-09-29:

  Select Module/Part of Unit
  SPTO 6D3 Control module
  SPCJ 4D29 Overcurrent and earth-fault module
  SPFS 3C4 Directional earth-fault module

  - Now acquire the data of the fault you wish to record as described previously.

A supplementary set of relay settings can be stored and activated when the corresponding system conditions prevail. For this purpose a further alternative is added to the list of menu options either as a supplementary element at the 'Unit' level or as an extension of the corresponding module. The latter case is illustrated for the 'SPCJ 4D29' module below:

- Select the level 'Module/Part of Unit'.
- Move the cursor to 'SPCJ 4D29...'.
- Add the new option for the supplementary set of settings with a comment on its use: 'SPCJ 4D29 If Line 4 is out of operation'
- The 'Mod/Part' list now becomes:

  Select Module/Part of Unit
  SPTO 6D3 Control module
  SPCJ 4D29 Overcurrent and earth-fault module
  SPCJ 4D29 If line 4 is out of operation
  SPFS 3C4 Directional earth-fault module

  - The parameter settings can now be downloaded by a remote command from either the standard 'SPCJ 4D29...' or the supplementary set as described previously.

3.2.3.12 Deleting a selection

Any part of the application structure can be deleted. Deleting starts at the selected alternative in one of the lists and continues down the branch of the structure.

For instance, if a station is deleted all its components and the station text are deleted. If a unit is deleted then all of its modules and text are deleted.
The deletion procedure is as follows:

- Position the cursor at the highest alternative in the branch to be deleted.
- Press <D>.
- Confirm the deletion by pressing <Y> in response to the question 'OK to delete this item and the substructure under it?'
  or exit without deleting by pressing <Esc> or any key other than <Y>.
- Wait for a one or two seconds while the deletion is executed.

Should it become necessary to delete the entire structure, the data should be deleted station by station, or do it by 'UTILITIES'/'Alter application structure'/'S = AS Handling'/'D=DELETE'.

3.2.3.13 Creating a new application structure

There are two possibilities to create a new application structure. The application structure has, however, always to be configured via 'Alter application structure'.

Method one is from SMS-BASE menu 'SETUP'/Appl. struct path', 3.2.3.13.1
Method two is by 'Alter application structure'/S = AS Handling'/A = ADD', 3.2.3.14.2

3.2.3.13.1 Using ‘SETUP’/’Appl. Struct. Path’

‘SETUP'/Appl. struct path’ is mainly used as an easy method for changing between already existing application structures, when the user already remembers the path. It enables also the user of creating new applications.

The procedure for creating a new application structure is as follows:

- Go to the main menu and select ‘SETUP’ followed by ‘Appl. struct. path’ to display this menu:

  ![Application structure path: C:\SMS\DATA-EX2](image)

- Enter the path, e.g. 'C:\SMS\DATA-EX2', to be used for the new structure and press <Enter>. It is recommended that the path be of the format 'drive\SMS\<Application structure home directory'> to enable SMS-BASE data to be easily located.

- Save the new path by ‘SETUP’/’Save setup’

- Select 'Alter Application structure' from the 'UTILITIES' menu.

  ![Select](image)

Fig. 3.2.3.13.1.A Selecting ‘Alter application structure’
The application structure isn’t valid yet, but can be confirmed to be a valid application structure:

```
Warning
The current AS is not valid.
The current AS can be made valid. It is also possible to create a new AS, or to select an existing AS.
[Scanning will give a list of valid Application structures].
Do you want to make the current AS valid [Y/N] ? Y
```

You may give a comment for the application structure in question:

```
Validate Application structure
Comment to current Application structure
Comment: This is the valid structure...
```

- Construct the new application structure as instructed in section 3.2.3.7.

### 3.2.3.14 AS Handling

The 'Alter application structure' has a sub function called ‘S = AS Handling’, that provides the user with efficient tools for coping with application structures. The ‘S = AS Handling’ functions enables the user to when necessary replicate the application structures between portable PC’s or solid PC’s in the station. This is done by using floppy disks and by compressing if necessary. Scanning for application structures is another exciting possibility, as well as adding and deleting. Current application structure cannot, by natural reasons, be deleted.

![Fig 3.2.3.14.A](image)

Starting up ‘S = AS Handling’ from ‘Alter application structure’ at ‘Organisation’ level

‘S = AS Handling’ is started from the ‘Organisation level’ in ‘Alter application structure. When starting the ‘AS Handling for the first time there might be no applications listed in the AS Handling window. This is easily corrected by using the ‘S = SCAN’ function.

The syntax for working in ‘S = AS Handling’ is that the application structure that is to be copied, deleted, packed etc. has to be marked by the cursor. One application structure at a time can be selected. The full path for the listed application structures are displayed. The amount of free diskspace is also shown, unit is kB.
Important!

Regarding adding, copying and unpacking, (deleting as well), of application structures is following to be noticed:

AVOID in any case to locate any application structure directly on the root of the harddisk where your SMS 010 system or other important programs are located. The problem that might occur is when deleting, because the ‘D = DELETE’ tool is powerful and erases any application structure that the user want to delete, including hidden files!

Definitions:
An application structure home directory contains at least the files SPACOM.CNF, CHOICES.DSC and some ‘Organisation’ directory, e.g. ‘O001’ and has an own name, e.g. DATA-EX

Examples:
C:\SMS\DATA-EX is the full path where DATA-EX is the application structure home directory, i.e. application structure name.

C:\SMS\ETT\TU\TRE is containing many directories, but TRE is the one regarded as the application structure name, i.e. the first one that will be deleted in a deleting operation.

After the application structure name are the application directories located, e.g. C:\SMS\DATA-EX\O001\S002\U007\...

3.2.3.14.1 Changing to another application structure

The user can choose an application structure by marking an application structure listed in the AS Handling list and pressing <Enter>. By confirming the selection will selected application structure be in active use by SMS-BASE programs.

If the selected application structure for some reason no longer is a valid application structure will it be removed from the list and the previous application structure is still in use.

Another possibility to change between application structures is to directly in the SMS-BASE program from the SMS-BASE ‘SETUP’ menu select the option ‘Appl. struct path’ and giving the proper path for the application structure. Finish by ‘Save setup’. (Refer to section 3.4.4)
3.2.3.14.2 Creating application structures

By using the ‘A = ADD’ function in AS Handling can new structures be added. The structure created has to be selected as the current application path.

Following recommendations are given for creating:

1. Avoid unnecessary amount of directory names. Use as few as possible.
2. Maximum directory length allowed by DOS is 8+3 characters.
3. Do never locate your application structure at the root where your operating system or other important data is located. Use e.g. C:SMS as "platform" for your application structure.
4. Do not locate your application structure inside another application structure.
5. If you create an application structure on a floppy disk, e.g. A: or B: might following difficulties occur: There is not enough space on the floppy for whole the structure or you will be astonished how slowly everything operates while working on the floppy disk.

After the application name is given, created and selected, (= made current application structure), the <Esc> button will return to the usual ‘Alter application structure’ functionality. The five level structure is always required, i.e. you will always have to add elements to every level in the application structure, otherwise it is not complete.

Fig 3.2.3.14.2.A Giving new application structure name

Fig 3.2.3.14.2.B Confirming the creating of the application structure home path \MYSTRUC\n
Fig 3.2.3.14.2.C Decide whether to start using the application structure or only creating it for later use

The configuration of the new application structure can be done as soon as the AS Handling session is finished. <Esc> will exit and the current application structure is shown.
3.2.3.14.3 Copying application structures

The application structures listed in the AS Handling list can be copied to any empty or not existing directory. What to do when wanting to copy an application structure elsewhere, but not listed in the list? Then can this “missing” application structure be added by using the ‘S = SCAN’ function and giving the scanning path. This enables both copying to/from a floppy disk to/from the harddisk. If the application structure is bigger than 1.44 MB it cannot be copied to a floppy.

Fig 3.2.3.14.3.A Copying selected application structure DEMO1 to the path defined

![Copy Application structure](image)

Copy: C:\SMS\DEMO1
Path: C:\sms\DEMOORIG
Comment: Saving original versi

Fig 3.2.3.14.3.B Confirm your intentions to copy DEMO1 to DEMOORIG

![Copy Application structure](image)

Target directory does not exist. Create?
Are You Sure [Y/N] ? Y

Fig 3.2.3.14.3.C The target directory ‘demoorig’ doesn’t exist, it has to be created

![Copy Application structure](image)

Fig 3.2.3.14.3.D At last, copying the application structure in ..\DEMO1\ to ..\DEMOORIG\.
3.2.3.14.4 Deleting application structures

Application structures can also be deleted, when they have become unnecessary. Everything, also hidden files, in the application structure will be deleted by the ‘D = Delete’ command. Deleting is started from the application structure home directory.

Examples:

Deleting the application structure C:\SMS\DATA-LEN will delete from and including ..\DATA-LEN\Oxxx\Sxxx\Uxxx\Mxxx. In the structure C:\SMS\STRUCS\MYRTLE\FAR... is deleting started from \FAR, because it is regarded as the home directory for the application structure \FAR.

(An application structure home directory contains at least the files SPACOM.CNF, CHOICES.DSC and some ‘Organisation’ directory, e.g. ‘O001’. This means followingly that unnecessary and empty directories in the beginning of the application structure path, before the home directory will remain undeleted.

Warning!
If you have an application structure located directly on e.g. the C: root or other root will the harddisk be totally cleaned by deleting the application structure! Do never delete any application structure that use only e.g. C:\SMS\ or C:\SMS\BASE as application structure names. (However, it isn’t normally possible to use these directories, while creating, because the directories aren’t empty).

A:\ or other drives can without trouble be deleted, when used as application structures only.

If you have some application located at a difficult location, e.g. the root of the C:\ drive, then will deleting have to be manually performed by e.g. DELTREE <application structure directory> on the DOS command line, (DOS version 6.0 or later), or from Windows file manager. In that case has deleting to be done starting from the ‘Organisation’, (O0n\...), directories.

Fig 3.2.3.14.4.A Do you want to delete the application structure in DEMO1?

Regarding directories created from e.g. DR-COM for uploaded disturbances, will these directories have to be deleted manually, unless they for some unfortunate reason would be located inside some application structure.

3.2.3.14.5 Editing application structure comments

The application structures listed in the AS Handling list are equipped with explanation texts, called comments.

These comments have no other purpose than to give explanations for the applications listed. The texts can be edited by the ‘E = EDIT’ function.

Fig 3.2.3.14.5.A Editing the comment to an application structure
3.2.3.14.6 AS Handling help function

The AS Handling function is equipped with an eminent 'H = HELP' function, that is giving information and guidance about the functions available in 'S = AS Handling'.

3.2.3.14.7 Scanning for application structures

When starting the 'S = AS Handling' function for the first time will the AS Handling list usually be empty.

The function 'S = SCAN' is a tool for searching after application structures. The application structures can be located anywhere; on the harddisk or on floppies. Except from giving the SMS 010 user control over his application structures, another reason for scanning is that some of the other functions cannot be done until the application structure concerned is included, (and valid), in the AS Handling list. It is not necessary to make the applications current, it is enough by marking with the cursor for copying purposes etc.

Scanning is performed according to the search path given, (C:\SMS\ C:\SMS\MINA\STRUKAR\ or A:\ etc.), but also according to some definitions. I.e. the application structure home directory has to contain both the files SPACOM.CNF and CHOICES.DSC, otherwise it is not fulfilling the minimum criteria for being a valid application structure.

Fig 3.2.3.14.7. A  Start scanning by <Enter> or ‘Y’

```
Application structure scan
Scan for application structures
Are You Sure [Y/N] ? Y
```

Fig 3.2.3.14.7.B  Give the path for where to scan for application structures

```
Scan for Application structure
Example: C:\ or C:\SMS\ Where To start: C:\ams\n
```

Fig 3.2.3.14.7.C  The first one is already found, and given the default comment Scanned #1

```
Scan for Application structure
A valid Application structure was found in C:\SSMS\nEKO1
Comment: Scanned #1
```

Fig 3.2.3.14.7.D  Scanned #1 is replaced by user made own remarks !

```
Scan for Application structure
A valid Application structure was found in C:\SSMS\nEKO1
Comment: Own remarks
```
CHAPTER 3

<table>
<thead>
<tr>
<th>Date</th>
<th>Comment</th>
<th>Path</th>
</tr>
</thead>
<tbody>
<tr>
<td>03/16/96</td>
<td>Own remarks</td>
<td>\C:\SMS\SMS01</td>
</tr>
<tr>
<td>03/16/96</td>
<td>Training structure</td>
<td>\C:\SMS\DATA-DO</td>
</tr>
<tr>
<td>03/16/96</td>
<td>Structure for Wasa</td>
<td>\C:\SMS\WASA</td>
</tr>
<tr>
<td>03/16/96</td>
<td>This is the installed example</td>
<td>\C:\SMS\DATA-EX</td>
</tr>
<tr>
<td>03/16/96</td>
<td>This was used for DR-COM manual</td>
<td>\C:\SMS\NANDOR</td>
</tr>
</tbody>
</table>

Free space on drive C: 3140 kB
Current Application path : \C:\SMS\DATA-EX\  
A=Add  C=Copy  D=Delete  E=Edit  H=Help  
S=Scan  P=Pack  U=Unpack  ESC=Exit  ENTER=Select

Fig 3.2.3.14.7.E  Five application structures were found at the scanning session

As appears is also the date for scanning given, but when creating new application structures, are the creating date automatically used. Note that always when starting a scanning session is the calculation done from Scanned #1, regardless of how many application structures there are from before in the list.

3.2.3.14.8  Packing application structures

The ‘P = Pack’ function enables the user to pack application structures into compressed *.ZIP files. This packing function becomes necessary when wanting to transfer an application structure from one SMS 010 installation to another and the application structure is too big to fit on a 1.44 MB floppy disk. The Pack function can also be used for making back-ups of the application structures. The application structure is compressed including the tree structure starting from the application structure home directory. The application structure home directory is not included. The user has to define where to locate the packed file to be created, e.g. at C:\SMS\ or A:\ etc. The extension ZIP is automatically added by the program unless other is specified.

The packing procedure is normally displayed on the screen, Fig 3.2.3.14.8.C, and the packing has to be acknowledged by the user, when packing is finished. The reason for this is that the user can read possible messages given during the packing procedure.

Start by selecting the application structure to pack by marking it by the cursor and press <P>.

Fig 3.2.3.14.8.A  Packing selected application structure, giving name and locating it on a floppy

Do not use wildcards in the names!

Note 1
Packing the file to the harddisk, e.g. C:\SMS\<name>.ZIP is faster than working directly on floppies. After the packing is ready can this packed file be copied to the floppy disk and the original file on the harddisk be deleted.

Fig 3.2.3.14.8.B  Confirm the packing operation

Page 48
The packing begins:

```
adding: c:\sms\0001\0001\CURAT01.UMI Deflating (65%), done
adding: c:\sms\0001\0001\0001\MO01\PACKAGE.PCK Deflating (38%), done
adding: c:\sms\0001\0001\0001\MO01\CHOICES.DSC Deflating (25%), done
adding: c:\sms\0001\0001\0001\MO01\MS02/ Storing (0%), done
adding: c:\sms\0001\0001\0001\MO01\CHANNEL.DSC Deflating (78%), done
adding: c:\sms\0001\0001\0001\MO01\EVENT.DSC Deflating (93%), done
adding: c:\sms\0001\0001\0001\MO01\LOGGING.DSC Deflating (94%), done
adding: c:\sms\0001\0001\0001\MO01\VALUES.VAL Deflating (74%), done
adding: c:\sms\0001\0001\0001\MO01\SLAVE.BAT Storing (0%), done
```

Fig 3.2.3.14.8.C A typical packing operation displayed on the screen

```
Pack selected application structure
Packing operation complete
Press any key...
```

Fig 3.2.3.14.8.D The packing session was successfully performed

Acknowledge the message in 3.2.3.14.8.D and the packing is completed. If unsuccessful, error messages are shown instead.

**Note 2** A packed file has to be unpacked before it can be used as an application structure. Refer to next section, 3.2.3.14.9.

**Packing manually:**

c:\sms>

c:\sms> base\support\pkzip -rpex! example.zip c:\sms\data-ex\*. *

In this case the file example.zip is created in the C:\SMS\ directory from the appl.struct. home path DATA-EX.

### 3.2.3.14.9 Unpacking application structures

Before a packed application structure can be used within SMS-BASE again it has to be unpacked by the ‘U = Unpack’ function. The file to unpack and its location has to be given as well as the destination and the new application structure home directory. The application structure is automatically restored with the tree structure, providing the free disk space on the target path is enough.

It is not supported to unpack an application structure back to the same application structure directory or to another directory with the same name. The user will have to define an application structure name that isn’t occupied already. Followingly can obsolete application structures be deleted, when replacing ones are added.

```
Unpack application structure
Path: A:\\aminskad.zip
```

Fig 3.2.3.14.9.A Unpack prompts for the location of the .ZIP file, location could be on the harddisk as well.

```
Unpack application structure
Give a new application structure home directory
Path: C:\sms\mentor
```

Fig 3.2.3.14.9.B Give name and location for where to unpack the application structure in the packed file
Fig 3.2.3.14.9.C  Start the unpacking by ‘Y’ or <Enter>

The unpacking procedure is rather similar to Fig 3.2.3.14.8.C.

After the unpacking is performed is the user asked for a comment to this new application structure:

Give comment for the new AS
Comment: This AS is from the station PC

Fig 3.2.3.14.9.D  Give comments for including into the AS Handling list

The unpacked structure is now ready for use.

Unpacking manually:
c:\sms\model2>...
In this case is the file example.zip unpacked to the C:\SMS\<model2> directory. Create the <New appl. struct. home directory> by MD, (Make Directory) command before unpacking.

3.2.3.15  Copying units within the application structure

Units can be copied within the current application structure by the function ‘M=Copy-Mark’. Any ‘Unit’ in the application structure can be copied and inserted again. The same ‘Unit’ can be installed several times at different locations, by selecting the source once. The buffer with the source ‘Unit’ to be copied is however emptied, when the deleting function, <D> is activated. This function is especially useful if the user have to configure many identical relays. The setting can be performed once and then copied into the same ‘Unit’ level or to any other ‘Unit’ level in the application structure.

The function enables also back-up functions. If only some modules within the source ‘Unit’ are worthwhile back-up, then can the unnecessary modules be manually removed by the <D> button at the ‘Module/Part of Unit’ level. The ‘Unit’ text can be edited before copying and the module addresses, (slave addresses), given in a flexible way.

Fig 3.2.3.15.A  Selecting the source to copy by cursor and <M> button
Fig 3.2.3.15.B  The paste option is at your disposal by opening the quick add window at ‘Select Unit’ level

By pressing <P> for Paste when the quick add of units is shown is the marked source unit going to be copied from the location given in the ‘Copy source’ box into the location where the cursor was before opening the add window.

Fig 3.2.3.15.C  The text for the unit to be added can be edited

Fig 3.2.3.15.D  Select how you want to give the slave addresses for all the modules in the unit to be copied

Slave number ‘equal for all modules’ means that all modules will become the same address. The ‘as original’ option will use the same slave addresses as in the source unit. The ‘manually per module’ option states the user to give the address for each module, but if the manual giving is interrupted by pressing <Escape> will the rest of the modules keep the original source addresses. When slave address is given ‘in ascending order’ is the address increased by one for each module starting from given value.

Fig 3.2.3.15.E  Giving slave addresses in ascending order starting from 22

After the addresses are set is the user prompted to start the copy process. The unit in question will upon a positive response from the user start the actual installation.

Fig 3.2.3.15.F  Confirm the copying of the unit
CHAPTER 3

The source unit can be copied:

![SPAA 341 C Feeder protection unit in bay 18
Wait while copying takes place...](image)

Fig 3.2.3.15.G The unit source unit SPAA 341 C is copied.

When making copies, which are not intended for “active” use, keep them outside the stations defined as REPORT and DR-COM stations, otherwise will also the back-ups be included.

3.2.3.16 Show transformer settings and relay inputs

The transformer settings can be viewed by ‘Alter application structure’/S = Show transducer file’, but editing is done from ‘UTILITIES’/Alter additional configuration’, (described in section 3.2.1)

At ‘Select Module/Part of Unit’ level is the ‘Show transducer values’ showed by pressing <S> as in Show !

![Select Organization][1]

Fig 3.2.3.16.A Show the transformer and relay input settings.

3.2.3.17 Working with data on floppy disks

Normally the application structure is installed on the hard disc. As an alternative data and the application structure can be stored on floppy disks. This has the following advantages:

- The data are only available to the person in possession of the floppy discs.
- The data can be processed on any PC with SMS-BASE.
- The storage of data can be clearly organised, e.g. one floppy per ‘Station’, (has to have five levels anyhow).
- Disk back-up is performed by the 'diskcopy' command.

The disadvantages of working from a floppy are that the system operates somewhat slower for certain operations and that one disc per station is needed.
The procedure for working with the application structure on a floppy is as follows:

1. Insert a new and 1.44 MB formatted floppy disk in drive 'A:'.

2. Change 'Appl. struct. path' as directed in Section 3.4.4 that the active path becomes 'A:\', or by 'S= AS Handling'. SMS-BASE will now always look for floppies in the A: drive and the path does not have to be entered every time a different floppy is inserted with a different application structure.

3. Change, add and delete items in the new structure until it corresponds to the station.

Recommendations:
- One floppy disc per station.
- Use the station name for the DOS 'Label' on each disc.
- Label the discs carefully with name of the station on each one.

3.2.4 Checking the application structure

'Check application structure' is an SMS-BASE utility program which is started 'UTILITIES'/'Check Application structure'. Its main purpose is to present the user a tree of the setting structure. The tree displayed by 'Check application structure' shows all the slave addresses, communication parameters and the location for the modules.

3.2.4.1 Using the ‘Check application structure’ menu

The ‘Check application structure’ menu includes the following functions:

- Data structure path: C:\SMS\DATA-EX
- Do not try repairing errors
- Check only
- Check and draw a tree

Fig. 3.2.4.1.A The ‘Check application structure’ menu

'Data structure path' shows the directory where the tree is located. Selecting it opens a window in which the user can enter a new path. If you wish to view a structure other than the one in use, type the path and press <Enter>.

'Do not try repairing errors' item defines whether the user wishes ‘Check application structure’ to correct errors. Selecting it opens a box with the alternatives and 'Repair errors, if possible'. The default setting is 'Do not try repairing errors'.

'Slave address checking level' specifies how multiple slave addresses shall be determined. Possible choices are 'No slave address checking', 'Slave address checking by communication parameters' and 'Slave address checking by level: ', the possible levels being 1, 2, 3, 4 or 5. 1 Correspond to ‘Unit’ level, 2 to ‘Obj/Bay’, 3 to ‘Station’, 4 to ‘Organisation’ and 5 to whole tree.

Fig. 3.2.4.1.B Selecting slave address checking levels

The last two items in the menu initiate processing. 'Check and draw a tree' scans the tree, displays a level report, displays the tree and prints any errors. The 'Check only' item does basically the same as 'Check and draw a tree' without actually displaying the tree.
If the structure is too large for the screen, the user can scroll the window up and down using the <arrow>, <PgUp>, <PgDn>, <Home>, <End>, <Ins> and <Del> function keys. <Home> brings to the first screen, while <End> brings to the last picture. <Ins> brings to the left, while <Del> brings to the right. F7 REPORT creates a printout to file or paper. When printing is done is it reason to notice that for the convenience of the user when scrolling the tree on the screen are the application trees wider than 80 characters, which means that by printing will two lines be required.

Example of a display generated by the 'Check and draw a tree' function:

```
Fig. 3.2.4.1.C  An application structure is examined
```

Example of a display generated by the 'Check only' function:

```
Fig. 3.2.4.1.D  An application structure report as 'Check only'
```

### 3.2.4.2 What does ‘Check application structure’ do?

‘Check application structure’ provides three kinds of reports: level, tree and error. The displays generated for the three reports are:

**Level report:**

```
Fig. 3.2.4.2.A  Level report for 'Check application structure'
```
3.2.4.2 Figure 3.2.4.2B: Tree report for ‘Check application structure’

As appears from Fig. 3.2.4.2C are I2 and I2Nom = 0, which means that the extended current transformers haven’t been set, because it is an application structure created by SMS-BASE v. 1.0 or 1.1. Most likely have the user set the values to ‘0’ because these modules don’t use them at all, (settings of no importance).

In Fig. 3.2.4.2C a message is displayed when the value is not set at all, thus is I2 or I2Nom not existing. Correct user set the values to ‘0’ because these modules don’t use them at all, (settings of no importance).

3.2.4.3 How does the ‘Check application structure’ check multiple slave addresses?

The slave addresses can be checked according to communication parameters or level or checking can be refused.

When checking according to communication parameters, a search is made for multiple addresses within one communication port. The following is an example of the display:

3.2.4.3.1 Figure 3.2.4.3A: Checking addresses for the same communication port
Although slave addresses 3 and 2 are also used in relay SPAA 323, they are not marked as multiple addresses because the relay has its own communication parameters. The following is an example of the display when checking level 1:

![Display example for checking level 1](attachment:image.png)

Fig. 3.2.4.3.B Checking addresses for level 1

In this example, level 1 is the relay level and therefore the slave address set inside each of the relays must be unique, which in this case they are.

The following is an example of the display when checking level 2:

![Display example for checking level 2](attachment:image.png)

Fig. 3.2.4.3.C Checking addresses for level 2
3.2.5 SPA- and Terminal emulators

The SPA_ and terminal emulators are quite similar, both are capable of reading and writing parameters to the relays, but the main differences are:

Only the Terminal emulator can be used for "on-line" communication with the SACO 100M, SRIO 500 M and SRIO 1000 M. In the Terminal emulator case complete messages has to be used, while the SPA-Terminal emulator uses only the parameters and settings when needed. Due to this feature the SPA-Terminal emulator is easier and faster to use with SPA relays. Both can be used for tracing communication difficulties, but it is however always important to make sure that the right cable is used between the PC comm.port and actual relay, when the reason for communication problems are solved. How to use these Terminal emulators is exemplified in sections 3.2.5.3.1 and 3.2.5.3.2.

**NOTE !** The changes made with these Terminal emulators are not linked to the application structure, because the Terminal emulators are working independently. This means that changed settings made in the relay modules will be valid also afterwards, (some relay demands an on-demand save parameter, V151), but the application structure with the relay settings and communication parameters will be unchanged, until a new read-out from the relays is made or the application structure settings are manually changed.

### 3.2.5.1 SPA-Terminal emulator

The SPA terminal emulator is activated by selecting 'UTILITIES' from the SMS-BASE main menu followed by 'SPA terminal emulator'. The routine provides the user with a convenient method of communicating with SPA relays without having to write the whole SPA protocol syntax.

<table>
<thead>
<tr>
<th>ABB Relays Strömsberg Power Ltd. RSO/SSa</th>
<th>SPA-program for SPACOM communication. Version 4.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type SPA-commands without address and checksum. For example: RF</td>
<td>not &gt;3RF:XX</td>
</tr>
<tr>
<td>To change a parameter value on line type -ParametreNew_Value</td>
<td>e.g. -a23 to change SPA address or -b1200 to change Baud rate.</td>
</tr>
<tr>
<td>To exit type CTRL/Z ENTER</td>
<td></td>
</tr>
<tr>
<td>com1 9600 bps 7 dbits parity-e No handshake SPA-addr-l</td>
<td>multi_line_answer=2 lower_case=0</td>
</tr>
</tbody>
</table>

Fig. 3.2.5.1.A The standard start-up screen for the SPA-Terminal emulator

In fig. 3.2.5.1.A the lowest to lines contain the actual communication parameter settings.

Syntax for using SPA-Terminal emulator:

The communication settings are changed by typing ".-" + corresponding letter + new value

Example: `-b4800` changes the baud rate into 4800 Baud.

Parameters that can be set:

- `-a` The address to which communication will take place
- `-b` Baud rate used
- `-c` The comm.port of the computer
- `-d` Databits
- `-p` Parity

After the setting of the comm.parameters is finished the communication to the relay modules can begin, when properly connected.

Messages: `R` = Read, `W` = Write

Refer to section 3.2.5.3.1 for detailed examples.
Exiting the SPA-Terminal session is done by the <Ctrl> Z command followed by <Enter>.

For example to view the data of relay modules types:

<table>
<thead>
<tr>
<th>Master - RF</th>
<th>SPA identity=F should give the relay module type as a reply.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Slave - &lt;1D:SPCJ 4D29:2X</td>
<td>This is an SPCJ 4D29 module !</td>
</tr>
</tbody>
</table>

### 3.2.5.2 Terminal emulator

The 'terminal emulator' function converts the PC into a straightforward terminal, which can send ASCII characters from the keyboard via the serial port to the host computer and receive characters from the host computer and display them on the screen. The function can be used to set and test communication equipment.

The terminal emulator is activated by selecting 'UTILITIES' from the SMS-BASE main menu followed by 'Terminal emulator'.

The first of two screens appears for configuring the communication channel:

![Configuring the Terminal emulator](image)

Fig. 3.2.5.A Configuring the Terminal emulator

- The five parameters are set in one continuous sequence. The different parameter values can for each parameter be scrolled using <Space> and the desired one selected by pressing <Enter> and next parameter will appear. The terminal screen appears after the last one has been set.

Syntax for Terminal emulator commands: Refer to section 3.2.3.5.2 for examples.

\[
\begin{align*}
\text{>} & = \text{Master sign, start character as well} \\
42 & = \text{Address of the unit} \\
R \text{ or W} & = \text{Read or Write} \\
\text{parameteridentification} & = \text{colon} \\
XX & = \text{Checksum}
\end{align*}
\]

Example: >75RS32:XX

**NOTE**! Use CAPITAL letters!
- The following alternative communication parameter settings are available:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>DEFAULT</th>
<th>SPA REQUIRE</th>
<th>ALTERNATIVE SETTINGS</th>
</tr>
</thead>
<tbody>
<tr>
<td>'Communication port:'</td>
<td>COM1</td>
<td>-</td>
<td>COM2, (COM3, COM4)</td>
</tr>
<tr>
<td>'Parity:'</td>
<td>EVEN</td>
<td>EVEN</td>
<td>NON, ODD</td>
</tr>
<tr>
<td>'Data bits:'</td>
<td>7</td>
<td>7</td>
<td>8, 6</td>
</tr>
<tr>
<td>'Stop bits:'</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>'Baud rate:'</td>
<td>2400</td>
<td>-</td>
<td>300, 600, 1200, 4800, 9600, 19200 Baud</td>
</tr>
</tbody>
</table>

SMS-BASE proposes the default values every time 'Terminal emulator' is selected. Note the settings required by the SPA loop. Two serial ports are always available. Ports COM3 and COM4 are only available if especially configured.

The following terminal screen appears after the last parameter has been set:

```
F1 to quit. F2 to change communication settings.
F3 toggle char/string mode (char mode)
AT
OK
>> AT
```

Fig. 3.2.5.2.B The terminal emulator screen

The characters entered at the keyboard are displayed in the lower half of the screen after the prompt ‘>>’ and the reply from the host computer in the upper half of the screen.

The example of Fig. 3.2.5.2.B illustrates the dialogue, when the microcomputer in a modem is acting as a host computer. A modem which understands "AT" commands should always reply to the command 'AT' with 'OK' or '0'. This dialogue serves to test the integrity of the communication between the PC and the modem.

The following function keys and their purposes are displayed at the top of the screen:

- **F1** quits the terminal emulator
- **F2** returns the user to the communication setting screen.
- **F3** selects the method of transferring characters entered at the keyboard:
  - 'Char mode' : Each character is transferred as soon as it is entered.
  - 'String mode' : A string of characters is transferred when pressing <Enter> to terminate the string.

Use this mode for SPA messages.
3.2.5.3 Examples for usage of SPA- & Terminal emulators.

Following sections aim at giving the operator some hints about using the SPA- and Terminal emulators.

3.2.5.3.1 SPA-Terminal emulator examples

In this example illustrates the communication between a SPAJ 140 C and the SPA-Terminal emulator. Refer to the SPAJ 140 C manual for listing of all available parameters, with settings, ranges and identity.

NOTE! Use the Terminal emulator for communication with the SACO 100M, SRIO 500M and SRIO 1000M data communication units.

R = Read, i.e. read parameters from the module, W = Write, i.e. send new settings for a parameter in the module. Sometimes the new settings has to be saved by the message WV151:1, consult the manual for information. Capital letters may also be used.

Conditions in this example:

A SPAJ 140 C relay with the module SPCJ 4D29 with slave address 12 is connected via a SPA-ZP-5A3 cable to the COM2 port of the PC.

To avoid mixing of comments with the original texts, the comments are written in bold.

Default settings:

```
com1 9600 bps 7 dbits pariteit-e No handshake SPA-addr=1
multi_line_answer=2 lower_case=0
```

Change the communication address to 12:

```
com1 9600 bps 7 dbits pariteit-e No handshake SPA-addr=1
multi_line_answer=2 lower_case=0 -a12
```

The address is changed to 12:

```
com1 9600 bps 7 dbits pariteit-e No handshake SPA-addr=12
multi_line_answer=2 lower_case=0
```

Change to comm. port 2 on the PC: (If you change to a port higher than the PC is equipped with, will the program immediately return to SMS-BASE)

```
com1 9600 bps 7 dbits pariteit-e No handshake SPA-addr=12
multi_line_answer=2 lower_case=0 -c2
```

A successful change of comm.port to COM2:

```
com2 9600 bps 7 dbits pariteit-e No handshake SPA-addr=12
multi_line_answer=2 lower_case=0
```

Identify the module with RF:

```
rf >12RF:13 (Master) <12D:SPCJ 4D29:2A (Slave)
```

The message >12RF:13 <12D:SPCJ 4D29:2A means: >12RF:13 is the message send by the master, (> sign), that has rounded the loop and returned. <12D:SPCJ 4D29:2A means that the slave, (< sign), is a SPCJ 4D29 module and has the address 12. 2A is the checksum.

What will happen if testing with the wrong address?

Change address to 13:

```
com1 9600 bps 7 dbits pariteit-e No handshake SPA-addr=12
multi_line_answer=2 lower_case=0
```

Page 60
The address is now 13, identification by RF:

```
>13RF:12
```

Identify yourself address 13!

The message >13RF:12 TIMEOUT***ERROR*** means that no contact has been established with address 13. The reason is not appearing, but usually some comm. parameters are set wrong, (check with the settings in the module on the front panel display), or wrong cable is used, etc.

Change back to the right address, 12:

```
>12RF:13
```

Let's check if connection is achieved.

Excellent! Next step is to make some parameter read-outs:

**Read parameter V201, (Baud rate):**
```
>rv201
```

**Answer:** 9600 Baud

**Read parameter V150 to find out whether the main or second settings are activated in the relay:**
```
>vw150:
```

**Answer:** 0, (zero), means that the main settings are activated.

If you want to activate the second settings, then type:
```
>vw150:
```

A check if the parameter setting has changed:
```
>rv150
```

**Answer:** 1 means that the secondary settings are in use.

Change back to main settings:
```
>vw150:
```

**Check:**
```
>rv150
```

Reading of parameter S29, which stands for the checksum of the switchgroup SGF1, main settings:
```
>rs29
```

**Answer:** The checksum is 24, which also can be confirmed locally from the module front panel display.

Exit by <Ctrl> <Z> and <Enter>
3.2.5.3.2 Terminal emulator examples

Two cases will be shown:

- Using the SPA protocol for communication with SPA protocol relays.
- Using the SACO 100 M protocol, (called SRIO protocol in SMS-BASE).

- Using the SPA protocol for communication with SPA protocol relays:

Start by selecting 'UTILITIES' /'Terminal emulator'

Continue by setting the communication parameters to be used.

Use the <F2> button to get change parameter mode.

<table>
<thead>
<tr>
<th>Communication set up. Space to change, Enter to confirm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Communication port: COM2</td>
</tr>
<tr>
<td>Parity: EVEN</td>
</tr>
<tr>
<td>Data bits: 7</td>
</tr>
<tr>
<td>Stop bits: 1</td>
</tr>
<tr>
<td>Baud rate: 9600</td>
</tr>
</tbody>
</table>

Fig. 3.2.5.3.2.A Settings for SPA relays communication.

Clarify the identity of the modules by the RF command, read parameter V201, (baud rate), see figure 3.2.5.3.2.B.

The commands are written in the lower half of the picture and the response on the upper half..

F1 to quit. F2 to change communication settings.
F3 toggle char/string mode (char mode)

->39RF:XX
<39D:SACO 16D2B:7F
>33RF:XX
<33D:SPCU 1C1:0E
>12RF:XX
<12D:SPCJ 4D29:2A

->12RV201:XX
<12D:9600:74

Fig. 3.2.5.3.2.B Identifying modules 12, 33 and 39 and also checking baudrate for slave 12.
• Using the SACO 100 M protocol, (called SRIO protocol in SMS-BASE):

In this example is the communication with a SACO 100M data communication unit illustrated, but SRIO 500M and SRIO 1000M are using the same principles, however the commands might differ.

This example comprises a SACO 148D4 including a SACO 100M data communication unit and some relays. (SPAJ 140 C with slave address 12, SPAU 320 C with the slave addresses 31, 32 and 33 and a SACO 16D1 with slave address 39). The relays are connected to the SACO 148D4 via optical fibres, while the terminal A of the SACO 148D4 is connected to the PC comm.port 2 via the cable SPA-ZP 13A3. The REPORT manual, appendix C, contains a summary for configuration of SRIO and SACO 100 M unit lists etc., also refer to corresponding manuals.

The address of the SACO 100 M happens to be 903 in this example, but it is not important to know when using the Terminal emulator, because it works anyhow! It is, though, very essential when using the REPORT program, for the REPORT station. If you do not know the SACO 100 M address it is quite easy to find out; make a RF request from a "dummy module". Example : >111RF:XX ⇒ answer <903N:4:7C. Fantastic! The SACO 100 M responds to the request by giving its own address, (in case the module isn’t existing or responding). The SRIO reveals its address by the SYSPAR 4 command.

Start by selecting 'UTILITIES' /'Terminal emulator'

Continue by setting the communication parameters to be used.

SACO 100 M to PC normally requires parity NONE and databits 8. In this case the PC comm.port is COM2. Remember when reading older manuals that CR = Carriage Return = <Enter>.

<table>
<thead>
<tr>
<th>Communication set up. Space to change, Enter to confirm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Communication port: COM2</td>
</tr>
<tr>
<td>Parity: NON</td>
</tr>
<tr>
<td>Data bits: 8</td>
</tr>
<tr>
<td>Stop bits: 1</td>
</tr>
<tr>
<td>Baud rate: 4800</td>
</tr>
</tbody>
</table>

Fig. 3.2.5.3.2.C Communication parameter settings for SACO 100 M.

Following step is to press <Enter>, because you need to get the "Ready-Go" sign ‘-‘, (minus mark), before you can start writing messages. Use capital letters and string mode, (achieved by the <F3> button), will send the messages as strings, when followed by <Enter>. Read the SACO 100 M unit configuration list by UNIT L command.
CHAPTER 3

F1 to quit. F2 to change communication settings.  
F3 toggle char/string mode (string mode)

-UNIT L
008.00:02:03  
012.00:02:01  
031.00:02:01  
032.00:02:01  
033.00:02:01  
001.16:00:01  
002.16:00:01  
003.16:00:01  
039.16:00:01  
-

>> UNIT L
>>

Fig. 3.2.5.3.2.D The UNIT L command shows the actual SACO 100 M unit list.

In Fig. 3.2.5.3.2.D 008.00:02:03 means that the slave address is 8, 00 = 0 channels, 02 = relay module and 03 means that the module is not responding! We will later on remove it from the unit list, while it is no longer existing. 039.16:00:01 stands for address 39, 16 channels, 00 = ON/OFF alarm channel module and the last 01 confirms that the module is answering and included in the event reporting. More information will be obtained from the SACO 148D4 manual.

Next measure is to remove the module from the SACO 100 M unit list, (the principle is almost the same for extending the unit list with more modules).

Type UNIT to get a row with this message: 000.00:00:00.  
Type the message 008.00.02.99  
Save it permanently with STORE

(99 means that it shall be removed from the list. In case of adding modules use 00 or 01 instead. 00 means that the actual module is incorporated in the system, but not comprised by the event reporting. 01 also enables event reporting).

F1 to quit. F2 to change communication settings.  
F3 toggle char/string mode (string mode)

032.00:02:01  
033.00:02:01  
001.16:00:01  
002.16:00:01  
003.16:00:01  
039.16:00:01  
-UNIT
000.00:00:00  
008.00:02:99  
000.00:00:00

>>
>> UNIT L
>> UNIT
>> 008.00:02:99
>> STORE

Fig. 3.2.5.3.2.E Deleting of slave no. 8 from the SACO 100 M unit list. STORE saves the change.

Page 64
F1 to quit. F2 to change communication settings.
F3 toggle char/string mode (string mode)
STORING...
STORING...
STORING...
STORING...
STORING...
STORING...
STORING...
STORING...
STORING...
STORING...

>> UNIT
>> 008.00:02:99
>> STORE
>>

Fig. 3.2.5.3.2.F Storing lasts about ½ a minute. STORING is shown during this action.

Checking: Has the module really been removed? Another UNIT L will make it clear, see fig. 3.2.5.3.2.G.

F1 to quit. F2 to change communication settings.
F3 toggle char/string mode (string mode)

- UNIT L
 012.00:02:01
 031.00:02:01
 032.00:02:01
 033.00:02:01
 001.16:00:01
 002.16:00:01
 003.16:00:01
 039.16:00:01
-

>>
>> UNIT L
>>

Fig. 3.2.5.3.2.G Yes! Unit no. 8 is no longer included in the unit list.

Sometimes it will be more efficient to increase the communication speed between the PC and the Saco 100 M. The default setting, 1200 Baud, is rather slow, let's increase the transfer rate to 4800 Baud.

Since the communication is done through the TERMINAL A on the SACO 148D4, type BAUDRATE A and <Enter>.
The result is: 1200:2:1:1:1
where 1200 is the baudrate, 2 means parity, (2 = no parity), 1, first from left, is the amount of stop bits, the 1 second means Line Feed and the 1 third means echo function, (1 = on).

Change baudrate by writing 4800:2:1:1:1:1
Save by STORE_BAUDRATE

Exit by the <F1> button.
CHAPTER 3

3.2.6 SMS-BASE colour settings

The SMS-BASE colour setting utility is a routine which enables the user to choose the colours of the displays generated by the SMS-BASE main program. The new settings are saved in a file called 'COLOR.CNF'. SMS-BASE uses the default colour settings, if the 'COLOR.CNF' file does not exist.

The SMS-BASE colour setting utility is a routine which enables the user to choose the colours of the displays generated by the SMS-BASE main program. The new settings are saved in a file called 'COLOR.CNF'. SMS-BASE uses the default colour settings, if the 'COLOR.CNF' file does not exist.

Table: SMS-BASE colour setting utility

<table>
<thead>
<tr>
<th>Not selected</th>
<th>Selected</th>
</tr>
</thead>
<tbody>
<tr>
<td>Selection menu</td>
<td>SMS BASE software colour setting</td>
</tr>
<tr>
<td>Normal text</td>
<td>Background Borders</td>
</tr>
<tr>
<td>Highlighted text</td>
<td>Text</td>
</tr>
<tr>
<td>Inversed text</td>
<td>Name</td>
</tr>
<tr>
<td>Ask window</td>
<td>Arrow keys - Move, Space - Change colour</td>
</tr>
<tr>
<td>Show window</td>
<td>Enter - Set colours, Esc - Exit without changing</td>
</tr>
<tr>
<td>Information text</td>
<td></td>
</tr>
</tbody>
</table>

Fig. 3.2.6.A The possibilities of changing colours are possible for different functions

The new colour settings are saved, if the routine is ended by pressing <ENTER>. Pressing <ESC> closes the colour setting routine without saving the changes. Your new settings are taken into use by next SMS-BASE start-up.

3.2.7 View readme

This menu item displays a short presentation of SMS-BASE and information for connecting the relays to the PC.

3.2.7.1 Introduction

The introduction in 'Utilities'/View readme' gives brief information about SMS-BASE.

3.2.7.2 Connecting the modem

This Section deals essentially with the connection of a PC to a telephone modem. Information is also included, however, on the direct connection of the PC to the SPA loop via an optical fibre modem when visiting the station as well as on the normal connection in the station between the optical fibre modem and the telephone modem.

The information contained in the following screen is displayed by SMS-BASE upon selecting 'Utilities' from the main menu followed by 'View readme'.
The connections for signal ground 'SG', transmitted data 'TXD' and received data 'RXD' must always be used. The remaining connections for status and control signals and their utilisation depend on the make of modem. Some modems require the status and control signals to be short-circuited in pairs if they are not used. This is indicated with the symbol 'Short'.

The cable between the telephone modem and the optical fibre modem has to be crossed, by connecting pins 2 and 3 at the telephone modem end with pins 3 and 2 respectively at the optical fibre modem end.

The last block of connection information concerns linking the PC by an RS-232 cable to a SPAC control module.
3.3 SMS-BASE ABOUT MENU

The COMM.PARAMETER menu in SMS-BASE versions 1.0 and 1.1 is replaced by the ABOUT window. The COMM PARAMETER menu is replaced with the communication parameter setting function in 'UTILITIES'/Alter application structure'/<C>.

The ABOUT window informs the about program version, user license, date and time in the PC etc. The PC date and time can be used for calibrating modules in the SMS 010 system.

3.4 SMS-BASE SETUP MENU

Normally neither screen nor printer settings need to be changed before the user has familiarised himself with SMS-BASE. The display adapter automatically senses and adjusts to the monitor in use and the printer setting is suitable for all printers similar to the IBM Proprinter and prints 62 lines per page.

The screen and the printer setting function can be selected from the main menu and the corresponding screen is as follows:

Move the bar to 'Set-up' using the <arrow> keys and select by pressing <Enter>. The display changes to that of Fig. 2.3.A showing a sub-menu with the parameters available for setting.

3.4.1 Display adapter

The type of monitor is set by selecting Display adapter which presents the following alternatives:

<table>
<thead>
<tr>
<th>NUMBER</th>
<th>ALTERNATIVE</th>
<th>COMMENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Hercules</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Toshiba T3100</td>
<td>and T2100</td>
</tr>
<tr>
<td>2</td>
<td>AT&amp;T</td>
<td>and Olivetti</td>
</tr>
<tr>
<td>3</td>
<td>MikroMikko 3</td>
<td>Nokia</td>
</tr>
<tr>
<td>4</td>
<td>Wyse GDA</td>
<td>modified Hercules</td>
</tr>
<tr>
<td>5</td>
<td>EGA</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>VGA</td>
<td></td>
</tr>
<tr>
<td>999</td>
<td>Auto detect</td>
<td>default setting</td>
</tr>
<tr>
<td>-1</td>
<td>other</td>
<td></td>
</tr>
</tbody>
</table>

Select an alternative, confirm the selection, save the result and restart SMS-BASE so that the new alternative becomes effective.
If the ultimate screen to be used is not available initially, the screen type number can be entered later using an editor by changing the number on the line ‘DISPLAY_TYPE = 999’ (‘Auto-detect’ is the default setting) in the file called GLOBAL.CNF located in the SMS-BASE directory.

The screen of a PS/2 type PC may need to be set to the standard text mode with 80 characters per line by entering the following command at the DOS prompt before starting SMS-BASE:

- "mode co80"

### 3.4.2 Printer port

Selecting ‘Printer port’ enables the port for the parallel printer to be changed. The default setting is ‘LPT1’.

### 3.4.3 Printer form lines

The number of lines per page is set by selecting ‘Printer form lines’. The default setting is 62 lines per page. A4 size paper has space for a total of 70 lines, 11 inch paper 66 lines and 12 inch paper 72 lines.

A printer with a serial interface may be used, if the PC has a serial port in addition to the one used for the modem. In this case, the output is directed firstly to the parallel port LPT1 and then redirected to a serial port, e.g. COM1 using the following two DOS commands for a printer operating at 9600 Baud:

- "mode com1:9600,n,8,1,p"
- "mode lpt1:=com1:"

These commands can be added to the AUTOEXEC.BAT file so that the redirection is automatically made every time the PC is switched on.

### 3.4.4 Appl. struct. path

‘Application structure path’ defines the directory for the application structure files (Section 3.2.3.13). When a new structure is created, the path where it is to be stored is entered by selecting this menu item. Where several structures are available on your hard disk, this menu item is selected to change the path to an existing structure. Always remember to save the new set-up (Section 3.4.6). Another possibility is to use the S=AS Handling in ‘UTILITIES’/’Alter application structure’, which also performs the same operations.

### 3.4.5 Save setup

This saves the changes made using the ‘Set-up’ menu. All the settings are stored in the file ‘GLOBAL.CNF’.
4 USING A PHONE MODEM

A telephone modem on a SPA loop can transfer the master function to a PC connected to another modem. This set-up is used for remote reading and setting.

Once the modem at the station end has been called and the link established between them, SMS-BASE data can be transferred between the devices at the two ends of the modem link. Furthermore is a message ‘MODEM ON-LINE’ glued on the SMS-BASE screen until a hang-up is done. The MODEM ON-LINE message can only be removed by the hang-up procedure, i.e. has also to be done when the connection is discontinued due to reasons apart from the hang-up described in this section. Programs like REPORT and DR-COM are automatically dialling and hanging up, without concern of the user.

The connections between a PC and a telephone modem are dealt with in Section 3.2.7 and the set-up for testing the link in Appendix C as well as an configuration example in C.5.

How to enter and change telephone numbers, serial port and the other serial communication parameters is presented in Chapter 3.

4.1 Calling

Providing the telephone number was entered using the 'Alter application structure' function, dialling and co-ordination of the modems is automatically controlled by SMS-BASE.

In Brief:

The calling procedure using telephone modems is:

1. Select 'SELECT' from the SMS-BASE main menu.
2. Select a 'Station'.
3. Select 'MODEM PhNo:...' or press the modem function key at the 'Object/Bay' level.
4. Confirm '- Handle the modem' if 'MODEM PhNo:...' was used.
5. Visually check the communication data and execute the call by pressing <D>.
6. Observe that the link is operating correctly and press <Esc> to exit.
7. Press <Esc> again to return to the station level.
8. Select 'Object/Bay' and 'Unit' in the usual way.

The following assumes that the modems are correctly connected and that the telephone number has been entered according to Chapter 3.

3. Select 'MODEM PhNo:...' or press the modem function key at the 'Object/Bay' level.

- The sub-menu at the 'Object/Bay' level appears as follows:

   - Select Object/Bay
   - B25 Grundvåg 1
   - B25 Sundom 1
   - MODEM PhNo: 021-12 34 56

- Select 'MODEM PhNo:...' for which the example telephone number has now been changed to the effective one for the application.

- Select 'MODEM' by pressing the corresponding function key. The desired telephone number is entered in the 'Edit communication parameters' window after selecting 'Alter application structure'.

   Note! If a telephone number is entered in according to both methods for the same station, the one entered for the modem function key takes priority!
4. Confirm 'Handle the modem'.

- After completing the selection at the 'Object/Bay' level, the next two levels are omitted since they only contain a single option and both are displayed in the lower part of the screen.

- The following window now appears:

```
Select function
Handle the modem
```

- Press <Enter> to confirm.

5. Visually check the communication data and execute the call by pressing <D>.

- The following screen for dialling appears upon pressing <D>:

```
**SMSSBASE**  **MODEM**  **MODEM1**  **TELEPHONE**  **MODEM HANDLING**  **- DIAL**
- NORMALLY, JUST DIAL

Communication settings from application structure
Communication port COM1  Baudrate  9600 bit/s

Modem Settings - Present values
PhNo= 021-12 34 56
Init- % RO V0

Modem Settings - New values
PhNo= 021-12 34 56
Init- % RO V0

Info: Normally, just Dial.
   Change PhNo or Modem Initiation if necessary.
   Version: MODERM2 V2.1

Station:  Vikom 1
Obj/Bay:  MODERM PhNo: 021-12 34 56
Unit:  MODERM
MoG/Part:  MODEM MODERM2 [1]

DIAL  Enter  ESC

Fig. 4.1.A  The modem dialling screen
```

- Providing the communication data is correct, dialling commences as soon as <D> is pressed.
- The four communication parameters are displayed:

<table>
<thead>
<tr>
<th>QUANTITY</th>
<th>STATUS</th>
</tr>
</thead>
<tbody>
<tr>
<td>'Communication port'</td>
<td>For information. Changed via 'UTILITIES'/ 'Alter application structure'</td>
</tr>
<tr>
<td>'Baud rate'</td>
<td>For information. Changed via 'UTILITIES' / 'Alter application structure'</td>
</tr>
<tr>
<td>'PhNo'</td>
<td>Make a temporary change here.</td>
</tr>
<tr>
<td>'Init'</td>
<td>Make a permanent change here.</td>
</tr>
</tbody>
</table>

The default setting of the 'Init' parameter includes the initialisation commands that most modems need in order to work with SMS-BASE. The commands are transferred as 'AT commands' to the modem before the telephone number is dialled. Refer to Appendix C for further information.

To change the value of 'PhNo' or 'Init', select the parameter using the <up/down arrow> keys. Open the edit field by pressing <Enter>, enter the new text and terminate by pressing <Enter>.

If the dialling procedure is unsuccessful, a corresponding message appears in the lower part of the screen. Press <Esc> to return to the menu and diagnose the error according to Appendix C.
CHAPTER 4

6. Observe that the link is operating correctly and press <Esc> to exit.

- The following screen appears providing the modems successfully establish contact:

![Image](image.png)

Fig. 4.1.B The hang-up screen for confirming that the modem link is intact

7. Press <Esc> again to return to the station level.

- Return to the ‘Station’ level to continue using SMS-BASE for acquiring, viewing and transferring data.

Note!
A successful connection is displayed by ‘MODEM ON-LINE’ on the SMS-BASE screen. This can be removed only by hang-up.

4.2 Hanging up

The procedure for terminating modem communication is simple and should be executed to keep telephone costs down. A modem link is discontinued in much the same way as it was established.

**In Brief:**

The procedure for terminating modem communication is:

1. Select ‘Select’ from the SMS-BASE main menu.
2. Select a ‘Station’.
3. Select ‘MODEM PhNo:...’ or press the modem function key on the ‘Object/Bay’ level.
4. Confirm ‘- Handle the modem’.
5. Hang up and terminate by pressing <H>.
6. A successful hang-up operation is confirmed by ‘- Handle the modem’.
7. Press <Esc> to return to the ‘Station’ level.

The screen of Fig. 4.1.B appears upon executing operation 5.

An error message is displayed if the hang-up procedure is unsuccessful. In this case, hang up manually, i.e. by pressing the button on the modem or physically disconnecting it from the telephone line. Press <Esc> to return to the ‘Station’ level. SMS-BASE now assumes that hanging up has taken place manually.
4.3 Warning of wrong connection or an unfinished call

The following errors can occur when using a modem:

- SMS-BASE can be closed without previously hanging up.
- The modem is connected to a station other than the one selected.

Error messages are displayed in both these instances, but it is possible to ignore them. SMS-BASE only gives a warning, because the user may wish to deliberately exit SMS-BASE but maintain the link to the remote station for some other purpose using different communication software.

Data acquired earlier on from one station may be viewed although the modem is connected on-line to another station.

The SMS-BASE error screen of Fig. 4.3.A shows the station connected originally and the one currently selected.

![Error Screen](image)

Fig. 4.3.A Error message warning that the modem is connected to a station other than the one called.
5  CHANGING THE PASSWORD

A password has to be entered before SMS-BASE permits any operation which changes the settings of a protection unit.

The password is the same for all protection devices in any one station according to the SMS 010 application structure. The password can be viewed or changed individually for each station by pressing button key F5 as illustrated in Fig.5.A.

![Fig.5.A. Checking or changing a password using button F5.](image)

SMS-BASE password management starts with lines 1 and 2 in Fig. 5.B. The following must be observed:

- The password is entered in an 8 character field. In practice, however, a password only ever comprises precisely 3 digits. The extra character places are only intended to mislead unauthorised operators.

- A new station in the application structure always starts with the password '000' (three zeros). This is the case shown in the following example.

![Fig.5.B Entering a password](image)

The procedure is as follows:

- Enter "000" and press <Enter>.
- Wait 5 s while SMS-BASE checks the password.
- The display now changes to that of Fig.5.C.

The standard procedure now requires <Enter> to be pressed to continue to the main operation.
- The password can be changed for any of the protection devices in a station’s application structure. To become effective, the password must be changed before the terminating this operation. Press <A> and the PC displays:

```
A State the new password 4
```

Figure 5.C Giving new password.

Enter precisely three digits for the new password and confirm by pressing <Enter>. 
6 BACKING UP DATA

All files changed in the course of an SMS-BASE session need to be backed up. The corresponding files are found, for example, in the directory 'DATA-EX' or the directory you created for your application structure. It is also advisable for safety reasons to make back-up copies of the SMS-BASE program disks.

Copy the program disks by entering "diskcopy a: a:" at the DOS prompt and following the instructions given on the screen.

The application structure stored on the hard disk changes as well as the data in the files when new information is acquired from the relays. All changed data should be backed up regularly by copying the files onto floppy disks using the appropriate DOS commands.

The command for backing up data from the hard disk to floppies in drive A: is:

"backup c:\sms\data-ex\*.\* a: /s /f"

Follow the instructions on the screen. The switches '/s' and '/f' at the end of the command line above mean respectively that all sub-directories should be backed up as well and that any unformatted disks should be formatted first.

A 1.4 Mbytes disk has sufficient capacity for the data of approx. 25 D-type or 100 C-type relays. If there are more relays on the system, DOS 'backup' automatically prompts the user remove the full disk and insert a new one. Number the back-up disks carefully.

The corresponding DOS command to restore data from floppy disks in drive A: to the SMS-BASE directory on drive C: is:

"restore a: c:\sms\data-ex\*.\* /s".

This operation is only necessary, however, in the event of a failure of the hard disk or data being deleted by mistake.

SMS-BASE can also be used such that all working data are on floppy disks, in which case the back-ups are created by copying one floppy to another using the DOS command "diskcopy a: a:".
7  DIAGNOSING ERRORS

The correct operation of SMS-BASE relies on correctly operating hardware. Refer to the relevant instructions for diagnosing hardware errors. This User’s Guide does, however, include a simple functional test of the communication equipment in Appendix C.

The SMS-BASE software will not operate properly, if it is installed or used incorrectly, or if the conditions for proper communication are not fulfilled.

Check the installation with reference to Chapter 2 and Appendix A.

How to use SMS-BASE correctly is presented in Chapter 3. Where an error does occur, the type and probable cause are displayed in the lower part of the screen. Should an error message indicate that a file is missing, then the application structure is probably not defined on the level below. In this case, first complete the structure using the ‘Alter application structure’ facility.

Make sure that a printer is connected and on line before attempting to print anything. Many PC’s stop and wait for the printer to respond and have to be restarted if the printer does not.

If the communication conditions are not fulfilled, data will not be transferred. After a number of unsuccessful attempts to transfer a variable, the program instructs the user to press <R> to retry or <Q> to quit. Usually it is better to press <Q>, cancel the call and try to get through again on a better line.

7.1  Error messages

Those error messages requiring some explanation are presented in this Section. A number of other messages such as ‘Waiting...’, ‘Solving...’, ‘Reading...’ etc., will be observed, but all of them are self-explanatory.

Error messages normally have to be confirmed by pressing <Enter>, after which the user is returned to the activity immediately before the one causing the error.

SELECTION OF ORGANISATION, STATION, OBJECT/BAY, UNIT AND SUB-PROGRAMS:

'Application structure is not valid!'
The application structure has not been correctly defined or the correct procedure for creating it was not followed. Try to correct the application structure by selecting ‘Alter Application structure’ from the main menu. The message could also indicate that there is insufficient main memory available or that it is defective.

'No sub-level for level "xx" found'
An attempt was made to select an item below level "xx", which is not fully defined. Try to complete the definition using ‘Alter Application structure’.

'No SPIN configuration files found!'
The file containing the serial communication settings is missing. Try entering the settings again using ‘Alter Application structure’.

'Illegal value: "The value"'
An attempt was made to enter an illegal value. Acknowledge and enter a value in the permissible range.

'Illegal input: "The string"'
An attempt was made to enter an illegal string. Acknowledge and enter a permissible string.
'Spawn failure, cannot execute program!' It was no possible to start a sub-routine. Acknowledge and make a different selection. Check your SMS-BASE installation.

'Spawn failure, cannot execute program!' It was no possible to start a sub-routine. Acknowledge and make a different selection.

'Program aborted!' The execution of a sub-routine aborted due to errors in the data or the program. Acknowledge and make a different selection.

'Program aborted because of user break!' The user has interrupted a sub-routine by pressing <Ctrl>+<C> or <Ctrl>+<Break>. Acknowledge.

'Out of disk space!' The hard disk or floppy is full. Acknowledge and create more space by deleting unnecessary files.

'Out of disk space!' The hard disk or floppy is full. Acknowledge and create more space by deleting unnecessary files.

'Writing of temporary file failed!' The hard disk or floppy is full. Acknowledge and create more space by deleting unnecessary files.

'Writing of temporary file failed!' The hard disk or floppy is full. Acknowledge and create more space by deleting unnecessary files.

'Not enough memory!' The main memory is full. Acknowledge and close other programs that occupy main memory space while SMS-BASE is running, e.g. resident programs and network drivers.

'Not enough memory!' The main memory is full. Acknowledge and close other programs that occupy main memory space while SMS-BASE is running, e.g. resident programs and network drivers.

COMMUNICATION - DATA TRANSFER TO AND FROM THE RELAY:

'Time-out while waiting for the slave to answer'
'Retry or quit (R/Q) ?'
No reply was received from a slave within the permissible time. Press <R> to try once more. If this is unsuccessful, terminate by pressing <Q> and check the communication link using the terminal emulator. It is possible that the address is incorrect.

COMMUNICATION - MODEM FUNCTIONS:

'Error the modem could not be reset. Press any key'
Probably the modem is not connected or has its power supply is switched off and consequently can not reply to the reset command, which is at the beginning of all modem operations.

'No dial tone detected. Press any key'
The modem did not receive a dialling tone. Probably the line is not connected. Acknowledge and check the connection. It is possible that the modem is unsuitable for the exchange or for use in the particular country. The system is such that the modem does not receive a dialling tone. Configure the modem not to wait for a dialling tone (ATX3). This can be done from the modem window.

'Busy signal detected. Press any key'
The line is engaged. Acknowledge and wait.

'Error in communication (modem). Press any key'
The modem at the other end does not reply. Acknowledge and check that the telephone number is correct and also correctly entered. Call the remote modem from an ordinary telephone set to check that it replies with a tone.
THE FUNCTION 'SELECT FUNCTION':

'Cannot find...' or 'No place for...' '...Spinfile path, selection, slave number or communication data'
The 'Edit buffer' functions cannot run, because basic data is missing. Exit and check the application structure using 'Alter Application structure' or 'Check application structure'.

'Wrong reply from relay. Check type and slave', 'number. Redo or exit'
The initial response from the relay is incorrect although the communication channel is operating properly. Possibly the wrong relay was called. Exit and check.
A DIRECTORIES AND FILES IN SMS-BASE

The SMS-BASE software product is delivered on two 3½” high-density (1.44 Mbytes) floppy disks. The disks are numbered and named as follows:

<table>
<thead>
<tr>
<th>NUMBER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>SMS-BASE program disk</td>
</tr>
<tr>
<td>2</td>
<td>Example application structure disk</td>
</tr>
</tbody>
</table>

Both floppy disks have the same directory structure or a part thereof and the same structure is retained when the system is installed. The files on the disks are compressed using the program ‘pkzip’ to minimise the number of installation disks. The contents of the disks are presented below as they appear when the disks are inserted into drive A: and are viewed using the program ‘pkunzip’:

A:\ Root directory of the system
SUPPORT\ SMS-BASE auxiliary programs
PACKAGES\ Basic files for starting the system
MODULES\ Descriptions
OTHER\ Descriptions of non-SPA modules and relays
MODEM\ Modem description
DATA-EX\ Application structure data

A.1 Disk 1

Disk 1 contains:

A:\
INSTALL.EXE Installation program
INSTALL.LAN Installation language texts
MUST.ZIP Archived files
PKUNZIP.EXE File expansion program
README Disk information

The file 'MUST.ZIP' contains:

RESOURCE.EXE Measure available memory
UTIL.EXE Terminal emulator
SMSBASE.EXE SMS-BASE program main menu
SMSLOGO.EXE SMS-BASE logo
AS.EXE Application structure generator
ASMAIN.EXE Application structure generator
SPA.EXE SPA on-line communication
STREE.EXE Check application structure
SMSME.EXE View readme
COLORSET.EXE SMS-BASE colour setting
ASLAN.HLP Help texts for AS Handling
APL.CNF List of application structures
PHN-LIST.CNF Phone-number list
COLOR.CNF Present colour settings
SMSBASE.ICO SMS-BASE icon used by Windows
SMSBASE.GRP SMS-BASE group file used by Windows
SMSBASE.PIF SMS-BASE pif file used by Windows
GLOBAL.CNF System and data path and screen and printer settings
A.2 Disk 2

Disk 2 contains an example application structure. The directory names consist of one letter and two digits. The SMS-BASE user never sees the directory names but texts entered to identify each level of the structure.

The directory structure is as follows, nn representing a consecutive number:

A:\DATA-EX\ Root directory for data
   Onnn\ 'Organisations' in the application structure
   Snnn\ 'Stations' in and 'Organisation'
   Onnn\ 'Object/Bays' in a 'Station'
   Unnn\ 'Units' in 'Object/Bays'
   Mnnn\ 'Modules/Parts' in a 'Unit'

B:\RELEASE Latest news
RELEASE How to connect serial ports and introduction
MAIN-KEY.SUP Function key definitions
\SUPPORT\ UTILS.SUP Configuration of 'UTILITIES' menu
SPIN.CNF Default communication values
PHN-STN.DAT Default communication values
SPACOM.CNF Default values for new application structure
PACKAGE.PCK Default transducer ratio values
APPENDIX A

Disk 2 contains the following:

<table>
<thead>
<tr>
<th>File/Folder</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>INSTALL.EXE</td>
<td>Installation program</td>
</tr>
<tr>
<td>INSTALL.LAN</td>
<td>Installation language texts</td>
</tr>
<tr>
<td>README</td>
<td>Disk information</td>
</tr>
<tr>
<td>SPACOM.CNF</td>
<td>Texts for selection heading</td>
</tr>
<tr>
<td>CHOICES.DSC</td>
<td>Texts for the ‘Organisation’ menu</td>
</tr>
<tr>
<td>Onnn\CHOICES.DSC</td>
<td>Texts for the ‘Station’ menu</td>
</tr>
<tr>
<td>Snnn\CHOICES.DSC</td>
<td>Texts for the ‘Object/Bay’ menu</td>
</tr>
<tr>
<td>SPIN.CNF</td>
<td>Communication settings</td>
</tr>
<tr>
<td>PHN-STN.DAT</td>
<td>Communication settings</td>
</tr>
<tr>
<td>Onnn\CHOICES.DSC</td>
<td>Texts for the ‘Unit’ menu</td>
</tr>
<tr>
<td>Unnn\CHOICES.DSC</td>
<td>‘Modules/Parts’ texts</td>
</tr>
<tr>
<td>PACKAGE.PCK</td>
<td>Transducer data, i.e. c.t. and v.t. ratios</td>
</tr>
<tr>
<td>Mnnn\VALUES.VAL</td>
<td>Settings created using the editor</td>
</tr>
<tr>
<td>VALUES.INS</td>
<td>Acquired values of variables</td>
</tr>
<tr>
<td>SLAVE.DAT</td>
<td>Slave number, i.e. node number (existing and new number)</td>
</tr>
</tbody>
</table>

A.3 The hard disk installation

The system installed on the hard disk has the same directory structure as the floppy disks. The default directory created for the system is:

C:\SMS\BASE.

The following files are not included in the floppy disks, but can appear on the hard disk:

<table>
<thead>
<tr>
<th>File/Folder</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>C:\SMS\BASE\MDMSTUS.TMP</td>
<td>Modem line or No. status.</td>
</tr>
<tr>
<td>C:\SMS\BASE\DEFAULT.REP</td>
<td>Appears when modem on-line.</td>
</tr>
<tr>
<td>C:\SMS\BASE\SUPPORT\MODEM.PAR</td>
<td>Report heading, ‘Org.’, ‘User’</td>
</tr>
<tr>
<td>C:.SMS\BASE\DATA-EX\SCANNED.SLV</td>
<td>Present modem setting, i.e. as copied from MODEM2.PAR.</td>
</tr>
<tr>
<td>C:.SMS\BASE\DATA-EX\ERROR.TXT</td>
<td>SPTOED error text</td>
</tr>
<tr>
<td>C:.SMS\BASE\DATA-EX\PS.$$</td>
<td>SPTOED print temp. file</td>
</tr>
<tr>
<td>C:.SMS\BASE\DATA-EX\SEND.ERR</td>
<td>SEND error text</td>
</tr>
<tr>
<td>C:.SMS\BASE\DATA-EX\RECEIVE.ERR</td>
<td>RECEIVE error text</td>
</tr>
<tr>
<td>C:.SMS\BASE\DATA-EX\SnnnPSW.DAT</td>
<td>Encrypted password/station</td>
</tr>
<tr>
<td>C:.SMS\BASE\DATA-EX\Mnnn951001_0.SRP</td>
<td>Report file on one data set made 95-10-01</td>
</tr>
</tbody>
</table>
B SMS-BASE SPECIFICATION

The SMS-BASE system requirements and the typical data of SMS-BASE functions are given in this Appendix.

B.1 Hardware requirements

The hardware requirements which have to be fulfilled for SMS-BASE to operate correctly are:

- **Relay end**
  - PYRAMID unit with an SPA communication option.
  - Optical fibre SPA loop with an optical fibre modem or a direct connection to the relay from the PC’s serial port.

- **Remote communication**
  - A leased line equipped with modems or a telephone modem and automatic answering function at the station.
  - A public telephone network according to CCITT.
  - A telephone modem which accepts "AT" commands at the PC end.

- **PC**
  - PC IBM-AT or better but 100% compatible, or a 80286 processor at minimum. No special configuration. No special accessories for controlling serial or parallel ports.
  - Operating system DOS 3.3 or higher
  - Main memory 500 Kbytes available
  - Hard disk 2,4 Mbytes for SMS-BASE shell. About 1 Mbytes of additional free disk space is recommended.
  - 1,5 Mbytes for a "normal" application structure, kBytes for the example structure
  - Serial port 1 for the modem and 1 for the mouse if using the SPTOED logic editor with SPACOM control modules, COM1 or COM2
  - Parallel port 1 for the printer, LPT1
  - Floppy drive 1 3½” 1,44 MByte
  - Code page 437, 860, 863 or 865

B.2 Application structure and data memories

The application structure represents the actual physical structure of stations, bays and relays and enables the user to select relay and data memories of interest.

- **Application structure**
  - No. of relays Unlimited
  - No. of structures Unlimited
  - No. of selections per level Unlimited

- **Storage of relay data on disk**
  - Storage of historic data Possible
  - Storage of prepared settings Possible
B.3 Communication

- **Communication settings**
  - Serial port: COM1, COM2, COM3, COM4
  - Baud rate: 300, 1200, 2400, 4800, 9600, 19200, 38400 Baud
  - Protocol: SPA, SRIO, (SACO 100M prot. is called SRIO in SMS-BASE)

- **Data transmission time at 2400 baud**
  - Remote acquisition: 0.15 s/variable
  - Remote send: 0.3 s/variable

- **Time to establish a modem link**: 10 s typical
C  SETTING AND TESTING THE PHONE MODEM

The telephone modems form an important link in the communication chain between the relay and SMS-BASE. For this reason, suggested modem settings and a step by step procedure for testing the communication link are included here.

C.1 Setting the phone modem

Modems that accept "AT" commands have an additional 'front panel', which is made visible by connecting the modem to a terminal. The micro-controller in the modem acts as the host computer and provides a command for each variable on the front panel displayed on the terminal. A modem is always in the "command mode" when it is switched on, which usually ends with the dialling of a telephone number. Once the modems at both ends have established contact they switch to the "data mode".

In the data mode, any string of characters must be permitted to pass without the modem switching back to the command mode. There is, however, one very special string that does switch the modem to the command mode, i.e. "+++", followed by a pause of at least one second. The call can then be terminated using the command "ATH" for 'Hang up'.

Typical modems have up to two hundred possible settings, but fortunately the defaults are generally good enough for the relatively simple purposes of SMS-BASE.

A command always commences with "AT" followed by characters defining the type of command and in some cases a digit for setting a value. The modem confirms the acceptance of a command by returning 'OK' or '0', respectively the non-acceptance of a command by returning 'ERROR' or '4'.

Some useful modem commands when using SMS-BASE are compiled in Table C.1.A.

The usual settings are marked by 'F' in the table and those needed by SMS-BASE by 'R'. Commands in brackets do not work on all types of modems.

Automatic modem management by SMS-BASE always has "ATZ" at the beginning. Commands are then transmitted for the cases where "R" and "F" differ, i.e. "ATV0" and "ATE0". These commands are therefore included in the SMS-BASE modem 'Init string'. The prefix 'AT' is not included, because it is automatically added.

Some more advanced modems do not load the default settings in response to the 'ATZ' command, but basic settings defined with the aid of 'AT&W' by the user. In such cases, 'AT&F' may be required in order to install the works settings to achieve a defined starting point.
APPENDIX C

COMMAND EXPLANATION

ATZ  'Reset' - A call has been terminated and the basic user settings (or the default settings if there are no user settings) are loaded and become the current settings. Baud rate and parity are automatically set to match those of the port that sent the command.

( AT&W  Stores the current settings as the user’s default settings)

ATV0  R  Status information in code, i.e. 0...4

ATV1  F  Status information in text, e.g. 'OK'

<table>
<thead>
<tr>
<th>Code</th>
<th>Text</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>OK</td>
</tr>
<tr>
<td>1</td>
<td>Connect</td>
</tr>
<tr>
<td>4</td>
<td>Error</td>
</tr>
</tbody>
</table>

ATX4  F,R  Alternative 4 gives normal status information regarding the call.

ATX3  Configure your modem not to wait for a dial tone.

ATE0  R  No echo in response to a character sent

ATE1  F  Echo in response to a character sent

( AT&S  List current settings)

Table C.1.A  Some "AT" commands for modem settings

*  SMS-BASE is not capable of handling X4 mode, use X0 instead

Two versions of Init string’ may be required according to the modem settings. The characters in brackets are automatically added by SMS-BASE:

- "(ATZ) (AT)E0 (AT)V0"
  - Useful in most cases where the user has not stored special default settings.
  - Useful where the user has stored special default settings in the modem, which particularly favourable for SMS-BASE.

- "(ATZ) (AT)&F (AT)E0 (AT)V0"
  - Useful where the user of another application has stored default settings in the modem, which are unfavourable for SMS-BASE.
C.2 Testing the phone modem

The connection between the personal computer and the modem can be tested with the terminal emulator.

In Brief:

The procedure for testing a modem is:

1. Connect the modem to the PC as described in Chapter 3.2.7.
2. Do not connect the modem to the telephone line.
3. Turn on the power supply.
4. Start the terminal emulator.
   - Select the same communication settings as those to be used in operation.
   - Select ‘Char mode’.
5. Send commands to the modem and observe the replies. Providing a reply is received, the connection is in order.

5. Send commands to the modem and observe the replies

- See if there is a reply to an "AT" query:

<table>
<thead>
<tr>
<th>NORMAL REPLY</th>
<th>VERSION WITH 'ATV0'</th>
<th>VERSION WITH 'ATE0'</th>
</tr>
</thead>
<tbody>
<tr>
<td>U - &gt;&gt; AT followed by &lt;Enter&gt;</td>
<td>AT</td>
<td>(no echo)</td>
</tr>
<tr>
<td>M - AT</td>
<td>0</td>
<td>OK</td>
</tr>
</tbody>
</table>

"U" stands for User and "M" for modem.

The first reply is just the echo of the command, providing the modem is set to return echoes using ‘ATV1’, otherwise this response is omitted. The reply ‘OK’ or ‘0’ confirms that the modem has understood the command.

- The link has now been established between the PC and the modem.

C.3 Dialling and hanging up manually

Once the PC and the modem are communicating, the next step is to test the modem to modem connection. If this is not possible, a considerable part of the system can be tested by dialling your own number and receiving the busy signal.

In Brief:

The procedure for testing the connection to another modem is:

1. Perform C.2.
   - Do not close the terminal emulator.
2. Call another modem
   - Observe the confirmation that the modems are connected.
3. Hang up
   - Observe the confirmation that the modems are disconnected.

A separate group of "AT" commands control the telephone line (Table C.3.B).
APPENDIX C

COMMAND EXPLANATION

ATD 'number'  Dial 'number'
- T  Tone signalling
- P  Pulse signalling. The latest 'D' or 'T' selections are valid.
'number'  Digits and control characters, e.g. 'W' - wait for dial tone
ATH  'Hang up'

Table C.3.B  Some "AT" commands for telephone line control

It is assumed in the following example that the modem has the most common setting immediately after switching on with the status information in text mode and echo turned on. Otherwise send the commands "ATV1" and "ATE1" to the modem in order to obtain the same dialogue.

2. Call another modem  USE A GENUINE PHONE NUMBER !!

- Connect the modem to the telephone line.
- Set the terminal emulator to the character mode.
- Wait for the dialling tone and call (123)456-7890 using tone signalling for purposes of this dialogue:

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>U - &gt;&gt;</td>
<td>ATDT (123)456-7890 followed by &lt;Enter&gt;</td>
</tr>
<tr>
<td>M -</td>
<td>ATDT (123)456-7890</td>
</tr>
<tr>
<td>M -</td>
<td>'1' (= CONNECT)</td>
</tr>
</tbody>
</table>

- Errors will produce different responses which are useful for tracing the cause.
- At least call your own number to perform a minimum test of the telephony parts of the modem:

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>U - &gt;&gt;</td>
<td>ATDT &quot;own number&quot; followed by &lt;Enter&gt;</td>
</tr>
<tr>
<td>M -</td>
<td>ATDT &quot;own number&quot;</td>
</tr>
<tr>
<td>M -</td>
<td>'4' (=Error)</td>
</tr>
</tbody>
</table>
3. **Hang up**

- The data channel is established as soon as the modems have co-ordinated their functions. In this status, the modem no longer reacts to "AT" commands. The dialogue below returns the user to the command mode.

- Ensure that the terminal emulator is in character mode.

```plaintext
U - >> +++ 1 sec. pause before and after +++
M - +++
M - '1' (=OK) indicates that the command mode is established.
```

- Hang up with:

```plaintext
U - >> ATH followed by <Enter> 'Hang up'
M - ATH
M - '1' (=OK) Hang-up successfully executed.
```

**C.4 Testing communication with the relay**

Once the connection to the modem in a station is established, the next step in the test procedure is to communicate with a relay on an SPA loop. If this step can be executed successfully, there are probably no communication channel problems.

**In Brief:**

The procedure for testing communication with a relay on an SPA loop is:

1. Perform C.2 and C.3.
   - Stay on line and in the data mode.
2. Change the terminal emulator to string mode.
3. Request the type designation from a relay on the loop.
4. Change the terminal emulator back to 'Char mode'.
5. Hang up as instructed in C.3.
   - Observe that the execution of the hanging up operation is confirmed.

Assuming the relay module has slave address 31 on the SPA loop and that its type is SPAJ 140 C with a module SPCJ 4D29.

2. **Change the terminal emulator to string mode.**

- A slave on an SPA loop requires that all the characters in a telegram be received in one continuous string. The terminal emulator is therefore not permitted to send the characters one by one as entered by the user.

- Press <F3> and observe that the terminal emulator changes back to 'String mode'. This causes the whole string of characters to be sent upon pressing <Enter> at the end of entering them.
- Set the modem to the non-echo mode. One echo is still received due to the original signal travelling all the way round the SPA loop:

| U - >> | ATE0 followed by <Enter> | Turn off the modem echo. |
| M -     | ATE0                      | One last echo.        |
| M - OK  |                          | Order acknowledged.   |

3. Request the type designation from a relay on the loop.

- The PC functions via the terminal emulator and the modems as a master on the SPA loop. The following dialogue can be used to request the type designation from a relay on the loop:

<table>
<thead>
<tr>
<th>'Master' - &gt;&gt;</th>
<th>&gt;3lROF:XX followed by &lt;Enter&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>'Master'</td>
<td>31 - SPA node 31</td>
</tr>
<tr>
<td>'Master'</td>
<td>R - Read</td>
</tr>
<tr>
<td>'Master'</td>
<td>0F - SPA ident. of type</td>
</tr>
<tr>
<td>'Master'</td>
<td>: - delimiter</td>
</tr>
<tr>
<td>'Master'</td>
<td>XX - checksum unknown</td>
</tr>
</tbody>
</table>

| 'Echo on SPA loop' - >> | >3lROF:XX | The echo around the loop |
| 'Slave' - | <3ID:SPCJ 4029:6A | The module reply |

- Make at least two attempts before deciding on a failure. Sometimes there are false telegrams in the reception queue of the slave which have to be pushed out of the way by genuine telegrams.

- Communication around the loop is working well if the message is echoed and response given, but check the following items if the module for some weird reason would not reply:

  - Was the message correctly entered?
  - Is the relay set to the baud rate in use?
  - Is the relay set to the parity in use?
  - Is the relay set to the node number in use?

Note that the SPA loop principle provides an additional safety feature in that a telegram transmitted by the master must always be received by the master again after travelling around the loop. This cannot happen if the loop is interrupted and is noticed right away.

4. Change the terminal emulator back to 'Char mode'.

- Calls are terminated by '+++’ (1 sec. pause before and after +++), which when entered have to be transmitted one at a time.
- Press <F3> to set the terminal emulator to 'Char mode' so that the characters are sent as soon as the corresponding key is pressed.
C.5       Modem configuration example for SMS-BASE

In this modem configuration example are two US Robotics Sportster 14400 bps fax modems (USRSP144FAX), used. One is used on the device side, (relay side), and the other one on the PC side. US Robotics is one manufacturer of modems, other modems can be used as well, but the modems in this example are available in most places all over the world.

C.5.1       Cables for connection

For connecting the modem on the PC side, the standard modem cable included in the modem package is used. On the device side it depends on the configuration used, which type of cables that should be used.

The following configurations will be dealt with in this example:

2.1.       Modem connected directly to the relay unit using

2.1a.       TTL-level signal (SPA)
2.1b.       RS 485 signal
2.1c.       RS 232 signal (e.g. SPAC/SPCR front)

2.2.       Modem connected to a fiber optic converter (e.g. SPA-ZC 22)
2.3.       Modem connected to a SACO 148, (SACO 100 data communicator)
2.4.       Modem connected to a SRI 500M/SRI 1000M

In case 2.1a should cables SPA-ZP 5A_ and SPA-ZP 10A_ be used.
In case 2.1b should cables SPA-ZP 6A_ and SPA-ZP 10A_ be used.
In case 2.1c should cables SPA-ZP 17A_ and SPA-ZP 10A_ be used.
In case 2.2 should cable SPA-ZP 10A_ be used.
In case 2.3 should cable SPA-ZP 10A_ be used.
In case 2.4 should cable SPA-ZP 9A_ be used.

C.5.2       Modem settings

Common for all modem settings, from the SMS-BASE point of view, is that the error correction facilities of the modems need to be switched off. The SPA protocol has its own error correction which consists of re-sending of messages that was not properly received.

The type of result code used, need to be taken into consideration when configuring the modems. Numerical result codes from a limited result code set need to be used for the modem on the PC side. The modem on the device side should not be allowed to display any result code at all. It has to be set in quiet mode. See the example in result code table C.5.2.3. It is also recommended not to let the modem on the device side answer on the first dialling tone, but wait until the second or third in order to assure a secure connection.

In table C.5.2.1 is a summary of the settings applied to the US Robotics modem.
## APPENDIX C

### Command Summary

<table>
<thead>
<tr>
<th>Command mode local echo</th>
<th>Code</th>
<th>Modem on PC side</th>
<th>Explanation</th>
<th>Modem on Device Side</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>E</td>
<td>0</td>
<td>Echo Off</td>
<td>0</td>
<td>Echo Off</td>
</tr>
<tr>
<td>Result code display</td>
<td>Q</td>
<td>0</td>
<td>Display result code</td>
<td>1</td>
<td>Quiet mode; no result codes</td>
</tr>
<tr>
<td>Verbal/numeric result codes</td>
<td>V</td>
<td>0</td>
<td>Numeric code</td>
<td>0</td>
<td>Numeric code</td>
</tr>
<tr>
<td>Result code set</td>
<td>X</td>
<td>0</td>
<td>See table C.5.2.3</td>
<td>0</td>
<td>See table C.5.2.3</td>
</tr>
<tr>
<td>Enable/disable ARQ result codes</td>
<td>&amp;A</td>
<td>0</td>
<td>ARQ result code disabled</td>
<td>0</td>
<td>ARQ result code disabled</td>
</tr>
<tr>
<td>Data compression enabled disabled</td>
<td>&amp;K</td>
<td>0</td>
<td>Data compression disabled</td>
<td>0</td>
<td>Data compression disabled</td>
</tr>
<tr>
<td>Normal or Error Control (ARQ) mode</td>
<td>&amp;M</td>
<td>0</td>
<td>Normal mode, error control disabled</td>
<td>0</td>
<td>Normal mode, error control disabled</td>
</tr>
<tr>
<td>Data rate, phone line interface</td>
<td>&amp;N</td>
<td>0</td>
<td>Variable rate</td>
<td>0</td>
<td>Variable rate</td>
</tr>
<tr>
<td>Data Terminal (DTR) operations</td>
<td>&amp;D</td>
<td>0</td>
<td>DTR override</td>
<td>0</td>
<td>DTR override</td>
</tr>
<tr>
<td>Dialling signals before Auto Answer</td>
<td>S0</td>
<td>0</td>
<td>No Auto Answer</td>
<td>3</td>
<td>Auto Answer on the third dialling tone</td>
</tr>
</tbody>
</table>

### Storing parameters to NVRAM

<table>
<thead>
<tr>
<th>Command mode local echo</th>
<th>Code</th>
<th>Modem on PC side</th>
<th>Explanation</th>
<th>Modem on Device Side</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Table C.5.2.1 Modem settings

Some of these settings can be made with the DIP switches on the back of the modem as follows:

<table>
<thead>
<tr>
<th>Dip Switch</th>
<th>For modem on PC side</th>
<th>For modem on Device Side</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Down</td>
<td>Down</td>
<td>DTR Signal used</td>
</tr>
<tr>
<td>2</td>
<td>Down</td>
<td>Down</td>
<td>Numeric result code</td>
</tr>
<tr>
<td>3</td>
<td>Down</td>
<td>Up</td>
<td>Display result code (Down)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Suppress result code (Up)</td>
</tr>
<tr>
<td>4</td>
<td>Down</td>
<td>Down</td>
<td>No echo, off-line commands</td>
</tr>
<tr>
<td>5</td>
<td>Down</td>
<td>Up</td>
<td>Suppress Auto answer (Down)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Auto answer on ring (Up)</td>
</tr>
<tr>
<td>6</td>
<td>Up</td>
<td>Up</td>
<td>Carrier Detect signal used</td>
</tr>
<tr>
<td>7</td>
<td>Up</td>
<td>Up</td>
<td>Load NVRAM on start-up</td>
</tr>
<tr>
<td>8</td>
<td>Down</td>
<td>Down</td>
<td>Smart mode</td>
</tr>
</tbody>
</table>

### Table C.5.2.2 Dip Switches
<table>
<thead>
<tr>
<th>Result code</th>
<th>Result code</th>
<th>X0</th>
<th>X1</th>
<th>X2</th>
<th>X3</th>
<th>X4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Numeric</td>
<td>Verbal</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0/OK</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1/CONNECT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2/RING</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3/NO CARRIER</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4/ERROR</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5/CONNECT 1200</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6/NO DIAL TONE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7/BUSY</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8/NO ANSWER</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10/CONNECT 2400</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13/CONNECT 9600</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18/CONNECT 4800</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20/CONNECT 7200</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>21/CONNECT 12000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>25/CONNECT 14400</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table C.5.2.3  Result code settings, e.g. X0 = result codes 0 to 4 are included, (used for SMS-BASE)
D USING SMS-BASE VIA A SRIO/SACO UNIT

In some applications, several systems have to be able to communicate with the relays on the same SPA loop. Often a station control system competes with SMS-BASE in attempting to establish contact with a particular relay.

A 'time multiplexer' has to be used in such circumstances which cyclically enables and disables the masters. The SRIO 1000 and SRIO 500 units permits two masters to be connected to the same SPA loop (Fig. D.1.A).

D.1 A SMS-BASE system with a SRIO/SACO

From the SMS-BASE point of view, the SRIO unit appears to be a optical fibre converter. This results in a system as shown in Fig D.1.A. The cable used between the PC running SMS-BASE and the SRIO unit is SPA-ZP 7A_.

When a SACO unit is used instead of the SRIO unit, it is not possible to connect a substation automation system to the same SPA loop. The cable used between the PC running SMS-BASE and the SACO unit is SPA-ZP 13A_.

![Diagram](image_url)

**Fig D.1.A**  Example of a SMS 010 system in parallel with a substation automation system with a SRIO data communicator used by two systems with a directly connected PC running SMS 010. From the SMS 010 point of view, the SRIO unit appears to be a opto/electrical fiber converter. The cable used between the PC running SMS 010 and the SRIO unit is SPA-ZP 7A_. When a SACO 100M unit is used instead of the SRIO, it is not possible to connect both SMS 010 and a substation automation system to the same SPA loop. The cable used between the PC running SMS-BASE and the SACO 100M unit is SPA-ZP 13A_.

A system with modems between the RS232 interfaces on PC and SRIO/SACO unit results in a system as shown in Fig. D.1.B. The interconnecting cables used are SPA-ZP 13A_ between the PC and the modem and SPA-ZP 9A_ between the modem and the SRIO unit. When using a SACO unit, the connecting cable for the modem is the one given in the connection information in section 3.2.7.2.
Fig D.1.B Example of a SMS 010 system in parallel with a substation automation system with a SRI0 data communicator used by two systems. A modem link is used between the RS-232 interfaces on the PC and the SRI0/SACO unit. The interconnecting cables used are SPA-ZP 13A_ between the PC and the modem and SPA-ZP 9A_ between the modem and the SRI0 unit. When using a SACO 100M unit, the connecting cable for the modem is the one given in the connection information in section 3.2.7.2 in this User’s Guide. Note that when a SACO 100M unit is used instead of the SRI0, it is not possible to connect both a substation automation system and SMS 010 to the same SPA loop.

D.2 SMS-BASE and a SRI0 unit

With the exception of the following operations, SMS-BASE is scarcely influenced by the introduction of an SRI0 unit:

The protocol for station communication should be set to 'SRI0', see section 3.2.3.6 Communication settings. The table below describes the recommended settings when the SRI0 or SACO unit has default settings.

<table>
<thead>
<tr>
<th></th>
<th>SRI0 500M/1000M</th>
<th>SACO 100M</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interface type:</td>
<td>Direct</td>
<td>Direct</td>
</tr>
<tr>
<td>Communication port:</td>
<td>COM1</td>
<td>COM1</td>
</tr>
<tr>
<td>Protocol:</td>
<td>SRI0</td>
<td>SRI0</td>
</tr>
<tr>
<td>Baud rate:</td>
<td>1200</td>
<td>1200</td>
</tr>
<tr>
<td>Parity:</td>
<td>No parity</td>
<td>No parity</td>
</tr>
<tr>
<td>Word length:</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>Stop bits:</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Handshake:</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>Echo handling:</td>
<td>None</td>
<td>Ignore</td>
</tr>
</tbody>
</table>

Table D.2.A Factory settings for SACO/SRI0 data communicators

Change the communication settings in SMS-BASE according to actual data communicator settings. Refer to figure 3.2.3.6.A in this User’s Guide. Change protocol from default SPA to SRI0.

The 'SRI0' protocol is simply an alternative provided by the SPA system. The differences to the SPA protocol are so small that the terminal emulator can be used as described earlier for sending and receiving SPA messages.
**D.3 Specification changes due to SRIO**

With the exception of the protocol and the data transfer times, the SMS-BASE specification is unaffected by the use of the SRIO unit.

- **SRIO protocol**
  - Version conforming to the SPA convention.
  - The reply from the slave does not start with a line feed.
  - The reply from the slave may be delayed up to 2.2 s.
  - A command from the master must not begin earlier than 0.05 s after the end of the preceding reply from the slave, see D.4.1.

- **Maximum transfer time of a single variable at 2400 Baud**
  - Remote read: 0.9 s/variable
  - Remote set: 1.8 s/variable

The increased data transfer time is caused by the time SMS-BASE has to wait for the SPA loop to become available. The station control system is always on-line and thus has to be able to transfer data without delay. It is therefore assigned a higher time division priority on the SPA loop than SMS-BASE which only accesses the loop when requested to do so by the user.

**D.4 Special communication settings**

The settings described in these sections are to be regarded as such additional information that only the very advanced user has advantage of. There is normally no reason to change the defaults settings.

**D.4.1 SRIO 500M /1000M and SACO 100M communication**

When the user, see figure 3.2.3.6.A, sets the protocol to SRIO by means of ‘Alter application structure’ <C> button, is an 50 ms communication delay automatically added. It is for SPA protocol zero ms, and the delay is automatically removed when setting protocol back to SPA. The parameter for this is ‘SPA-DLY_MIN_REC_TO_TRM_MS’ and it is located in the communication settings file SPIN.CNF. The value can be between 0...65535 ms and is the time that SMS-base will wait before next SPA message is sent after receiving the answer on previous one. This can from SMS-BASE v. 2.0 also be used with the SPA protocol, but then the user has to edit the file SPIN.CNF and manually give the delay in milliseconds.

```
# SPIN communication parameter file
COMMENT=default parameters
INTERFACE=NULL MODEM
BAUD RATE=9600
PARITY=None
WORD LENGTH=8
STOP BITS=1
PROTOCOL=SRIO
HANDSHAKE=None
ECHO=ignore
DEVICE=COM1
SPA-DLY_MIN_REC_TO_TRM_MS=58
```

Fig. D.4.1.A Normally the file SPIN.CNF is located at the station level in the application structure

**D.4.2 General settings**

The file GLOBAL.CNF in C:\SMS\BASE\ (default directory), contains two global parameters, i.e. TIME_OUT_WAITING_FOR_SLAVE and NO_OF_RETRIES. These settings in these parameters are used when the communication for some reason is interrupted or contact cannot be established, (‘Receive parameters’ function etc. ). These parameters define the amount of retries and the time between each retry.

A special case is when the slave answers NAK:1, i.e. “Slave busy”. Then is an additional 1 s added before retry.
The largest SPA system operates in a master/slave mode with one master and 899 slaves connected to nodes around a ring bus. The bus hardware and the communication protocol are defined for the SPA system. The communications controller in an SMS 010 system including REPORT has addresses between 901 and 999.

Although no knowledge of the SPA concept is necessary when using SMS-BASE normally, a short presentation is included at this juncture to give the user an insight into what goes on behind the scenes while communication is in progress. A full description of the SPA concept is given in other documents available from ABB Transmit Oy, Relays & Network Control division.

The bus comprises essentially an optical fibre conductor running in a loop from the master to each of the slaves in turn and back to the master. The bus is looped through each slave via a transparent optical/electrical/optical transceiver such that all the messages are available at the inputs of all the slaves. Each of the slaves are standing by and waiting for a message with its own unique address.

It is always the master that initiates a dialogue with the slaves, addressing each one in turn. Only the slave which recognises its own address in the telegram from the master responds.

The communication protocol is based on ASCII characters. A character requires 7 Bits which together with a parity bit (EVEN) makes a total of 8 Bits.

Since the SPA bus is a ring, the telegram from the master returns to it immediately and is followed after a slight delay by the reply from the slave concerned.

A simple telegram from the master has the following structure:

<table>
<thead>
<tr>
<th>&gt;</th>
<th>Start character, master signature</th>
</tr>
</thead>
<tbody>
<tr>
<td>Slave number</td>
<td>The number of the addressed slave</td>
</tr>
<tr>
<td>Message type</td>
<td>E.g. R=read, W=write</td>
</tr>
<tr>
<td>Channel number</td>
<td>Channel = Module channel</td>
</tr>
<tr>
<td>Data category</td>
<td>E.g. I=input, O=output, F=identification etc.</td>
</tr>
<tr>
<td>Data number</td>
<td>Consecutive number</td>
</tr>
<tr>
<td>:</td>
<td>Delimiter</td>
</tr>
<tr>
<td>Data</td>
<td>Data</td>
</tr>
<tr>
<td>:</td>
<td>Delimiter</td>
</tr>
<tr>
<td>Checksum</td>
<td>2 characters</td>
</tr>
<tr>
<td>Carriage return</td>
<td>Stop character</td>
</tr>
</tbody>
</table>

A reply from a slave has the following structure:

| Line feed                      | The character for a new line     |
| <                               | Slave signature as a start character|
| Slave number                   | The number of the addressed slave|
| Message type                   | E.g. A=acknowledged, D=data      |
| :                              | Delimiter                        |
| Data                           | Data                             |
| :                              | Delimiter                        |
| Checksum                       | 2 characters                     |
| Carriage return                | Stop character                   |
| Line feed                      | Stop character                   |

The checksum may be replaced by XX if it is not calculated. A telegram containing no data only requires a colon `:`.

An example of a typical dialogue is the following initial identification of a slave. The slave is assumed to have 'slave number' 31:
APPENDIX E

<table>
<thead>
<tr>
<th>Master - &gt;31RF:XX</th>
<th>No data, checksum unknown. The SPA identification F should prompt the relay type in reply.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Slave - &lt;31D:SPCJ 4D29:6A</td>
<td>Yes! This is a SPCJ 4D29 module.</td>
</tr>
</tbody>
</table>

This dialogue can be used to check the communication with the terminal emulator or SPA terminal emulator, refer to SMS-BASE User’s Guide for examples.

Each variable read or set via the SPA bus requires a dialogue. The master, SMS-BASE, requests or gives commands and the slave SPCJ 4D29 replies. Every variable transferred comprises the exchange of approximately 25 ASCII characters.
USING SMS-BASE WITH WINDOWS 3.1

SMS-BASE also runs under Windows 3.1. This is achieved with the aid of the Windows program manager command 'file\new\program group' which opens a window 'Program group properties'. Type 'C:\SMS\BASE\SMS' after the item 'Group File' in this window and click on 'OK' to create a new group window. SMS-BASE can then be started by clicking the left-hand mouse button with the cursor on the SMS-BASE. The following screen shows the SMS-BASE icon which is displayed after using the 'file\new\program group' command.

Fig. F.1 The icon for starting SMS-BASE

The following screen shows SMS-BASE and the Windows program manager running at the same time.

Fig. F.2 Running SMS-BASE in window mode

Notice! When starting the SMS-BASE from Windows 3.1 and using windowed mode you might have to acknowledge some Windows messages before continuing. The resolution might be pointed out, but by pressing <Alt> <Enter> twice you can continue in windowed mode as desired.
G  USING THE TOOL ERASEALL

This tool is found in the directory 'SMS\BASE\SUPPORT'. When using this tool you must make sure that the syntax of the command you have written is correct before pressing <Enter>.

The command erases all files and sub-directories that are found in the directory named after 'ERASEALL'.

The syntax of the command is:

C:\SMS\BASE\SUPPORT\ERASEALL  C:\SMS\BASE <directory name>

The directory can be any existing directory on your hard or floppy disk.

This tool is very convenient if you wish to delete a complete version of SMS-BASE from your hard disk. It can also be used to delete just the application structure. Assuming you accepted the default directories during program installation, the syntax in this case would be:

C:\SMS\BASE\SUPPORT\ERASEALL  C:\SMS\BASE\DATA-EX

Execution of this command deletes the entire application structure in the 'DATA-EX' directory.