

1 Preface

VTSU (Virtual Terminal Support) is a software product of the BS2000 operating system. It implements a virtual line terminal. Unlike a physical terminal, a virtual terminal permits programming that is independent of the physical characteristics of any particular terminal. The principal function of VTSU is to convert the logical control characters into their physical, terminal-specific equivalents. VTSU does not have a direct user interface. The TIAM, DCAM and UTM access methods call VTSU indirectly as a subprogram.

1.1 Target group

This manual is intended for users of the DCAM, TIAM and UTM access methods and for system and network administrators.

In order to understand this manual, users require knowledge of the BS2000 operating system, the DCAM, TIAM and UTM access methods, the programming languages used (Assembler, COBOL, FORTRAN, PL/I and C) and the software components FHS and PDN, if used.

System administrators also require system-administrator knowledge of the BS2000 operating system.

Other, related manuals are listed at the back of this User Guide.

1.2 Summary of contents

Macros and data structures that support virtual terminals are described in this User Guide.

The chapter '**Introduction to VTSU**' contains general information on virtual terminals and the functions of VTSU. This chapter also describes how VTSU is called, how it interworks with other terminal support modules, and how it is embedded in BS2000.

The chapter '**ASSEMBLER interface**' describes the ASSEMBLER macros of the VTSU interface with the VTSU control block, the logical control characters for line mode, and the status messages.

The chapter '**COBOL interface**' describes the COBOL data structures of the VTSU interface with the VTSU control block, the logical control characters for line mode, and the status messages.

The chapter '**FORTRAN interface**' lists the data structures for the VTSU FORTRAN interface. The structures are described in the corresponding sections of the chapters dealing with the ASSEMBLER and COBOL interfaces. This chapter applies to TIAM applications only.

The chapter '**PL/I interface**' lists the data structures for the VTSU PL/I interface. The structures are described in the corresponding sections of the chapters dealing with the ASSEMBLER and COBOL interfaces. This chapter applies to TIAM applications only.

The chapter '**C interface**' lists the data structures for the VTSU C interface. The structures are described in the corresponding sections of the chapters dealing with the ASSEMBLER and COBOL interfaces. This chapter applies to TIAM applications only.

The chapter '**Setting operating parameters**' describes how the system administrator can define certain terminal characteristics for the access methods before BS2000 is loaded. This chapter also describes how, using the PDN free-text parameter, the network administrator can define the type of connection and the printer type, and the 8-bit variants to be supported for the printers concerned.

The chapter '**VTSU trace**' explains to system administrators how to trace messages to terminals.

The chapter '**Special characteristics of 3270 terminals**' lists the ways in which the 3270 terminal differs from the 9750 DDT.

The chapter '**Support for special terminals**' describes support for Arabic/ Persian 8-bit terminals, Euro 7-bit terminals and Escape printers. This chapter is of interest to system administrators only.

The chapter '**Type 9750 Data Display Terminals**' describes the features characteristic of this type of terminal.

The chapter '**PLUS**' describes the program of the same name, which is used to load and store programmable keys. This chapter also includes important tables and diagrams.

The **Appendix** contains a table of standardized function-key codes and a list of the terminals supported by VTSU.

The Appendix is followed by a reference section with a **list of abbreviations** and a **glossary** that explains some of the terms used in this manual.

1.3 Changes since the last version of the manual

The VTSU macros and the VTSU data structures were described in the previous versions of the TIAM User Guide and the DCAM Macros and DCAM COBOL Calls manuals. The changes in the descriptions as they appear in this manual are as follows:

BASIC

The BASIC operand in the DCSTA macro includes the new fields STARMODE (physical read mode), STALLECH (logical end-of-line character), STASUBCH (substitute character) and STAPERHC (permanent hardcopy).

NOPOS/NO-POSITION

The VTSUCB now includes the new parameter NOPOS for defining the position of the output message.

READ/READ-MODE

The VTSUCB now includes the new parameter READ, which allows you to choose between return of modified fields only, or return of all unprotected fields.

CURPOS/CURSOR-POSITION

The VTSUCB now includes the new parameter CURPOS, which you can use to determine whether the cursor position is returned after an input or not.

UPDATE/SCREEN-UPDATE

The VTSUCB now includes the new parameter UPDATE. This parameter allows you to opt for either a complete screen refresh or update of the modified lines only.

AUTOTAB/AUTOMATIC-TABULATION

The VTSUCB now includes the new parameter AUTOTAB for defining an automatic tabulator stop.

COL/SELECT-COLOR

COL is a new parameter for logical control characters. If you are working on a terminal with color monitor, COL allows you to define colors for particular fields.

EXT RPT/REPEAT-SYMBOL

EXT RPT is a new parameter for logical control characters. It allows you to repeat the characters it precedes any number of times.

EXT DIS/DISPLAY-ATTR

EXT DIS is a new parameter for logical control characters. It allows you to assign output attributes to the field it precedes.

EXT FLD/FIELD-CHARACTERISTICS

EXT FLD is a new parameter for logical control characters. It allows you to assign field attributes to the field it precedes.

Combination of VPA and HPA/CONT-LINE-N and CONT-COL-N

The possibility of combining the VPA and HPA control characters for positioning by line and column is a new option for terminal display. The combination of these control characters is supported in extended line mode only. Combining these control characters is a way of defining the position of the cursor on the screen.

VTSU operating parameters

UTM-PERM8, DCAM-PERM8 and TIAM-PERM8 are new VTSU operating parameters. You can use these parameters to set an application to 8-bit mode.

PDN free-text parameter

As of V11, VTSU employs the new PDN free-text parameter to define the connection type and the printer type and to define the 8-bit variants supported for the printers in question. The PDN free-text parameter replaces the corresponding VTSU operating parameters.

Special terminals

An installation procedure and a configuration file have been introduced to support Arabic/Persian 8-bit terminals, Euro 7-bit terminals and Escape printers.

Readme file

Information on any functional changes and additions to the current product version can be found in the product-specific README file. You will find the file on your BS2000 computer under the name `SYSDOC.product.version.READ-ME.E`. The user ID under which the README file is cataloged can be obtained from your system administrator. You can view the README file using the `/SHOW-FILE` command or an editor, and print it out on a standard printer using the following command:

```
PRINT-FILE FILE-NAME=filename,LAYOUT-CONTROL=PARAMETERS(CONTROL-CHARACTERS=EBCDIC)
```


1.4 Notational conventions

In order to simplify the use of the manual, this description uses characters as so-called metasympols. These are already used widely in other BS2000 user manuals and are explained in the following table.

Character	Explanation	Examples
UPPERCASE LETTERS	Denote constants which the user must specify in this form	"YES"
lowercase letters	Denote variables whose contents may vary from case to case. The user must replace them with the current values.	partnername
{ }	Alternative entries are listed below each other and enclosed within braces.	{ "YES" } { "NO" }
[]	Optional entries are enclosed within square brackets. If the comma accompanying an optional entry is inside the brackets, it must be included only if this optional entry is used. If, on the other hand, the comma is outside the brackets, it must always be included even if the optional entry is omitted.	[password4] [{ "YES" } { "NO" }] filename[,ERASE]
<u>underscored</u>	Default values are underscored. The default is the value assumed by the system if no value is specified by the user.	{ <u>"YES"</u> } { <u>"NO"</u> }
...	Ellipsis indicates repetition. It shows that the preceding specification may be given more than once.	(vsn,...)

Character	Explanation	Examples
()	An expression that is used to represent a variable is enclosed within parentheses. This notation is intended to show the value range at a glance. The parentheses are needed because several characters are necessary to show the range.	(0 < length < 9)

The following notational conventions also appear in the body of the text:

<i>Note</i>	precedes particularly useful or important items of information
boldface	highlights important terms
	indicates passages that are applicable only to certain access methods or user groups

2 Introduction to VTSU

VTSU (Virtual Terminal Support) is a system-software product of the BS2000 operating system. It supports the dialog between virtual terminals and applications in BS2000. VTSU is used by the access methods TIAM (Terminal Interactive Access Method) for timesharing or interactive mode, DCAM (Data Communication Access Method) for the inquiry-and-transaction mode, and UTM (Universal transaction Monitor) for transaction processing. Terminals can be video display units, printer terminals via which a dialog can be maintained with applications in data-processing systems, and (remote) printers serving purely as output devices.

2.1 Virtual terminals

A virtual terminal constitutes a model of a terminal having certain standard characteristics. Unlike a physical terminal, a virtual terminal permits programming that is independent of the physical characteristics of a particular terminal. The physical terminal on which he or she wants to output messages is therefore of no significance to the user, who is aware only of the virtual terminal. This end is achieved by a program interface employing logical, hardware-independent control characters. The user addresses the terminals via the standard interface, and the system software for terminal support generates the physical, device-specific control characters needed for the various functions and attributes.

There are two types of virtual terminal:

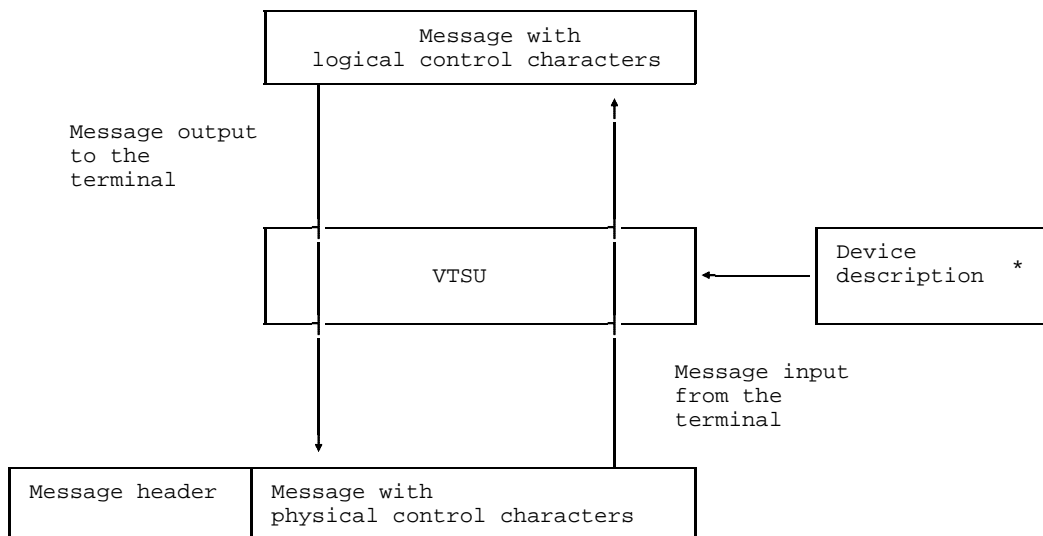
Virtual line/page terminal. An output is divided into a number of lines or pages. Users can define a virtual line structure for a virtual line terminal.

Format terminal. Forms (screen forms or masks) are output on the format terminal. The user completes only the predefined fields. The format terminal offers the program structured forms with a field structure. Format terminals are supported by restricted forms mode, which is selectable with VTSU, and by the software product FHS (Format Handling System).

VTSU implements a virtual line terminal. The principal function of VTSU is to convert the logical control characters into their physical, device-specific equivalents, e.g. record control characters, display control characters, delete character, backspace or transparent processing. In the case of transparent processing, certain parts of a message can be specified directly in physical mode.

Data transfer from an X.28 terminal via and X.25 packet switching network receives virtual support from the software product VTSU-X.29 (see the manual 'X.25PORT, VTSU-X.29'). Note that VTSU-X.29 does not use message headers.

The diagram below shows the relationship between virtual and physical terminals.



* VTSU uses tables to generate device descriptions. These tables contain control-character tables and the device-specific data.

Line terminal

The line terminal supports two modes, line mode and extended line mode. Line mode represents the true line terminal, with the line as the sole physical field structure. Extended line mode constitutes a structured line terminal offering both line and field structures.

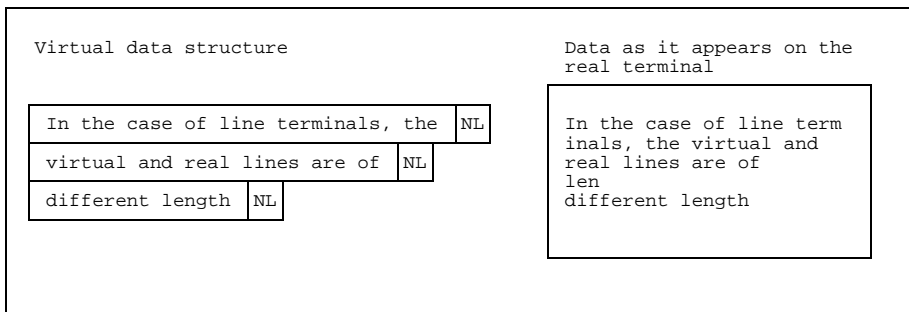
In line mode, inputs are transferred with the physical read mode 'Read Modified', in other words only data that has been modified is transferred. In extended line mode 'Read Unprotected' is the default physical read mode for data inputs. In this mode, all unprotected, user-modifiable fields are read. In order to ascertain which fields have been modified, the application program must compare the generated output fields with the corresponding received input values. In extended line mode the screen is protected and the fields are blanked out by default. If a line is only partially filled, the rest of the

line remains blanked. This means that an elementary forms mode is possible in extended line mode. The forms can be generated on the fly by the application program.

The diagrams below show the virtual data structure and output to virtual line terminals and virtual format terminals.

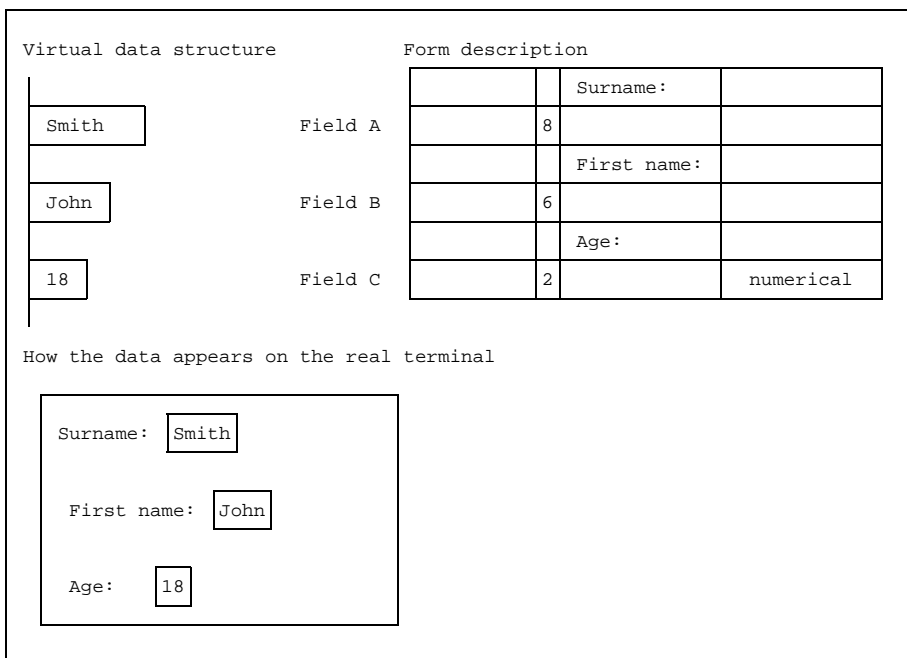
When you use a **line terminal** you can divide messages into virtual lines. Each virtual line is terminated by a 'new line' character.

At the terminal, the text following this character is automatically placed at the start of the next line. If a virtual line is longer than the line length permitted by the terminal, the virtual line is automatically split. The start of a message is always positioned at the start of a new line.



Virtual line terminals

Supported by VTSU and FHS, the virtual **format terminal** is based on a separate form description. This description is generated by the Interactive Forms Generator (IFG) and stored in a forms library. The form description contains a description of the fixed texts (e.g. "Surname:", "First name:", "Age:") and a description of the fields (length and type). The application uses the logical data structure. FHS builds the output screen with the aid of the form description and the virtual data structure. The input screen is analyzed by FHS, which returns the modified virtual data structure to the application.



Virtual format terminal

810 protocol

The 810 protocol is a device protocol which describes the hardware/software interface of the terminals. The structure of the actual message is defined in the 810 protocol. The transport functions are described by the NEA, TCP or ISO transport protocols and not by the 810 protocol.

A message consists of the message prefix and the message. The prefix contains the start of message character and, if applicable, the message header (device protocol header). The message header consists of parameter areas which control the global functions of the terminal, such as automatic hardcopy. The parameter areas can also be sent in the message along with the message text, in which case the message does not have a header and the message prefix contains only the start of message character. The message text is accompanied by physical control characters or control character strings of the 810 protocol. These control characters influence the character-related functions of the terminal. Control characters that can be included in the message text are: absolute positioning instructions, field separators and device functions.

The printer control characters differ from printer type to type and are therefore not included in the 810 protocol. The structure of the printer messages is such that the message headers correspond to the 810 protocol, but the type and structure of the control characters differ for individual printer types.

Connection of 3270 terminals

Users operating in a non-homogeneous landscape (using terminals corresponding to IBM's 3270, in other words) can employ VTSU's facility for operating these terminals like 9750 terminals. VTSU converts the logical control characters into physical control characters in accordance with the 3270 protocol.

Note that the device protocol for the non-Siemens Nixdorf terminal (IBM 3270) supported by VTSU is not part of the 810 protocol.

The interface between the TRANSDATA and SNA networks is implemented by a communication computer. The TRANSIT-CD product is used as the gateway.

Physical mode

In addition to the (extended) line mode, VTSU also supports a physical mode. Logical control characters are not processed in this mode. Users must generate their messages complete with the correct, hardware-specific control characters.

This mode should be selected by users wanting to use special functions not available as standard functions of a virtual terminal.

In this mode, VTSU does not process the message. If the user did not define the standard message header, VTSU automatically prefixes it to the message prior to transmission.

2.2 Functions of VTSU

VTSU supports the following functions

Output message handling:

- Conversion of logical control characters into device-specific control characters for output
- Compilation of the device protocol in the message header or in the parameter specifications
- Overflow control
- Tagging of system outputs or input requests from the system or TIAM application program with special prompt characters (% , / , *)
- 8-bit code conversion and insertion of SI/SO control characters, if necessary, in accordance with device connection (7-bit or 8-bit connection) in the case of 8-bit devices (see XHCS manual). This function cannot be used unless the software product XHCS (Extended Host Code Support) is available. Note that the SI/SO characters are not inserted until the end of VTSU processing. If the buffer is full, the 8-bit characters are replaced by the hexadecimal value X'3F'.

Input message handling:

- Conversion of incoming device-specific control characters to their logical equivalents
- Code conversion, e.g. uppercase and lowercase, diacriticals, 8-bit code conversion if XHCS is used
- Interpretation of the SI/SO characters, followed by update of the hexadecimal values of 8-bit characters for 8-bit devices communicating via a 7-bit connection (see XHCS User Guide). For example, a BAM procedure permits only the transmission of a 7-bit code frame. This function cannot be used unless the software product XHCS is available.

Monitoring functions

- Information on the current status of the connection is returned on request (i.e. to the TIAM macro TSTAT or the DCAM macro YINQUIRE). VTSU receives this information from PDN via the NEABV user service protocol at the time when the connection is set up (generation information) or directly from the device in response to a status query.
- Modification of device attributes (e.g. number of virtual lines, overflow checking, substitute characters, ...) either directly using commands to the terminal or via the application program.

Plausibility checks:

- The validity of the edit options selected by the user and the VTSUCB parameter specifications is checked. If VTSU discovers any inconsistency either the call is rejected with an explanatory return code (e.g. XHCS conversion in wrong environment) or suitable corrective measures are effected and a warning return code is issued.

Generation:

- VTSU sets certain device functions by means of the terminal generation in PDN or by means of the device types defined when the connection was set up by the access methods, for example generation of the terminal as PTERM for UTM. VTSU, in other words, does not incorporate true generation functionality.
- General functions or device attributes are configured by the entries in a structured parameter file when VTSU is initialized. No subsequent changes to these values are possible; they remain active until the end of the session or until VTSU is unloaded.

8-bit support:

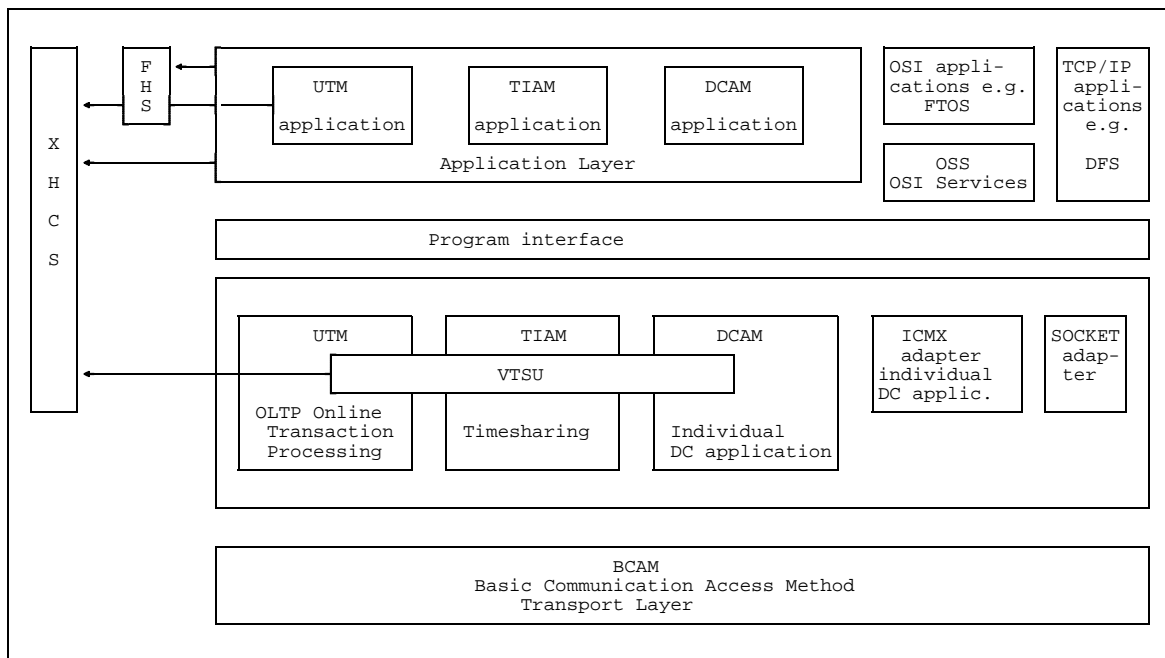
- VTSU converts 8-bit codes, adding the SI/SO control character when necessary (see the XHCS User Guide). Note, however, that this means using 8-bit devices and the software product XHCS.

2.3 VTSU in the system environment

The functions discharged by VTSU can be assigned to the Presentation Layer of the OSI Reference Model. VTSU does not discharge functions assignable to the Transport Layer.

This means that VTSU does not have a direct user interface. The TIAM, DCAM or UTM access methods must be used for inputs and outputs.

The diagram below shows VTSU and how it is embedded in the system environment.

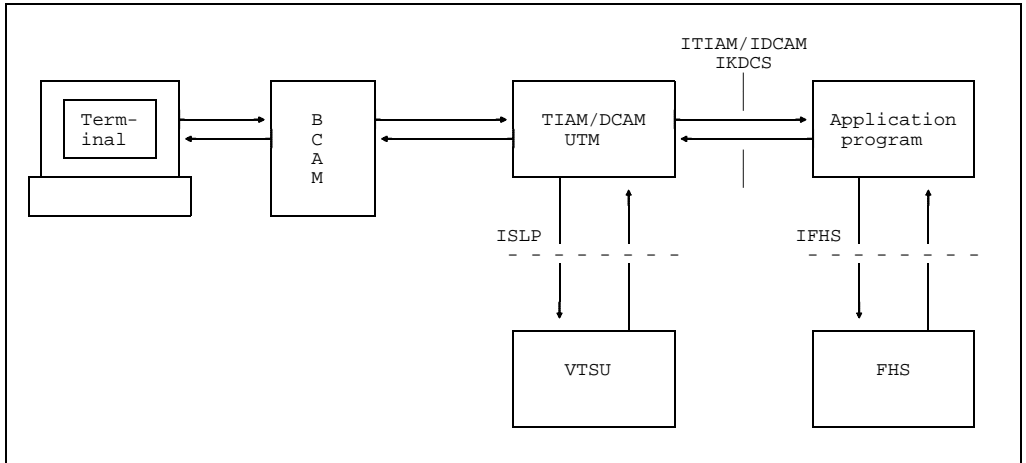


VTSU in the system environment

The TIAM, DCAM and UTM access methods call the VTSU software component indirectly as a subprogram. If you are using a TIAM or DCAM application, the VTSUCB enables you to send a number of options indirectly to VTSU for processing. VTSU is completely invisible to UTM users. VTSU connects the TIAM, DCAM and UTM access methods to XHCS. The appropriate TIAM, DCAM or UTM application program enables you to access XHCS. Note, however, that UTM users can use XHCS only via FHS. DCAM and TIAM users have access to XHCS either 'directly' (with the VTSUCB), or by means of FHS.

2.3.1 Calling VTSU

The DCAM, TIAM and UTM access methods call VTSU indirectly as a subprogram via the privileged program interface ISLP (Interface Station Level Processor).



How TIAM, DCAM and UTM call VTSU

If you are using a TIAM or DCAM application, you can send certain options to VTSU indirectly for processing in the VTSUCB (VTSU control block). The VTSUCB passes these options to VTSU via the application interface of TIAM or DCAM.

The VTSUCB replaces the edit options integrated in the interfaces of the access methods. Its purpose is to offer new functions directly to the user without the need to modify the access method used.

If you operate your terminals in format mode with FHS, the virtual format terminal is made available by the presentation functions of FHS.

As of BS2000/OSD-BC V1.0, VTSU is called directly by join management as part of LOGON processing in order to check and process the extended user standard code. This check ascertains whether the extended user standard code is supported by XHCS and the connected terminal. If it is not, the EDF03IRV code is used instead.

2.3.2 VTSU as a subsystem

VTSU is a subsystem of DSSM and is loaded by DSSM during system start. Every time a connection is set up, VTSU calls DSSM to link the calling task to the VTSU subsystem. This mechanism prevents VTSU being unloaded while the task is still using it.

2.3.3 Internationalization

International usage requires adjustment to allow for national standards and customs. Code conversions are necessary to permit the use of national character sets. As of V10.0A, VTSU supports international 8-bit character sets, but this entails the use of 8-bit devices (9758-M4 and 9763-D7 DDTs, 9011-28/29, 9021,.. Printers) and the software product XHCS. VTSU implements all requisite code conversions for both input and output messages. In the case of 8-bit devices communicating via a 7-bit connection, VTSU adds the SI/SO characters to outgoing messages and suppresses them in incoming messages. VTSU does not implement code conversion for other devices or by way of supporting national character sets.

VTSU incorporates special routines for adaptation to national character sets. These special routines are run for every message handled and are used to pre-edit (input) and to post-edit (output) messages.

2.4 How VTSU interworks with other terminal-support implementations

Terminal support in BS2000 is an implementation that controls terminals and how terminals and BS2000 applications communicate. This implementation is located above the transport system (Layers 1 - 4).

2.4.1 The OSI Reference Model

The OSI Reference Model is now acknowledged as a standard model for communication architectures. It describes the possibilities for open communication between differing systems.

The OSI Reference Model divides the extensive communication functions across seven different layers. The functions that must be discharged are defined for each layer. These functions are outlines in the table below. The seven layers form a hierarchy, i.e. each layer has at its disposal the functions of the subordinate layers as services. The way in which the functions of a particular layer are implemented is of no significance to the higher layers.

The protocols lay down the rules for interaction and the formats used by the communicating partners. Any two partners belonging to the same layer communicate by means of a common protocol.

Layer	Designation	Functions	
Layer 7	Application Layer	Controls the execution of communication functions for an application	} A p p l i c a t i o n
Layer 6	Presentation Layer	Determines the presentation of the information and permits user- and device-independent communication	
Layer 5	Session Layer	Controls the procedure for communication	
Layer 4	Transport Layer	Controls the reliable exchange of data between two communicating partners	} T r a n s p o r t
Layer 3	Network Layer	Sets up and clears down the connection between the sending and receiving systems	
Layer 2	Data Link Layer	Protects data transfer on the individual links that make up the transmission route (procedures)	
Layer 1	Physical Layer	The physical connection (by means of the medium used for communication)	

The OSI reference Model covers only aspects relating to communication, ignoring all other aspects such as program interfaces.

2.4.2 Terminal support in BS2000

The purpose of terminal support in BS2000 is to provide applications in BS2000 with easy-to-use interfaces for terminal access and control and to offer users a standardized access to and dialog with applications in BS2000.

BS2000 supports only 97xx Terminals and 3270 terminals. Users wanting to access BS2000 applications from other terminals must run a 97xx terminal emulation either locally in a PC or in a central terminal server.

Terminal support is furnished by components in the BS2000 host and by components in the terminal nodes (e.g. intelligent terminals) or terminal servers (e.g. SINIX multiuser systems) to which the terminals are connected.

The terminal support component on the BS2000 host is known as the terminal access method. This component incorporates all the functions needed to convert the functions of the local interfaces for application programming into virtual protocols to control the remote terminals. The terminal support component running on the BS2000 host is in no way dependent on the means by which the terminals are connected to the TRANSDATA communication network.

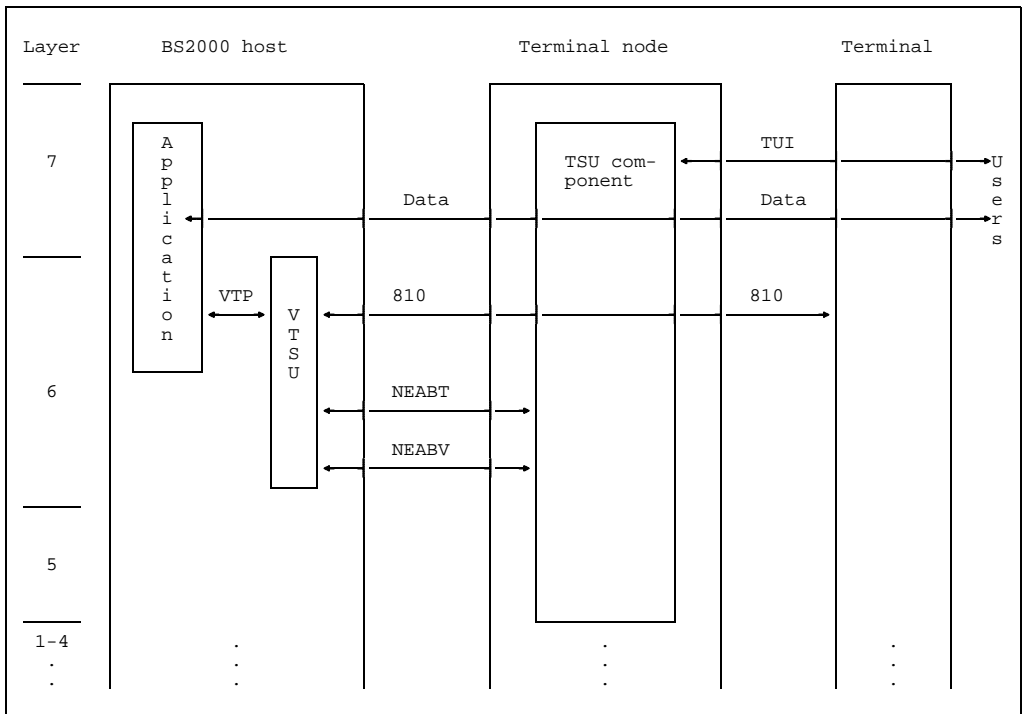
The terminal support component running in the terminal node or terminal server implements the functions for local terminal control in accordance with the terminal control protocol received via the network. This component is the terminal support unit (TSU) in terminal nodes (PDN, INCA, C30/C50/C70), or the 9750 terminal emulation in the terminal servers. The control protocol is incorporated in the NEABV and NEABT protocols for BS2000 terminals.

The NEA user service connection protocol (NEABV) exchanges information between the communicating partners in the TRANSDATA network in order to set up the connection. The information exchanged in this way defines the user service and its parameters such as partner characteristics and partner character sets.

The TIAM user service protocol (NEABT) is used to implement timesharing mode with the TIAM access method, inquiry-and-transaction mode with the DCAM access method and transaction processing mode with the UTM access method in BS2000. The NEABT protocol enables the exchange of user messages, control of the message flow and message editing.

The diagrams on this page and the next show how VTSU interworks with the TSU and terminal emulation components in the TRANSDATA network.

How VTSU interworks with TSU



Terminal node (e.g. PDN, INCA, C30/C50/C70): end system in the sense of the transport link
Terminal (e.g. 9750, 9001, etc.)

TUI (Terminal User Interface): user inputs and messages for connection setup
(e.g. 'o \$dialog'; 'CN04 Connected with ...')

Data: net data between user (terminal operator) and application

VTP (Virtual Terminal Protocol): virtual device protocol between application and VTSU

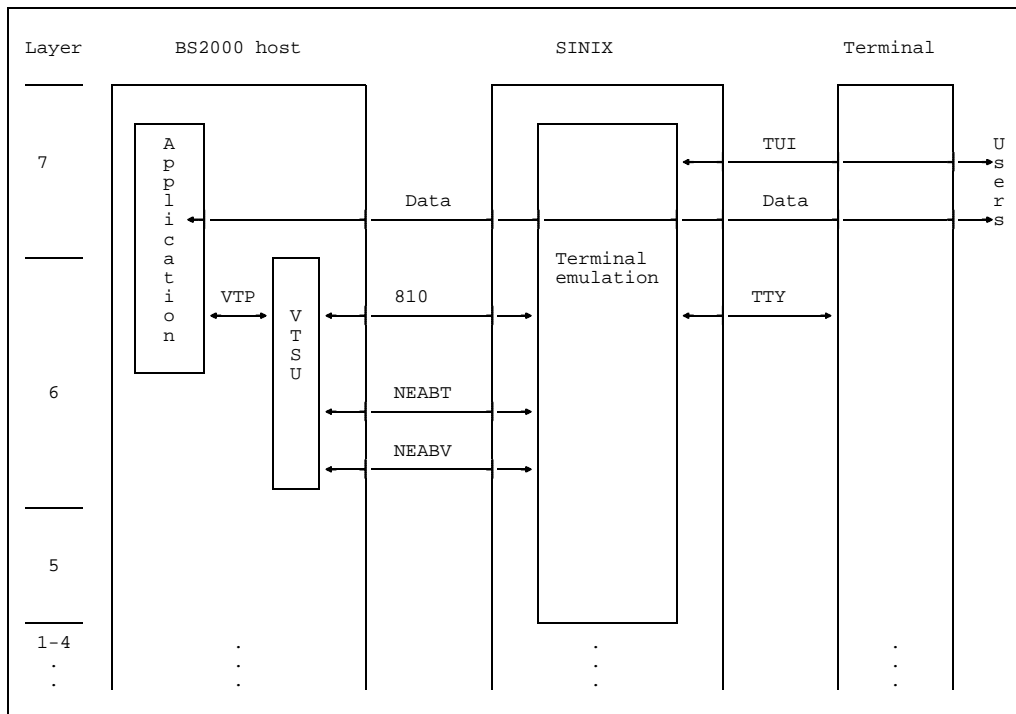
810 : Device protocol of the 9750 etc. block terminals

NEABV: Protocol regulating data exchange for connection setup

NEABT: Protocol regulating the data-transfer phase

How VTSU interworks with TSU components

How VTSU interworks with the terminal emulation



SINIX: End system in the sense of the transport link

Terminal (Sinix) (e.g. 97801)

TUI (Terminal User Interface): User inputs and messages for connection setup
(e.g. 'o \$dialog'; 'CN04 Connected with ...')

TTY: Device protocol of the 9780x character-oriented terminal

Data: Net data between user (terminal operator) and application

VTP (Virtual Terminal Protocol): virtual device protocol between application and VTSU

810 : Device protocol of the 9750 etc. block terminals

NEABV: Protocol regulating data exchange for connection setup

NEABT: Protocol regulating the data-transfer phase

How VTSU interworks with the terminal emulation

This diagram applies by analogy to DOS computers running the MT9750 emulation.

2.5 BS2000 and version independence

VTSU is a DSSM subsystem and is therefore independent of BS2000 versions. VTSU V11.0 has a link to certain BS2000 interfaces and is therefore divided internally into two different variants. The variant is dependent on the BS2000 version installed on the host. DSSM activates the right variant on the basis of the subsystem catalog.

2.6 VTSU in UTM applications

If you use a UTM application, you do not have an interface to the VTSU macros and data structures. UTM uses VTSU as a system-internal subsystem for editing messages from and to the terminals. VTSU is invisible to UTM users. Note that for UTM-T applications in V3.3A or higher, the VTSU module library must be assigned explicitly (see UTM manual Generating and Administrating Applications). For UTM-T applications < V3.3A the VTSU modules are automatically loaded when the application starts.

3 ASSEMBLER interface

This chapter describes how virtual terminals support ASSEMBLER applications. The macros available for this purpose are DCSTA, VTCSET and VTSUCB.

The **DCSTA** macro generates destination fields or symbolic field names for status information. This information can be obtained with the aid of the TSTAT macro (TIAM) or YINQUIRE (DCAM). The information queried is as follows:

- Partner characteristics, e.g. partner type, terminal type
- Description of the terminal and the character sets, e.g. type and number of virtually addressable character sets, screen format
- Description of connected peripherals, e.g. ID card reader, chip card terminal
- Basic terminal information, e.g. type of terminal, number of colors

The **VTCSET** macro converts logical control characters into device-specific control characters for the line terminal. Logical control characters are:

- Record control characters such as new line, new page, characters per line, etc.
- Display control characters for message display, including highlight, select character set, character spacing, line spacing, etc.
- Miscellaneous functions such as delete character, backspace, etc.

The **VTSUCB** macro allows you to set VTSU parameters for input and output regardless of the access method. These parameters correspond to the edit options; note that new edit options will be implemented only by the VTSUCB. The VTSUCB can be included in YSEND, YRECEIVE and YSENDREC calls (DCAM applications) and in WROUT, WRTRD and RDATA calls (TIAM applications).

All macros are contained in the \$TSOS.SYSLIB.VTSU-B.110 library.

3.1 Status information: DCSTA

The DCSTA macro generates destination fields or symbolic field names (DSECTs) for information about terminal characteristics.

If you are using a TIAM application, you can query this information with the TSTAT macro. If you are using a DCAM application, the macro is YINQUIRE using the OPTCD values PTNCHAR, BTERMINF, PEROTERM and MONCHARS (see the 'TIAM' or 'DCAM' User Guide, as appropriate).

Name	Operation	Operands
[name]	DCSTA	<div style="display: flex; align-items: center;"> <div style="margin-right: 10px;"> $\left\{ \begin{array}{l} [C] \\ [D] \end{array} \right.$ </div> <div style="margin-right: 10px;">[,prefix],TYPE=</div> <div style="display: flex; align-items: center;"> <div style="border-left: 1px solid black; padding-left: 5px; margin-right: 5px;"> TCHAR PHDIM LIDIM VDT[YP] EDOPT OFLOW STNAM PRNAM </div> <div style="margin-left: 10px;"> $\left. \vphantom{\begin{array}{l} TCHAR \\ PHDIM \\ LIDIM \\ VDT[YP] \\ EDOPT \\ OFLOW \\ STNAM \\ PRNAM \end{array}} \right\}$ </div> <div style="margin-left: 10px;"> TIAM only </div> </div> </div> <div style="display: flex; align-items: center;"> <div style="margin-right: 10px;"> $\left. \vphantom{\begin{array}{l} ALL \\ MONCS \\ PERPH \\ BASIC \end{array}} \right\}$ </div> <div style="margin-left: 10px;"> DCAM and TIAM </div> </div>

Description of parameters

name	<p>becomes the symbolic name of the first DS statement in the macro expansion if the C operand is specified. (The length attribute is zero).</p> <p>If operand D is specified, 'name' is the name of the dummy section (DSECT).</p> <p>If the 'name' entry is omitted, the system creates a name from the currently valid prefix (see 'prefix' operand) and the operand value for TYPE.</p>
C	A storage area (CSECT) is generated.
D	A dummy section (DSECT) is generated.
prefix	<p>1 to 3 characters with which the generated field name is to begin. The specified prefix replaces the string 'STA' with which the field names would otherwise begin (as the default option).</p>
TYPE	<p>specifies the type of destination field or the field names to be generated: users of DCAM applications cannot query the individual operands TCHAR though PRNAM.</p>
=TCHAR	queries the terminal characteristics.
=PHDIM	queries the physical dimensions of the terminal.
=LIDIM	queries the logical dimensions of the terminal.
=VDT	queries the virtual device type of the terminal.
=EDOPT	queries the static edit options.
=OFLOW	queries the type of overflow control.
=STNAM	queries the station (terminal) name.
=PRNAM	queries the processor name.
=ALL	queries all information from TCHAR through PRNAM.
=MONCS	queries the monitor and character sets.
=PERPH	queries the connected peripherals.
=BASIC	queries the basic information on the terminal.

The following pages show the symbolic names of the individual fields and their possible contents.

Description of the information supplied:

TCHAR: **Physical type** (area length: 8 bytes)

Byte	Symb. name	Meaning
0	STAPTTYP	Partner type:
	(STADCAMP)	Partner is a DCAM program
	(STADCAMT)	Partner is a terminal
1	STADVITYP	Processed device type (e.g.):
	(STAD1000)	1000 Teleprinter
	(STAD100E)	100-E Teleprinter
	(STADT100)	100 Teleprinter
	(STADPT80)	PT80 Printer Terminal
	(STAD8110)	8110 Printer Terminal
	(STAD8151)	8151 Data Display Terminal
	(STAD8152)	8152 Data Display Terminal
	(STAD8160)	8160 Data Display Terminal
	(STAD8162)	8162 Data Display Terminal
	(STAD9731)	9731 Graphics Workstation
	(STAD9750)	9750/9749 Data Display Terminal
	(STAD9751)	9751 Data Display Terminal
	(STAD9752)	9752 Data Display Terminal
	(STAD9753)	9753 Data Display Terminal
	(STAD9754)	9754 Data Display Terminal
	(STAD9755)	9755 Data Display Terminal
	(STAD9763)	9763 Data Display Terminal
	(STAD8122)	8122 Printer
	(STAD8121)	8121 Printer
	(STAD9001)	9001 Printer
	(STAD9002)	9002 Printer
	(STAD9003)	9003 Printer
	(STAD9004)	9004 Printer
	(STAD9012)	9012 Printer
	(STAD9013)	9013 Printer
	(STAD0131)	9001-31 Printer
	(STAD0189)	9001-8931 Printer
	(STAD9022)	9022 Printer
	(STAD1118)	9011-18 Printer
	(STAD1119)	9011-19 Printer
	(STAD3270)	3270 terminal
	(STADHOST)	Program on host computer
	(STADAP)	AP station
	(STAD9021)	9021 Printer
	(STAD3287)	3287 Printer
(STAD9014)	9014 Printer	
(STAD9026)	9026 Printer (HDL, 9025-compatible)	
(STADFE)	Front-end data display terminal (FHS-DOORS)	

Notes

- The 9749 Data Display Terminal can be generated as a separate device type in PDN. For application programs, however, it is always identified as a 9750 in the TSTAT/YINQUIRE macros.
- The 9758 M4 DDT can be generated as a 9755 or 9763 in PDN. It must, however, be generated as a 9763 DDT in order to operate in 8-bit mode. For application programs, the 9758 M4 is always identified as a 9750 in the TSTAT/YINQUIRE macros.

Byte	Symb. name	Meaning
2	STATCHR2 (STATC2EX) (STATC2LC) (STATC2DT) (STATC2DF)	Character set: Second character set available Lowercase letters available German (instead of international) keyboard generated Byte 2 defined
3	STATCHR3 (STATC3H1) (STATC3H2) (STATC3IC) (STATC3FD) (STATC3AP) (STATC3GF) (STATC3DZ) (STATC3DF)	Terminal options: Local hardcopy unit generated or assigned with TCHNG Central hardcopy unit generated or assigned with TCHNG ID card reader generated or assigned with TCHNG Floppy disk generated or assigned with TCHNG APL capability generated or assigned with TCHNG Graphics feature generated or assigned with TCHNG Decentralized formatting Byte 3 defined
4	STATCHR4 (STATC4CO) (STATC4ZF) (STATC4ST) (STATC4HI) (STATC4C8) (STATC4HP) (STATC4DF)	Terminal functions: 4 colors Character and field attributes Status of terminal possible Hardware system status line available 8 colors HP Laser Jet II Byte 4 defined
5	STATTCHRS (STATCSDT) (STATCSHC) (STATCSIC) (STATCSDF)	Information from the status message German keyboard connected Local hardcopy unit connected ID card reader connected Status of terminal available
6	STACTRLU	Device type of printer controller if the terminal is a printer: X'00' for a type 8112 Printer Terminal Controller; generated device type (see above) for data display terminals
7	STACHCAD	Channel address of central hardcopy unit

PHDIM: Physical dimensions (area length: 8 bytes)

PHDIM: Physical dimensions (area length: 8 bytes)

Byte	Symb. name	Meaning
0 - 1	STALLEN	Physical line length
2 - 3	STANOLIN	Physical number of lines (unlimited on printer terminals: X'7FFF')
4 - 5	STAMAXDB	Maximum physical device buffer, i.e. maximum number of characters that can be transmitted to the terminal with one output request. X'7FFF' (unlimited): limited only by the access method or line
6 - 7	-	reserved

Bit 2¹⁵ = 1 means (in each case): value not available.

LIDIM: Logical dimensions (line mode) (area length: 8 bytes)

Byte	Symb. name	Meaning
0 - 1	STALLEEN	Number of characters per physical line in line mode; 'NL' in text is to be counted as two characters.
2 - 3	STALNOLN	Number of physical lines that can be output in line mode without activating the overflow control.
4 - 5	STALMAXB	Number of characters that can be transmitted in one message in line mode without activating the overflow control (usually: lines times columns minus 1).
6 - 7	-	Reserved

Bit 2¹⁵ = 1 means (in each case): value not available.

VDT[YP]: Virtual device type (area length: 8 bytes)

Byte	Symb. name	Meaning
0	STAVDT (STALINCP) (STAFORCP) (STACMPCP) (STAFYSCP) (STAEXLCP) (STAAUTLF) (STANOINP) (STAEOM=0) (STAEOM=1)	Virtual device type: Line/page terminal (LINE MODE) Format terminal (FORM MODE) Terminal support compatible with earlier operating system versions (COMP MODE) Physical terminal support (PHYS MODE) Line/page terminal (ext. LINE MODE) Automatic Line Feed for printer Terminal is a printer Terminal is a printer terminal Terminal is a data display terminal
1	STAVDTPR (STATD810) (STAT3270)	Virtual device protocol: 810 protocol 3270 protocol
2 - 7	-	Reserved

EDOPT: Static edit options (area length: 8 bytes)

The symbolic operands of the WRTRD macro are used to represent these options.

Byte	Symb. name	Meaning
0	STASEWR1 (STAWR1MM) (STAWR1CD) (STAWR1LE) (STAWR1RE) (STAWR1HO) (STAWR1PT) (STAWR1HC)	Output edit option byte 1 Mask for output edit mode =STAWR1CO: MODE=COMP =STAWR1LI: MODE=LINE =STAWR1FO: MODE=FORM =STAWR1FY: MODE=PHYS =1: OTRSUP =Y =0: =N =1: ONLINEND =Y =0: =N =1: OMANUAL =Y =0: =N =1: OHOM =Y =0: =N =1: OPTAPE =Y =0: =N =1: OHCOPY =Y =0: =N

Byte	Symb. Name	Meaning
1	STASEWR2	Output edit option byte 2
	(STAWR2HD)	=1: OHDR =Y =0: =N
	(STAWR2NO)	=1: ONOLOGC =Y =0: =N
	(STAWR2EX)	=1: EXTEND =Y =0: =N
	(STAWR2ET)	=1: OETB =Y =0: =N
	(STAWR2BL)	=1: OBELL =Y =0: =N
	(STAWR2TP)	=1: OTRANS =Y =0: =N
	(STAWR2IM)	=1: OINFO =Y =0: =N
	(STAWR2PN)	=1: ONOPOSN =Y =0: =N
2	STASERD1	Input edit option byte 1
	(STARD1MM)	Mask for input edit mode =STARD1CO: MODE=COMP =STARD1LI: MODE=LINE =STARD1FO: MODE=FORM =STARD1FY: MODE=PHYS
	(STARD1CD)	=1: ITRSUP =Y =0: =N
	(STARD1LE)	=1: ILINEND =Y =0: =N
	(STARD1BS)	=1: IGETBS =Y =0: =N
	(STARD1PT)	=1: IMANUAL =Y =0: =N
	(STARD1LC)	=1: ILCASE =Y =0: =N
	(STARD1HD)	=1: IHDR =Y =0: =N
3	STASERD2	Input edit option byte 2
	(STARD2FC)	=1: IGETFC =Y =0: =N
	(STARD2IC)	=1: IGETIC =Y =0: =N
	(STARD2CF)	=1: ICFD =Y =0: =N
	(STARD2EX)	=1: EXTEND =Y =0: =N
4 - 7	-	Reserved

OFLOW: Screen overflow control (area length: 8 bytes)

Byte	Symb. name	Meaning
0	STAOFLOW	Type of overflow control:
	(STAOFCTM)	Bit 2 ⁰ =1 Overflow control when using the timer. n seconds waiting time (hexadecimal notation) according to STAOFTIM
	(STAOFCAK)	Bit 2 ¹ =1 Overflow control with request for acknowledgment if overflow occurs
	(STAOFCTL)	Bit 2 ⁰ =0 } Bit 2 ¹ =0 } No overflow control
	(STAOFPGM)	Bit 2 ⁵ =0 Overflow control by the system
		Bit 2 ⁵ =1 Overflow control by the application program (see TCHNG macro, MODIFY-TERMINAL-OPTIONS command)
1	STAOFTIM	Waiting time (hexadecimal entry, in seconds)
2 - 7	-	Reserved

STNAM: Station name, as specified in PDN (area length: 8 bytes)

Byte	Symb. name	Meaning
0 - 7	STASTNAM	Name of station

PRNAM: Processor name, i.e. processor to which station is connected, as specified in RDF generation (area length: 8 bytes)

Byte	Symb. name	Meaning
0 - 7	STAPRNAM	Name of processor

ALL:

The symbolic addresses of the table fields are defined as described above. The order of the table entries is as follows:

TCHAR
PHDIM
LIDIM
VDTYP
EDOPT
OFLOW
STNAM
PRNAM

Note

If TYPE=ALL is specified, the size of the area is 64 bytes for TIAM applications. In the case of DCAM applications, the size is 48 bytes. TYPE=ALL does not include MONCS, PERPH and BASIC.

MONCS: Description of monitor and character sets (area length: min. 14 bytes)

Information is supplied via the status message of the terminal, if available; otherwise default values are taken.

Byte	Symb. name	Meaning
0	STAMOCPR (STAMOCY) (STAMOCN)	Status message of the station Status message available. The following information is taken from the status message. Status message not available. VTSU default values are taken for the following information.
1	STAMOTYP (STAMONO) (STACOLOR) (STAPRINT)	Monitor type of the station Terminal with monochrome display Terminal with color display Terminal is a printer
	STAFAT (STAFATY) (STAFATN)	Extended field attributes Extended field attributes can be used (9759 and 9763 DDTs only) Extended field attributes cannot be used
3	-	reserved

Byte	Symb. name	Meaning
4	STADIM1 (STADIMY) (STADIMN)	Screen size 24 lines x 80 characters Format is supported (addressable with DIM) Format is not supported (not addressable with DIM)
5	STADIM2 (STADIMY) (STADIMN)	Screen size 32 lines x 80 characters Format is supported (addressable with DIM) Format is not supported (not addressable with DIM)
6	STADIM3 (STADIMY) (STADIMN)	Screen size 43 lines x 80 characters Format is supported (addressable with DIM) Format is not supported (not addressable with DIM)
7	STADIM4 (STADIMY) (STADIMN)	Screen size 27 lines x 132 characters Format is supported (addressable with DIM) Format is not supported (not addressable with DIM)
8-11	-	Reserved
12-13	STACSNO	Number of addressable character sets
14	STACS0T (STACSSIN) (STACSTRI) (STACSNO)	Type of character set 0 Loadable monochrome character set Loadable color character set Non-loadable character set
15	STACS0S (STACSNLO) (STACSDSS) (STACSDVN) (STACSDVA)	Status of character set 0 Character set can be loaded Character set assigned by the terminal Character set loaded by the DP system Character set loaded and assigned by the DP system
16	STACS1T (STACSSIN) (STACSTRI) (STACSNO)	Type of character set 1 Loadable monochrome character set Loadable color character set Non-loadable character set

Byte	Symb. name	Meaning
17	STACS1S (STACSNLO) (STACSDSS) (STACSDVN) (STACSDVA)	Status of character set 1 Character set can be loaded Character set assigned by the terminal Character set loaded by the DP system Character set loaded and assigned by the DP system
18	STACS2T (STACSSIN) (STACSTRI) (STACSNO)	Type of character set 2 Loadable monochrome character set Loadable color character set Non-loadable character set
19	STACS2S (STACSNLO) (STACSDSS) (STACSDVN) (STACSDVA)	Status of character set 2 Character set can be loaded Character set assigned by the terminal Character set loaded by the DP system Character set loaded and assigned by the DP system
20	STACS3T (STACSSIN) (STACSTRI) (STACSNO)	Type of character set 3 Loadable monochrome character set Loadable color character set Non-loadable character set
21	STACS3S (STACSNLO) (STACSDSS) (STACSDVN) (STACSDVA)	Status of character set 3 Character set can be loaded Character set assigned by the terminal Character set loaded by the DP system Character set loaded and assigned by the DP system
22	STACS4T (STACSSIN) (STACSTRI) (STACSNO)	Type of character set 4 Loadable monochrome character set Loadable color character set Non-loadable character set
23	STACS4S (STACSNLO) (STACSDSS) (STACSDVN) (STACSDVA)	Status of character set 4 Character set can be loaded Character set assigned by the terminal Character set loaded by the DP system Character set loaded and assigned by the DP system

Byte	Symb. name	Meaning
24	STACS5T (STACSSIN) (STACSTRI) (STACSN0)	Type of character set 5 Loadable monochrome character set Loadable color character set Non-loadable character set
25	STACS5S (STACSNLO) (STACSDSS) (STACSDVN) (STACSDVA)	Status of character set 5 Character set can be loaded Character set assigned by the terminal Character set loaded by the DP system Character set loaded and assigned by the DP system
26	STACS6T (STACSSIN) (STACSTRI) (STACSN0)	Type of character set 6 Loadable monochrome character set Loadable color character set Non-loadable character set
27	STACS6S (STACSNLO) (STACSDSS) (STACSDVN) (STACSDVA)	Status of character set 6 Character set can be loaded Character set assigned by the terminal Character set is loaded by the DP system Character set is loaded and assigned by the DP system
28	STACS7T (STACSSIN) (STACSTRI) (STACSN0)	Type of character set 7 Loadable monochrome character set Loadable color character set Non-loadable character set
29	STACS7S (STACSNLO) (STACSDSS) (STACSDVN) (STACSDVA)	Status of character set 7 Character set can be loaded Character set assigned by the terminal Character set loaded by the DP system Character set loaded and assigned by the DP system

If the status area is too small, bytes 14-29 are discarded without a return code being supplied. Information on the character sets is only supplied if the character sets are addressable via bytes 12-13.

PERPH: Connected peripherals (area length: 8 bytes)

Information is supplied via the status message of the terminal, if available; otherwise it is taken from the generation parameters.

Byte	Symb. name	Meaning
0	STAPERPR (STAPERY) (STAPERN)	Status of the station Terminal status message available No terminal status message available
1-2	-	Reserved
3	STALOCHC (STALHCY) (STALHCN)	Local hardcopy unit Local hardcopy unit connected No local hardcopy unit connected
4-5	-	Reserved
6	STAIDCAR (STAIDCY) (STAIDCN)	ID card reader ID card reader connected No ID card reader connected
7	STACKT (STACKTY) (STACKTN)	Chip card terminal Chip card terminal connected No chip card terminal connected

BASIC: Basic information on the terminal (area length: 64 bytes)

Information is supplied via the status message of the terminal, if available; otherwise it is supplied from the generation parameters or VTSU default values are taken.

If you define a length greater than 24 bytes and less than 33 bytes, information is supplied on bytes 0-23 only.

If you define a length of 33 bytes, information is supplied on bytes 0-32 only. If XHCS is not loaded or if the data terminal does not support extended character sets, no information is supplied on the terminal type and name of the extended standard character set.

If you define a length greater than 33 bytes and less than 52 bytes and the data display terminal is operating in 8-bit mode, the first byte in this area is set to X'00. The incoming information is truncated, because the receive area you specified is too small.

If you define a length of 52 bytes, information is supplied on bytes 0-51. Unused bytes are set to X'00'. No information is supplied for printers.

If you define a length of 60 bytes, information is supplied on bytes 0-59. Unused bytes are set to X'00'.

If you define a length of 64 bytes, the information returned is complete (bytes 0-63). Unused bytes are set to X'00'.

Byte	Symb. name	Meaning
0	STAINFO	Status of the station
	(STAINFOY)	Terminal status message available
	(STAINFON)	No terminal status message available
1	STAINFP	Status of the station
	(STAINFPY)	Status message from the terminal is possible
	(STAINFPN)	No status message from the terminal is expected

Byte	Symb. name	Meaning
2-9	STAPTNAM	Printable terminal type
	TYP00	Unknown device type
	DSS-X.29	X.29 display terminal
	RECHNER	Terminal/computer
	SS-8102	8102 Printer
	DSS-8151	8151 Data Display Terminal
	DSS-8152	8152 Data Display Terminal
	SS-8110	8110 Printer Terminal
	SS-8121	8121 Printer
	FS100	100 Teleprinter
	FS100-E	100-E Teleprinter
	DRS90037	90037 Printer
	DRS-8122	8122 Printer
	DSS-8162	8162 Data Display Terminal
	DSS-8160	8160 Data Display Terminal
	DRS-8124	8124 Printer
	AP	Application program
	SST-X.29	X.29 printer terminal
	DSS-9750	9749 or 9750 Data Display Terminal
	DRS-9003	9003 Printer
	DSS-9770	9770 Data Display Terminal
	DRS-9002	9002 Printer
	DSS-3974	3974 Display Unit
	DSS-9751	9751 Data Display Terminal
	DSS-9752	9752 Data Display Terminal
	DSS-9753	9753 Data Display Terminal
	DRS-9001	9001 Printer
	DSS-9731	9731 Graphics Workstation
	DSS9770R	9770R Terminal
	DRS-9004	9004 Printer
	DSS-9754	9754 Data Display Terminal
	DSS-9755	9755 Data Display Terminal
	DSS-9763	9763 Data Display Terminal
	DRS-9012	9012 Printer
	DRS-9013	9013 Printer
	DSS-3270	3270 display terminal
	DRS-0131	9001-31 Printer
	DRS-0189	9001-8931 Printer
	DRS-9022	9022 Printer
	DRS-1118	9011-18 Printer
	DRS-1119	9011-19 Printer
	DRS-3287	3287 Printer
	TCP-IP	TCP-IP application
	DRS-9021	9021 Printer
	DRS-9014	9014 Printer
	DRS-9026	9026 Printer (HDLC, 9025-compatible)
	DSS-FE	Front-end data display terminal (FHS-DOORS)

Byte	Symb. name	Meaning
10	STAHCOPY	Local hardcopy unit
	(STABLHCY)	Local hardcopy unit connected
	(STABLHCN)	No local hardcopy unit connected
11	STAIDCR	ID card reader
	(STAIDCRY)	ID card reader connected
	(STAIDCRN)	No ID card reader connected
12	STACOL	Number of colors at the terminal
	(STACOLNO)	No colors
	(STACOL4)	4 colors
	(STACOL8)	8 colors
13-15	-	Reserved
16-19	STALINES	Physical number of lines printable (decimal) (taken from generation or default values)
20-23	STACOLUM	Physical number of characters per line printable (decimal) (taken from generation or default values)
24	STATTYPE	Station type
	(STATYPE7)	Terminal can operate in 7-bit mode only
	(STATYPE8)	Terminal can operate in 7-bit or 8-bit mode
25-32	STACURCH	Name of the extended standard character set. The value is supplied only if the data display terminal supports 8-bit mode.
33	STACCSNN	Number of 8-bit character sets supported
	(STATRINF)	The combination of X'00' and STATYPE8 means that the information to be output is longer than the specified length. The information will be truncated.
34	STACSS1	1st supported character set The variant number is stated in hexadecimal form.
35	STACSS2	2nd supported character set The variant number is stated in hexadecimal form.
36	STACSS3	3rd supported character set The variant number is stated in hexadecimal form.
37	STACSS4	4th supported character set The variant number is stated in hexadecimal form.

Byte	Symb. name	Meaning
38	STACSS5	5th supported character set The variant number is stated in hexadecimal form.
39	STACSS6	6th supported character set The variant number is stated in hexadecimal form.
40	STACSS7	7th supported character set The variant number is stated in hexadecimal form.
41	STACSS8	8th supported character set The variant number is stated in hexadecimal form.
42	STACSS9	9th supported character set The variant number is stated in hexadecimal form.
43	STACSS10	10th supported character set The variant number is stated in hexadecimal form.
44	STACSS11	11th supported character set The variant number is stated in hexadecimal form.
45	STACSS12	12th supported character set The variant number is stated in hexadecimal form.
46	STACSS13	13th supported character set The variant number is stated in hexadecimal form.
47	STACSS14	14th supported character set The variant number is stated in hexadecimal form.
48	STACSS15	15th supported character set The variant number is stated in hexadecimal form.
49	STACSS16	16th supported character set The variant number is stated in hexadecimal form.
50-51	-	reserved
52-59	STAACTCH	Name of the activated extended character set. Name is supplied only if the data display terminal supports 8-bit mode.

Byte	Symb. name	Meaning
60	STARMODE (STARMODM) (STARMODU)	Physical read mode Only modified fields are read. All unprotected fields are read.
61	STALLECH	Logical end-of-line character for terminals without the equivalent hardware functions.
62	STASUBCH	Substitute character for characters less than X'40' that are not logical control characters.
63	STAPERHC (STAPERHY) (STAPERHN)	Permanent hardcopy All output messages for a terminal are printed simultaneously on a connected hardcopy unit. No hardcopy log of output messages is printed.

If the value of STACCS_n (bytes 34-49) is such that fewer than 16 different character sets are supported, the remaining variant numbers (16-n) are set to X'00'.

*Examples of symbolic names of all dummy sections**Example TYPE=ALL*

```

DCSTA1  START
        DCSTA D,TYPE=ALL
1       IDLKG VER=050, ID=ALL, SECT=D, P=STA, ALIGN=D
2       *, VERSION 050
2 STAALL  DSECT
1 *
1 *           DEFINE TOTAL STATUS AREA
1 *
1 STASTALL DS  0CL64           TOTAL STATUS AREA
1 STAMNALL DS  0CL48           MINIMUM TOTAL STATUS AREA
1 *
1 *           DEFINE TOTAL STATUS AREA TERMINAL CHARACTERISTICS
1 *
1 STAALTCH DS  0D             TERMINAL CHARACTERISTICS AREA
1 *
1 *           DEFINE TERMINAL CHARACTERISTICS FIELDS
1 *
1 STASTTCH DS  0D             TERMINAL CHARACTERISTICS AREA
1 STAMNTCH DS  0D             MINIMUM TERMINAL CHARICS. AREA
1 *
1 STAPTTYP DC  AL1(0)         PARTNERTYPE
1 STADVTYPE DC  AL1(0)         DEVICE TYPE
1 STATCHR2 DC  AL1(0)         TERMINAL CHARACTERISTICS BYTE 2
1 STATCHR3 DC  AL1(0)         TERMINAL CHARACTERISTICS BYTE 3
1 STATCHR4 DC  AL1(0)         TERM. CHARACTERISTIC BYTE 4 901
1 STACHRS DC   AL1(0)         TERM. CHAR FROM STATION      920
1 STACTRLU DC  AL1(0)         CONTROL UNIT FOR PRINTER     701
1 STACHCAD DC  AL1(0)         CENTRAL HARDCOPY ADDRESS
1 *
1 *           DEFINE PARTNER TYPES (PTTYP)
1 *
1 STADCAMP EQU  X'00'         PARTNER IS A PROGRAM
1 STADCAMT EQU  X'01'         PARTNER IS A TERMINAL
1 *
1           DCDEVCH STA
2 *
2 *           DEFINE DEVICE TYPES (DVTYP)
2 *
2 STAD8103 EQU  X'02'         TELETYPE 8103
2 STAD8150 EQU  X'04'         VIDEO TERMINAL 8150
2 STAD8153 EQU  X'05'         *NO VTSU* VIDEO TERMINAL 8153
2 STADHOST EQU  X'08'         INTELLIGENT PARTNER
2 STAD8151 EQU  X'15'         VIDEO TERMINAL 8151
2 STAD8152 EQU  X'16'         VIDEO TERMINAL 8152
2 STAD8110 EQU  X'17'         SS-8110
2 STAD6154 EQU  X'18'         *NO VTSU* VIDEO 8161 54 CHAR PER LINE
2 STAD6164 EQU  X'19'         *NO VTSU* VIDEO 8161 64 CHAR PER LINE
2 STAD6180 EQU  X'1A'         *NO VTSU* VIDEO 8161 80 CHAR PER LINE
2 STAD8161 EQU  X'1A'         *NO VTSU* VIDEO 8161
2 STAD8121 EQU  X'1C'         PRINTER STATION 8121
2 STADPT80 EQU  X'1D'         *AS 8103* TELETYPE PT80
2 STAD1000 EQU  X'1E'         *AS 8103* TELETYPE T1000
2 STADT100 EQU  X'23'         *AS 8103* TELETYPE T100
2 STAD100E EQU  X'26'         *AS 8103* FS100-E

```

```

2 STAD8122 EQU X'2B' PRINTER STATION 8122
2 STAD8162 EQU X'2C' VIDEO 8162
2 STAD8160 EQU X'2D' VIDEO 8160
2 STAD8124 EQU X'2E' PRINTER STATION 8124
2 STAD8167 EQU X'2F' *AS 8160* VIDEO 8167
2 STADAP EQU X'30' *AS HOST* AP-STATION
2 STAD9750 EQU X'35' VIDEO 9750 OR 9749
2 STAD9003 EQU X'36' PRINTER STATION 9003
2 STAD9770 EQU X'39' *AS 8151* DS 9770
2 STAD9002 EQU X'3B' PRINTER STATION 9002
2 STAD3974 EQU X'3D' VIDEO TERMINAL 3974
2 STAD9751 EQU X'3F' *AS 8160* DSS 9751
2 STAD9752 EQU X'40' *AS 9750* DSS 9752
2 STAD9753 EQU X'41' *AS 9750* DSS 9753
2 STAD9001 EQU X'42' PRINTER 9001
2 STAD9731 EQU X'43' *AS 3974* GRAFIC STATION 9731
2 STAD9004 EQU X'45' PRINTER 9004
2 STAD9754 EQU X'4C' *AS 8160* VIDEO 9754
2 STAD9755 EQU X'4E' DSS-9755 922
2 STAD9763 EQU X'4F' DSS-9763
2 STADBTXF EQU X'55' *AS HOST* BTX-STATION T-3000 (FELDVERS.)
2 STADBTXE EQU X'56' *AS HOST* BTX-EDITIER-STATION (DIENST)
2 STADBTXA EQU X'57' *AS HOST* BTX-ABFRAGE-STATION (DIENST)
2 STADUTC EQU X'5A' UTC FUER TELETEx
2 STAD9012 EQU X'5B' PRINTER 9012
2 STAD9013 EQU X'5C' PRINTER 9013
2 STAD3270 EQU X'5E' DSS-3270
2 STAD0131 EQU X'65' PRINTER 9001-31
2 STAD0189 EQU X'66' PRINTER 9001-8931
2 STAD9022 EQU X'68' PRINTER 9022
2 STAD1118 EQU X'6B' PRINTER 9011-18
2 STAD1119 EQU X'6C' PRINTER 9011-19
2 STAD3287 EQU X'6E' PRINTER 3287 956
2 STADPCL EQU X'70' PRINTERS PCL 960
2 STAD9021 EQU X'70' PRINTERS 9021 / 9022-200, HP LJ960
2 STAD9014 EQU X'72' PRINTER 9014 001
2 STAD9026 EQU X'73' PRINTER 9026 (HDLC,COMP.9025) 001
2 STADFE EQU X'78' DSS-FE BOF-FRONT-END 999
2 *
2 * DEFINE TERMINAL CHARACTERISTICS BYTE 2 (TCHR2) BITS
2 *
2 STATC2EX EQU 8 SECONDARY CHARACTER SET
2 STATC2LC EQU 32 LOWER CASE
2 STATC2DT EQU 64 GERM KEYB WITH GERM NAT CHAR901
2 STATC2DF EQU 128 BYTE 2 DEFINED
2 *
2 * DEFINE TERMINAL CHARACTERISTICS BYTE 3 (TCHR3) BITS
2 *
2 STATC3H1 EQU 1 HARDCOPY BIT 1 (LOCAL)
2 STATC3H2 EQU 2 HARDCOPY BIT 2 (CENTRAL)
2 STATC3HC EQU 3 HARDCOPY BITS
2 STATC3IC EQU 4 IDENTITY CARD READER
2 STATC3FD EQU 8 FLOPPY DISK
2 STATC3AP EQU 16 APL CAPABILITY
2 STATC3GF EQU 32 GRAPHICS
2 STATC3DZ EQU 64 DEZENTRAL FORMATING
2 STATC3DF EQU 128 BYTE 3 DEFINED
2 *

```

```

2 *          DEFINE TERMINAL CHARACTERISTICS BYTE 4 (TCHR4) BITS
2 *
2 STATC4CO EQU 1          4 COLOURS (ITALIC/HALFBRIGHT)
2 STATC4ZF EQU 2          NEW ZAT AND FAT POSSIBLE          920
2 STATC4ST EQU 4          STATUS QUERY POSSIBLE            920
2 STATC4HI EQU 8          HARDWARE INFOLINE AVAILABLE      920
2 STATC4C8 EQU 16         8 COLOURS                          953
2 STATC4HP EQU 32         HP LASER JET II                    954
2 STATC4DF EQU 128        BYTE 4 DEFINED
2 *
2 *          DEFINE TERM CHAR FROM STATION  BYTE  (TCHRS) BITS  920
2 *
2 STATCSDT EQU 1          GERMAN KEYBOARD                    920
2 STATCSHC EQU 2          LOCAL HARDCOPY PRINTER            920
2 STATCSIC EQU 4          ID-CARD RAEDER                     920
2 STATCSDF EQU 128        TERM CHAR FROM STAT RECEIVED     920
2          *,DCDEVCH    001    920406    53113038
1 *
1 *
1 *          DEFINE TOTAL STATUS AREA PHYSICAL TERMINAL ATTRIBUTES
1 *
1 STAALPV DS 0D          PHYSICAL TERMINAL ATTR. AREA
1 *
1 *          DEFINE PHYSICAL TERMINAL ATTRIBUTES FIELDS
1 *
1 STASTPV DS 0D          PHYSICAL TERMINAL ATTR. AREA
1 STAMNPV DS 0D          MINIMUM PHYS. TERM. ATTR. AREA
1 *
1 STALLEN DC H'0'        PHYSICAL LINE LENGTH
1 STANOLIN DC H'0'        PHYSICAL NUMBER OF LINES
1 STAMAXDB DC H'0'        MAX. PHYSICAL DEVICE BUFFER
1          DC 2AL1(0)     RESERVED FOR FUTURE DEVELOPMENT
1 *
1 *          DEFINE TOTAL STATUS AREA VIRTUAL TERMINAL ATTRIBUTES
1 *
1 STAALLV DS 0D          VIRTUAL TERMINAL ATTR. AREA
1 *
1 *          DEFINE VIRTUAL TERMINAL ATTRIBUTES FIELDS
1 *
1 STASTLV DS 0D          VIRTUAL TERMINAL ATTR. AREA
1 STAMNLV DS 0D          MINIMUM VIRTUAL TERM ATTR AREA
1 *
1 STALLEN DC H'0'        VIRTUAL LINE LENGTH
1 STALNOLN DC H'0'        VIRTUAL NUMBER OF LINES
1 STALMAXB DC H'0'        MAXIMUM VIRTUAL DEVICE BUFFER
1          DC 2AL1(0)     RESERVED FOR FUTURE DEVELOPMENT
1 *
1 *          DEFINE TOTAL STATUS AREA VIRTUAL DEVICE TYPE AREA
1 *
1 STAALVDT DS 0D          VIRTUAL DEVICE TYPE AREA
1 *
1 *          DEFINE VIRTUAL DEVICE TYPE FIELDS
1 *
1 STASTVDT DS 0D          VIRTUAL DEVICE TYPE AREA
1 STAMNVDT DS 0D          MINIMUM VIRTUAL DEV TYPE AREA
1 *
1 STAVDT DC AL1(0)        VIRT. DVTYP BYTE 0                901
1 STAVDTPR DC AL1(0)     VIRT. DVTYP BYTE 1                901

```

```

1          DC      6AL1(0)          RESERVED          901
1 *
1 *          VIRTUAL DEVICE TYPE (STAVDT) BITS
1 *
1 STALINCP EQU    1          LINE MODE CAPABILITY
1 STAFORCP EQU    2          FORM MODE CAPABILITY
1 STACMPCP EQU    4          COMP MODE CAPABILITY
1 STAFYSCP EQU    8          PHYS MODE CAPABILITY
1 STAEXLCP EQU   16          EXT.LINE MODE CAPABILITY    901
1 STAAUTLF EQU   32          AUTOMATIC LF FOR PRINTER    952
1 STAEOM EQU     64          EVANESCENT OUTPUT MESSAGES
1 STANOINP EQU  128          OUTPUT ONLY DEVICE
1 *
1 *          VIRTUAL DEVICE DATASTREAM (STAVDTPR) BITS          901
1 *
1 STATD810 EQU    1          810 DATASTREAM          901
1 STAT3270 EQU    2          3270 DATASTREAM          901
1 *
1 *          DEFINE TOTAL STATUS AREA STATIC EDIT OPTIONS AREA
1 *
1 STAALEDT DS     0D          STATIC EDIT OPTIONS AREA
1 *
1 *          DEFINE STATIC EDIT OPTIONS FIELDS
1 *
1 STASTEDT DS     0D          STATIC EDIT OPTIONS AREA
1 STAMNEDT DS     0D          MINIMUM STATIC EDIT OPTS. AREA
1 *
1 STAEDIT DS      0A          STATIC EDIT OPTIONS
1 STASEWR1 DC     AL1(0)      STATIC WRITE EDIT OPTION BYTE 1
1 STASEWR2 DC     AL1(0)      STATIC WRITE EDIT OPTION BYTE 2
1 STASERD1 DC     AL1(0)      STATIC READ EDIT OPTION BYTE 1
1 STASERD2 DC     AL1(0)      STATIC READ EDIT OPTION BYTE 2
1          DC      4AL1(0)      RESERVED FOR FUTURE DEVELOPMENT
1 *
1 *          OUTPUT EDIT OPTION BYTE 1 (STASEWR1) BITS
1 *
1 STAWR1CD EQU    1          CODE TRANSLATION
1 STAWR1LE EQU    2          LINE END TREATMENT
1 STAWR1RE EQU    8          RESET CONTROL
1 STAWR1HO EQU   16          HOMOGENEOUS OUTPUT
1 STAWR1PT EQU   32          PAPER TAPE CONTROL
1 STAWR1HC EQU  128          HARDCOPY
1 *
1 STAWR1M1 EQU    4          MODE BIT 1
1 STAWR1M2 EQU   64          MODE BIT 2
1 STAWR1MM EQU   68          MODE MASK
1 STAWR1LI EQU    4          LINE MODE
1 STAWR1FO EQU   64          FORMAT MODE
1 STAWR1CO EQU    0          COMPATIBLE MODE
1 STAWR1FY EQU   68          PHYSICAL MODE
1 *
1 *          OUTPUT EDIT OPTION BYTE 2 (STASEWR2) BITS
1 *
1 STAWR2HD EQU    1          HEADER PRESENT
1 STAWR2NO EQU    2          NO LOGICAL CHARS INTERPRET    801
1 STAWR2EX EQU    4          EXTENDED LINE MODE          701
1 STAWR2IM EQU    8          INFORMATIVE MESSAGE
1 STAWR2RB EQU   16          RETURN INFORMATION REQUIRED

```

```

1 STAWR2PN EQU 32 POSITIONING NOT REQUIRED
1 STAWR2TP EQU 32 TRANSPARENT CODE
1 STAWR2BL EQU 64 BEL INDICATED
1 STAWR2ET EQU 128 USE ETB
1 *
1 * INPUT EDIT OPTION BYTE 1 (STASERD1) BITS
1 *
1 STARD1CD EQU 1 CODE TRANSLATION
1 STARD1LE EQU 2 LINE END TREATMENT
1 STARD1BS EQU 4 BACKSPACE CONTROL
1 STARD1PT EQU 8 PAPER TAPE CONTROL
1 STARD1LC EQU 16 LOWER CASE CONTROL
1 STARD1HD EQU 128 HEADER REQUIRED
1 *
1 STARD1M1 EQU 32 MODE BIT 1
1 STARD1M2 EQU 64 MODE BIT 2
1 STARD1MM EQU 96 MODE MASK
1 STARD1CO EQU 0 COMPATIBLE MODE
1 STARD1LI EQU 32 LINE MODE
1 STARD1FO EQU 64 FORMAT MODE
1 STARD1FY EQU 96 PHYSICAL MODE
1 *
1 * INPUT EDIT OPTIONS BYTE 2 (STASERD2) BITS
1 *
1 STARD2FC EQU 1 FUNCTION CODE REQUIRED
1 STARD2CF EQU 4 CONFIDENTIAL INPUT EXPECTED
1 STARD2IC EQU 8 INPUT FROM CARD READER
1 STARD2EX EQU 32 EXTENDED LINE MODE 701
1 *
1 * DEFINE TOTAL STATUS AREA OVERFLOW CONTROL BYTE AREA
1 *
1 STAALOF1 DS 0D OVERFLOW CONTROL BYTE AREA
1 *
1 * DEFINE OVERFLOW CONTROL BYTE AREA FIELDS
1 *
1 STASTOFL DS 0D OVERFLOW CONTROL BYTE AREA
1 STAMNOFL DS 0D MINIMUM OVERFLOW CTL. BYTE AREA
1 *
1 STAOFLOW DC AL1(0) OVERFLOW CONTROL BYTE
1 STAOFTIM DC AL1(0) OVERFLOW TIME IN SECONDS 890
1 *
1 DC 6AL1(0) Reserved for future development
1 *
1 * OVERFLOW CONTROL BYTE (STAOFLOW) BITS
1 *
1 STAOFCTM EQU 1 OVERFLOW CONTROL BY TIME
1 STAOFACK EQU 2 OVERFLOW CONTROL BY ACKNOWL.
1 STAOFCAK EQU 2 OVERFLOW CONTROL BY ACKNOWL.
1 STAOFCNP EQU 4 AUTOMATIC NEW PAGE 801
1 STAOFCTL EQU 7 OVERFLOW CONTROL TEST
1 STAOFPGM EQU 32 OVERFLOW CONTROL BY USER PGM
1 *
1 * DEFINE TOTAL STATUS AREA STATION NAME AREA
1 *
1 STAALSTN DS 0D STATION NAME AREA
1 *
1 * DEFINE STATION NAME AREA FIELD

```



```
1 *
1 STASTSTN DS      0D              STATION NAME AREA
1 STAMNSTN DS      0D              MINIMUM STATION NAME AREA
1 *
1 STASTNAM DC      CL8 '          ' STATION NAME
1 *
1 *
1 *
1 *              DEFINE TOTAL STATUS AREA PROCESSOR NAME AREA
1 *
1 STAALPRN DS      0D              PROCESSOR NAME AREA
1 STALPDT EQU      8
1 *
1 *              DEFINE PROCESSOR NAME AREA FIELDS
1 *
1 STASTPRN DS      0D              PROCESSOR NAME AREA
1 STAMNPRN DS      0D              MINIMUM PROCESSOR NAME AREA
1 *
1 STAPRNAM DC      CL8 '          ' PROCESSOR NAME
1 *
1 *              *,DCSTA      050      930203      53531014
1 *
1 *              END
```

Example TYPE=BASIC

```

      BDCSTA  START
DCSTA  D,TYPE=BASIC
1      IDLKG  VER=050, ID=BASIC, SECT=D, P=STA, ALIGN=D
2
2      STABASIC DSECT
1      *
1      *          DEFINE BASIC TERMINAL INFORMATION AREA
1      *
1      STASTBAS DS      0D                                          920
1      *
1      STAINFO  DC      AL1(0)          STATUS INFO CONTROL BYTE    920
1      STAINFOY EQU     C'Y'           STATUS FROM TERMINAL PRES  920
1      STAINFON EQU     C'N'           NO STATUS INFO FROM TERMINAL
1      *
1      STAINFP  DC      AL1(0)          INFO FROM TERM CTRL BYTE    920
1      STAINFPY EQU     C'Y'           STATUS FROM TERMINAL POSSIBLE
1      STAINFPN EQU     C'N'           NO STATUS FROM TERMINAL POS 920
1      *
1      STAPTNAM DC      8AL1(0)        PRINTABLE TERMINAL TYPE    920
1      *
1      STAHCOPY DC      AL1(0)        LOCAL HARDCOPY CTRL BYTE    920
1      STABLHCY EQU     C'Y'          LOCAL HARDCOPY AVAILABLE    920
1      STABLHCN EQU     C'N'          LOCAL HARDCOPY NOT AVAIL    920
1      *
1      STAIDCR  DC      AL1(0)        ID-CARD READER CTRL BYTE    920
1      STAIDCRY EQU     C'Y'          ID-CARD READER AVAILABLE    920
1      STAIDCRN EQU     C'N'          ID-CARD READER NOT AVAIL    920
1      *
1      STACOL   DC      AL1(0)        COLOURS CONTROL BYTE      920
1      STACOLNO EQU     C'N'          NO COLOURS                 920
1      STACOL4  EQU     C'4'          4 COLOURS                 920
1      STACOL8  EQU     C'8'          8 COLOURS                 920
1      *
1      DC      AL3(0)          RESERVED                          920
1      *
1      STALINES DC      AL4(0)        MAX PHYSICAL LINES        920
1      *
1      STACOLUM DC      AL4(0)        MAX PHYSICAL COLUMNS     920
1      *
1      STABASML EQU     24           STATUS BASIC INF MIN LEN    920
1      *
1      STATTYPE DC      AL1(0)        Terminal type 7 or 8-BIT    001
1      STATYPE7 EQU     C'7'          Terminal type 7-bit        001
1      STATYPE8 EQU     C'8'          Terminal type 8-bit        001
1      *
1      STACURCH DC      CL8'          Current character set name  001
1      *
1      STATTPM  EQU     *-STASTBAS    Minimum length for current  001
1      *          Char set and terminal type 001
1      *
1      STACCSNN DC      AL1(0)        Number of supported        001
1      *          Character set          001
1      STACCS1  DC      AL1(0)        1st supported character set 001
1      STACCS2  DC      AL1(0)        2nd supported character set 001
1      STACCS3  DC      AL1(0)        3rd supported character set 001
1      STACCS4  DC      AL1(0)        4th supported character set 001

```

```

1 STACCS5 DC AL1(0) 5th supported character set 001
1 STACCS6 DC AL1(0) 6th supported character set 001
1 STACCS7 DC AL1(0) 7th supported character set 001
1 STACCS8 DC AL1(0) 8th supported character set 001
1 STACCS9 DC AL1(0) 9th supported character set 001
1 STACCS10 DC AL1(0) 10th supported character set 001
1 STACCS11 DC AL1(0) 11th supported character set 001
1 STACCS12 DC AL1(0) 12th supported character set 001
1 STACCS13 DC AL1(0) 13th supported character set 001
1 STACCS14 DC AL1(0) 14th supported character set 001
1 STACCS15 DC AL1(0) 15th supported character set 001
1 STACCS16 DC AL1(0) 16th supported character set 001
1 *
1 DC AL2(0) Reserved 001
1 *
1 STACCSML EQU *-STASTBAS Minimum length for extended 001
1 * info with all character set 001
1 *
1 STAACTCH DC CL8' ' Activated variant 002
1 *
1 STAACTML EQU *-STASTBAS Minimum length for info 002
1 * with activated variant 002
1 *
1 STARMODE DC AL1(0) Operating mode (M or U) 050
1 STARMODM EQU C'M' Operating mode = 'Modified' 050
1 STARMODU EQU C'U' Operating mode = 'Unprotected'
1 *
1 STALLECH DC AL1(0) Logical line end character 050
1 *
1 STASUBCH DC AL1(0) Substitute character 050
1 *
1 STAPERHC DC AL1(0) Permanent hardcopy (Y or N) 050
1 STAPERHY EQU C'Y' Perm. hardcopy requested 050
1 STAPERHN EQU C'N' Perm. hardcopy not requested 050
1 *
1 STAOLSPL EQU 64 Minimum length for info 050
1 * about Operating mode, Logical-
1 * line-end, Substitute char and
1 * Permanent hardcopy 050
1 * ,DCSTA 050 930203 53531014
1 END

```

Example TYPE=MONCS

```

DCSTA3   START
         DCSTA D,TYPE=MONCS
1        IDLKG VER=050, ID=MONCS, SECT=D, P=STA, ALIGN=D
2        *, VERSION 050
2 STAMONCS DSECT
1 *
1 *           DEFINE MONITOR AND CHARACTER SET AREA
1 *
1 STASTMOC DS    0D                                     920
1 *
1 STAMOCPR DC    AL1(0)                               STATUS INFO CONTROL BYTE 920
1 STAMOTYP DC    AL1(0)                               TYPE OF MONITOR          920
1 STAFAT   DC    AL1(0)                               NEW FIELD ATTR CTRL BYTE 920
1         DC    AL1(0)                               RESERVED                 920
1 STADIM1  DC    AL1(0)                               DIM 24X80 CONTROL BYTE   920
1 STADIM2  DC    AL1(0)                               DIM 32X80 CONTROL BYTE   920
1 STADIM3  DC    AL1(0)                               DIM 43X80 CONTROL BYTE   920
1 STADIM4  DC    AL1(0)                               DIM 27X132 CONTROL BYTE  920
1         DC    AL4(0)                               RESERVED                 920
1 STACSNUM DC    AL2(0)                               NUMBER OF USABLE CHAR SETS 920
1 STACS0T  DC    AL1(0)                               TYPE OF CHARACTER SET 0   920
1 STACS0S  DC    AL1(0)                               STATUS OF CHARACTER SET 0 920
1 STACS1T  DC    AL1(0)                               TYPE OF CHARACTER SET 1   920
1 STACS1S  DC    AL1(0)                               STATUS OF CHARACTER SET 1 920
1 STACS2T  DC    AL1(0)                               TYPE OF CHARACTER SET 2   920
1 STACS2S  DC    AL1(0)                               STATUS OF CHARACTER SET 2 920
1 STACS3T  DC    AL1(0)                               TYPE OF CHARACTER SET 3   920
1 STACS3S  DC    AL1(0)                               STATUS OF CHARACTER SET 3 920
1 STACS4T  DC    AL1(0)                               TYPE OF CHARACTER SET 4   920
1 STACS4S  DC    AL1(0)                               STATUS OF CHARACTER SET 4 920
1 STACS5T  DC    AL1(0)                               TYPE OF CHARACTER SET 5   920
1 STACS5S  DC    AL1(0)                               STATUS OF CHARACTER SET 5 920
1 STACS6T  DC    AL1(0)                               TYPE OF CHARACTER SET 6   920
1 STACS6S  DC    AL1(0)                               STATUS OF CHARACTER SET 6 920
1 STACS7T  DC    AL1(0)                               TYPE OF CHARACTER SET 7   920
1 STACS7S  DC    AL1(0)                               STATUS OF CHARACTER SET 7 920
1 *
1 STAMOCML EQU    14                                STATUS MONCS INF MIN LEN  920
1 *
1 *           STATUS INFO CONTROL BITS                920
1 *
1 STAMOCY  EQU    C'Y'                               STATUS FROM TERMINAL PRES 920
1 STAMOCN  EQU    C'N'                               NO STATUS INFO FROM TERMINAL
1 *
1 *           TYPE OF MONITOR BITS                    920
1 *
1 STAMONO  EQU    C'M'                               MONOCHROM SCREEN         920
1 STACOLOR EQU    C'C'                               SCREEN WITH COLOURS      920
1 STAPRINT EQU    C'P'                               PRINTER                  920
1 *
1 *           NEW FIELD ATTR CONTROL BITS              920
1 *
1 STAFATY  EQU    C'Y'                               NEW FIELD ATTR POSSIBLE  920
1 STAFATN  EQU    C'N'                               NEW FIELD ATTR NOT POSSIBL 920
1 *
1 *           DIMENSION CONTROL BITS (&P.DIM1 - &P.DIM4) 920

```

```
1 *
1 STADIMY EQU C'Y' DIMENSION AVAILABLE 920
1 STADIMN EQU C'N' DIMENSION NOT AVAILABLE 920
1 *
1 * TYPE OF CHARACTER SET BITS 920
1 *
1 STACSSIN EQU C'S' SINGLE PLANE SET 920
1 STACSTRI EQU C'T' TRIPLE PLANE SET 920
1 STACSNO EQU C'N' NOT LOADABLE CHAR SET 920
1 *
1 * STATUS OF CHARACTER SET BITS 920
1 *
1 STACSNLO EQU C'0' CHAR SET NOT LOADED 920
1 STACSDSS EQU C'1' FROM DSS LOADED + ASSIGNED 920
1 STACSDVN EQU C'2' FROM DVA LOADED 920
1 STACSDVA EQU C'3' FROM DVA LOADED + ASSIGNED 920
1 *,DCSTA 050 930203 53531014
END
```

Example TYPE=PERPH

```

DCSTA4  START
DCSTA  D,TYPE=PERPH
1      IDLKG VER=050, ID=PERPH, SECT=D, P=STA, ALIGN=D
2      *,VERSION 050
2 STAPERPH DSECT
1 *
1 *          DEFINE PERIPHERIE OF TERMINAL AREA
1 *
1 STASTPER DS      0D                                920
1 *
1 STAPERPR DC     AL1(0)          STATUS INFO CONTROL BYTE  920
1          DC     AL2(0)          RESERVED                    954
1 STALOCHC DC     AL1(0)          LOCAL HARDCOPY CTRL BYTE  920
1          DC     AL2(0)          RESERVED                    920
1 STAIDCAR DC     AL1(0)          ID-CARD READER CTRL BYTE  920
1 STACKT  DC     AL1(0)          PIN-PAD CONTROL BYTE       920
1 *
1 STAPERML EQU    8              STATUS PERPH INF MIN LEN    920
1 *
1 *          STATUS INFO CONTROL BITS                        920
1 *
1 STAPERY  EQU    C'Y'           STATUS FROM TERMINAL PRES  920
1 STAPERN  EQU    C'N'           NO STATUS INFO FROM TERMINAL
1 *
1 *          LOCAL HARDCOPY CONTROL BITS                      920
1 *
1 STALHCY  EQU    C'Y'           LOCAL HARDCOPY AVAILABLE    920
1 STALHCN  EQU    C'N'           LOCAL HARDCOPY NOT AVAIL    920
1 *
1 *          ID-CARD READER CONTROL BITS                      920
1 *
1 STAIDCY  EQU    C'Y'           ID-CARD READER AVAILABLE    920
1 STAIDCN  EQU    C'N'           ID-CARD READER NOT AVAIL    920
1 *
1 *          PIN-PAD CONTROL BITS                              920
1 *
1 STACKTY  EQU    C'Y'           PIN PAD AVAILABLE            920
1 STACKTN  EQU    C'N'           PIN PAD NOT AVAILABLE        920
1          *,DCSTA      050      930203      53531014
END

```

3.2 Logical control characters: VTCSET

Logical control characters for line and page terminals can also be specified in symbolic form. The VTCSET macro converts logical control characters into device-specific control characters. Logical control characters can be used only if line mode or extended line mode is set at the input/output interface or a VTSUCB is assigned with MODE=LINE, EXTEND or INFO.

The operands COL, EXT RPT, EXT DIS and EXT FLD are new, as is the combination of the VPA and HPA operands for data display terminals.

Operation	Operands
VTCSET	prefix

prefix Character string to be prefixed to the symbolic name; it may be up to 5 characters in length.

In the explanations, '&P.' is used as the prefix instead of the character string prefixed to the symbolic name VTCSET.

1. Logical record control characters

&P.NL logical end of line (new line)

Effect on output:

- Special display formats on the terminal are reset to the standard display (normal, standard color, unprotected in line mode, protected in extended line mode). The character set is not reset. If the message is output in extended line mode, the remainder of the line is blanked and protected.
- The defined logical end-of-line character is output (except in extended line mode).
- The next line is set to standard status: normal, low intensity, standard color, standard character set, unprotected in line mode, protected in extended line mode. The standard character set is character set 0 for 9763 Data Display Terminals and character set 0 for other display terminals and printers.
Status is not set to standard if operating mode 2 (see page 189) and HOM=YES were specified for the terminal.

- If continuing the output would cause data overflow at the terminal, the defined overflow check is executed (except in extended line mode).
- The cursor is positioned at the start of the next line.
- In the case of field-oriented display and structured output (standard) a start of field is generated.

&P.NP Logical end of page (new page)

Effect on output:

- Special terminal display formats are reset to the standard format (normal, standard character set, standard color, unprotected in line mode, protected in extended line mode with UPDATE=NO, unprotected in extended line mode with UPDATE=YES). The screen format is reset to 24x80. The standard character set is character set 0 for 9763 Data Display Terminals and character set 0 for other display terminals and printers.
- The defined logical end-of-line character is output.
- Hardcopy output is initiated (if HCOPY=YES).
- The defined overflow control action is executed.
- A new page is created (on display terminals the screen is cleared and the screen format 24x80 set, on printer terminals a page feed is performed). If page feeds were already initiated in the same message by means of ASF (automatic sheet feed), then the last ASF used replaces NP (for printers only).
- The cursor is positioned at start of line.
- In the case of field-oriented display and structured output (standard) a start of field is generated.

&P.CL Logical end of record (current line)

Effect on output (printers and teleprinters only):

- Special forms of display on the terminal are reset (to normal, standard character set, unprotected). The standard character set is character set 1 for the devices concerned.
- The defined logical end-of-line character is output.
- The cursor is set to the start of the current line.

&P.VPAddd (data display terminals only, see page 70 for printers)

Position on first unprotected field of a line (vertical position absolute)
(binary or three-digit decimal value)

Effect on output:

Absolute line positioning on the first unprotected field in line ddd of the data display terminal. Can be combined with HPA for simultaneous positioning on the absolute column.

Note

- If ddd is zero or larger than 255, the substitute character (SUB) is inserted instead of VPA and ddd is output.
- If ddd is less than or equal to 255 but greater than the maximum number of lines, NL is executed instead of VPA.

&P.HPAddd (data display terminals only, see page 70 for printers)

Position on column (horizontal position absolute) (three-digit decimal value)

HPA is processed in extended line mode only and only when specified immediately after a valid VPA. HPA specifies the absolute column in a line defined beforehand by VPA. If the VPA preceding HPA was invalid or if HPA is not specified immediately after VPA, HPA ddd is ignored. HPA on a 3270 terminal positions on the start of the next unprotected field.

Effect on output:

Absolute column positioning in the line defined beforehand by the VPA.

Note

- If ddd is zero or larger than 255, the substitute character (SUB) is inserted instead of HPA and ddd is output.
- If ddd is less than or equal to 255 but greater than the maximum number of columns, HPA ddd is ignored and only the VPA function is executed.
- In VTSU versions < 11 the combination of VPA and HPA for data display terminals is ignored.

2. Logical display control characters

&P.EM1 Emphasized layout 1)

Effect on output:

Subsequent text characters are "emphasized" (highlighted) on the terminal as appropriate to the device type used (see table on page 75).

Emphasized layout is reset by:

- the logical record control characters (NL, NP, HPA, VPA)
- the display control characters (EM2, EM3, EM4, DAR, DIS, NOR)
- the field control characters (EPA, SPA, NUM, CHS, COL, FLD)

&P.EM2 Emphasized layout 2

Effect on output:

Subsequent text characters are "emphasized" on the terminal as appropriate to the device type used (see table on page 75).

Emphasized layout is reset by:

- the logical record control characters (NL, NP, HPA, VPA)
- the display control characters (EM2, EM3, EM4, DAR, DIS, NOR)
- the field control characters (EPA, SPA, NUM, CHS, COL, FLD)

&P.EM3 Emphasized layout 3

Effect on output:

Subsequent text characters are "emphasized" on the terminal as appropriate to the device type used (see table on page 75).

Emphasized layout is reset by:

- the logical record control characters (NL, NP, HPA, VPA)
- the display control characters (EM2, EM3, EM4, DAR, DIS, NOR)
- the field control characters (EPA, SPA, NUM, CHS, COL, FLD)

&P.EM4 Emphasized layout 4

Effect on output:

Subsequent text characters are "emphasized" on the terminal as appropriate to the device type used (see table on page 75).

Emphasized layout is reset by:

- the logical record control characters (NL, NP, HPA, VPA)
- the display control characters (EM2, EM3, EM4, DAR, DIS, NOR)
- the field control characters (EPA, SPA, NUM, CHS, COL, FLD)

- &P.NOR** Normal layout
- Effect on output:
Subsequent text characters are displayed normally on the terminal (i.e. emphasized layout is reset).
- &P.SO** Shift out into character set extension in accordance with the table on page 78 (Shift Out Into Character Set Extension).
- Effect on output on certain terminals only (see table on page 78):
Subsequent text characters are displayed on the terminal in the device-specific character set according to the table on page 78.
- Meaning for input:
Subsequent text characters were entered using the terminal-specific character set.
- &P.SI** Shift into basic character set in accordance with the table on page 78.
- Effect on output:
Subsequent text characters are displayed on the terminal in the basic character set.
- Meaning for input:
Subsequent text characters belong to the basic character set.
- &P.DAR** Dark layout
- Support for this operand is continued only for the sake of compatibility. Instead of &P.DAR, you should use the operands &P.EXT DIS and &P.EXT FLD.
- Depending on the value of the DARPRINTABLE parameter, DAR is either a field or a logical display control character.
- Effect on output:
The value of DARPRINTABLE is 'N' (default).
In the case of field-oriented display and structured output DAR generates a start of field. Subsequent text characters are blanked and cannot be printed. The field is omitted from hardcopy.
- DAR is reset by the logical record control characters (NL, NP, HPA, VPA) or the field control characters (EPA, NUM, CHS, COL and FLD).
- The value of DARPRINTABLE is 'Y'.
Subsequent text characters are blanked.
No start of field is generated.

3. Logical field control characters

&P.SPA Start protected area

Effect on output (data display terminals only):

In the case of field-oriented display and structured output, SPA generates a start of field. Subsequent text characters are displayed at reduced intensity on the monitor of the data display terminal and are protected, i.e. they cannot be overwritten and returned to the data processing unit.

SPA is reset by the logical record control characters (NL, NP, HPA, VPA) or the field control characters (EPA, NUM, CHS, COL and FLD).

Note

This function has a significant impact on the display attributes of certain terminals (newline on 8152 DDT, hardcopy function on 816x, 975x and 976x terminals). It should therefore be used with extreme caution.

&P.EPA End protected area

Effect on output:

In the case of field-oriented display and structured output, EPA generates a start of field. Subsequent text characters are output to the terminal unprotected and displayed at high intensity.

EPA is reset by the logical record control characters (NL, NP, HPA, VPA) or by the field control characters (EPA, NUM, CHS, COL and FLD).

&P.NUM Numeric area

Effect on output:

In the case of field-oriented display and structured output, NUM generates a start of field. Subsequent text characters are output to the terminal unprotected and displayed at high intensity.

You can enter only numeric data (digits, . * / + -) in this field.

NUM is reset by the logical record control characters (NL, NP, HPA, VPA) or by the field control characters (EPA, SPA, CHS, COL and FLD).

&P.CHS dd

Loadable character set

CHS is effective only on terminals of type 9763.

Effect on output:

In the case of field-oriented display and structured output, CHS generates a start of field and a loadable character set from the repertory available to the data terminal is selected for this field. dd is a two-digit decimal in the range 00-07 and designates the desired loadable

character set. You can use the TSTAT/YINQUIRE macro to query the loadable character sets available.

Example

If dd is 00, you address loadable character set 0 with the symbolic name STACS0T in DCSTA (see MONCS).

Only characters from the selected character set can be entered in this field.

CHS is reset by:

- the logical record control characters (NL, NP, HPA, VPA)
- the display control characters (EM1, EM2, EM3, EM4, DAR, NOR, DIS)
- the field control characters (EPA, SPA, NUM, COL, FLD)

Note

The user must ensure that the character sets of the data display terminal that are to be addressed logically via the control character CHS are loaded with the correct character sets.

CHS is not supported in the system line.

The character sets can be loaded physically in "phys" mode or with EXT TRA in line mode. Character sets can be created with the software product ICE.

&P.COL dd

Choice of colors

COL is effective only on color terminals of type 9763.

Effect on output:

In the case of field-oriented display and structured output, COL generates a start of field (even on monochrome screens). If the terminal is a 9763 color DDT, COL can be used to select a color from those available for this field. dd is a two-digit decimal in the range 00-07 designating the desired color.

00	default	04	green
01	blue	05	cyan
02	red	06	yellow
03	magenta	07	white

Note that COL is not supported in the system line.

COL (color selected) is reset by:

- the logical record control characters (NL, NP, HPA, VPA)
- the display control characters (EM1, EM2, EM3, EM4, DAR, NOR, DIS)
- the field control characters (EPA, SPA, NUM, CHS, FLD)

&P.EXT DIM ll ccc

Physical screen dimension

EXT DIM is effective only on terminals of type 9763, and only if specified immediately after NP.

Effect on output:

EXT DIM can be used to select the screen format on display terminals of type 9763. The standard format is 24 x 80. Additional formats are 32 x 80, 43 x 80 and 27 x 132. The possible screen formats can be queried with TSTAT/YINQUIRE.

The number of lines ll is specified in two bytes. The number of columns ccc is specified in 3 bytes.

When the device is switched on, and during connection setup, the standard (default) format is reset automatically. The screen format is reset to the standard format by NP, at end of program, and when the screen is cleared by the system.

If a program is interrupted (K2, BKPT) in LINE mode, the screen format is not reset to the standard format.

CHS is not supported in the system line.

&P.EXT FLD x (data display terminals only)

Set field characteristics

Effect on output

In the case of field-oriented display and structured output, a start of field is generated and certain characteristics are assigned to this new field. By default, this field is not printable. If emphasized display preceded this operand, it is set to reduced intensity for this field and to bright for all other fields.

x is a hexadecimal value selected by means of the following equates:

&P.PNS	protected field not sendable
&P.PRS	protected field sendable (automatic input)
&P.NUF	numeric field
&P.MOD	premodified field (only in extended line mode)
&P.MAR	markable field
&P.PRT	printable field
&P.INP	unprotected field not numeric, not markable, not printable
&P.ASK	protected field with automatic tabulator (3270 terminals only)

x is either the value of one of these equates or the sum of a combination of equates.

e.g. &P.EXT.FLD &P.NUF+&P.MAR field The field is numeric and
markable

If a combination of characteristics is defined, the output attribute 'reduced intensity' is retained. You must use the DIS control character to request other display attributes. Note that the field characteristics must always precede the output attributes.

FLD is reset by the logical record control characters (NL, NP, HPA, VPA) or by the field control characters (EPA, SPA, NUM, CHS, COL and FLD).

FLD is not supported in the system line.

Note that a markable field is of significance to the application only if the VTSUCB parameter READ=MODIFIED is used.

If the characteristics protected, not sendable and markable are assigned to a field, they correspond to the FHS attribute `PROTECTION=DETECTABLE` (protected selection field). If a field of this type is marked in read mode, the user receives only the field coordinates VPA and HPA and not the contents of the field. If you press the MAR key again, you can cancel 'marked' as an attribute of this field.

4. Logical local control characters

&P.LOC ctl

Set local attributes

LOC is effective only on data display terminals of type 9763.

Effect on input and output:

The attribute which is addressed with the control character ctl immediately following LOC is only created locally. ctl can be: EM1-EM4, NOR, DAR, DIS, CHS and COL.

A local attribute is valid for all subsequent characters up to the next start of field, if not explicitly reset before this. The attribute is reset using LOX (see page 66). Local attributes are character-specific. When the character disappears, so also does the local attribute; any character that is inserted does not bear the local attribute, whilst the characters moved to the right retain their local attribute.

The following substitute representation is used on data display terminals other than type 9763 and for printers:

The local display attributes LOC EM1- LOC EM4, LOC NOR, LOC DIS and LOC DAR are replaced by the corresponding display attributes EM1-EM4, NOR, DIS and DAR. There is no substitution for local character set attributes (LOC CHS) and color attributes (LOC COL).

In the case of LOC DAR the subsequent text characters, which are not visible on the screen, can be printed. The value of DARPRINTABLE defines whether characters substituted for invisible text characters can be printed.

Local attributes can only be entered if the LOCIN=YES option is set in the VTSUCB. Otherwise they are deleted from the input message. Local attributes not supported logically by VTSU are always deleted from the input message.

Note

The logical display control characters EM1-EM4 and NOR, which are located after LOC and followed by EM1-EM4, NOR and DAR, are only effective once the local attribute has been reset (using LOX NOR or LOX LOX). These display control characters only have an effect on characters which are overwritten or inserted.

&P.LOX ctl

Reset local attributes

Effect on input and output:

Local attributes are reset with LOX.

ctl specifies which local attributes are reset.

ctl can be:

- NOR (local EM1, EM2, EM3, EM4, DIS, NOR or DAR is reset)
- CHS (local loadable character set is reset)
- COL (choice of color is reset)
- LOX (all local attributes are reset).

LOX resets the local attribute to the last valid non-local value.

The following substitute representation is used on data display terminals other than type 9763 and for printers:

LOX NOR and LOX LOX reset the attribute to the last display attribute which was not generated by a substitute layout. There is no substitute representation for local character set attributes (LOX CHS) and local color attributes (LOX COL).

Local attributes can only be received if the LOCIN=YES option is set in the VTSUCB. Otherwise they are deleted from the input message. Local attributes not logically supported by VTSU are always deleted from the input message.

Note

- In the case of EXTEND=NO, a logical display control character (EM1, EM2, EM3, EM4, DIS, DAR or NUM) acts on the subsequent input field. This provides a means of imposing the dark, numeric or emphasized attributes on a normal line mode input. If the output is followed not by an input but by another output, the control character has no effect.
- If you want to specify a number of characteristics for a field, enter them in the following sequence: &P.CHS dd &P.COL dd &P.EXT &P.FLD xx &P.EXT &P.DIS xx. Note that the logical display control characters must always follow the logical control characters for the field characteristics. If you use the control character 6P.COL and the terminal supports color, the other display attributes are ignored. Physical sequences are still generated correctly. The same logical buffers are used for color and monochrome screens of type 9763.

- If the two logical control characters VPA and HPA are combined, the physical sequences generated differ depending on the positions of the control characters in the buffer.
 - If the control characters are at the start of the message, the cursor is positioned on the requested coordinates and a start of field with standard attributes is generated.
 - If the control characters are at the end of the message the cursor is positioned on the requested coordinates and special display formats of the data display terminal are reset to the standard display (normal, standard color, protected). The standard character set is retained.
 - If the control characters are within the message, the cursor is positioned on the requested coordinates and a start of field with standard attributes is generated. Special display formats of the data display terminal are reset to the standard display (normal, standard color, unprotected in line mode, protected in extended line mode). The standard character set is retained.
- Hardcopy support differs in line mode and extended line mode. In extended line mode a hardcopy of the entire screen is produced. In line mode a hardcopy is created only of the last field generated by a preceding control character.
- Fields generated with the EXT FLD control character must be defined as printable fields.

5. Logical control characters for printer support

Use of the logical control characters for printer support is only effective with EXTEND=NO.

&P.PLD Partial line down (line subscripted)

Effect on output:

The printer moves half a line up and the subsequent text is printed half a line lower. This is reset by PLU, at the logical end of the line or at end of message.

&P.PLU Partial line up (line superscripted)

Effect on output:

The printer moves half a line down and the subsequent text is printed half a line higher. This is reset by PLD, at the logical end of line or at end of message.

&P.VMI d

Vertical motion index (line spacing)

VMI is permitted only at the logical start of a page.

Effect on output:

specifies the line spacing:

- d=1: normal spacing (1/6 inch).
- d=2: condensed spacing (1/8 inch).
- d=3: half-line spacing (1/12 inch).

Is reset by MLN at end of message. The number of lines per page is adjusted to match the new line spacing.

&P.HMI d

Horizontal motion index (character pitch)

HMI is permitted only at the logical start of a line. Any left margin that has been set with LM is reset.

Effect on output:

specifies the character pitch:

- d=1: normal pitch (1/10 inch).
- d=2: condensed font A (character pitch 1/12 - 1/15 inch).
- d=3: condensed font B (character pitch 1/15 - 1/17 inch).

Reset by MLL and at end of message. The number of characters per line is automatically adjusted.

&P.NLQ

Near letter quality start

Effect on output:

On printers with near-letter quality capability, this feature is turned on by NLQ; otherwise NLQ is ignored. NLQ is reset at the end of the message.

&P.NLX

Near letter quality exit

Effect on output:

The output control character NLX turns off the near-letter quality feature.

&P.LM ddd

Left margin

Sets a left margin. The first character of all subsequent lines is printed in the column specified by ddd.

LM is only permissible at logical start of line and at the start of a message if there is no CAP at this point. It is reset by HMI, MLL and at the end of the message.

When the left margin is set, several successive backspaces (BS) are permitted. This enables the cursor to be positioned to the left of the left margin.

&P.PTS Proportional type start

Effect on output:

A visually more pleasing printed result is achieved by individual character spacing.

PTS is effective until the end of the message or PTX.

Column counting is disabled and is not started again until an NL, NP, ASF, VPA or HPA following a PTX is received.

&P.PTX proportional type exit:

Effect on output:

Deactivates proportional type.

&P.MLL ddd

Maximum line length:

Effect on output:

ddd defines the new maximum number of characters per line. Also, the character spacing is set to 1/10 inch, and any left margin is reset. MLL only works at logical start of page and for printers. The defined maximum line length remains effective for the entire duration of the connection unless redefined.

&P.MLN ddd

Maximum line number:

Effect on output:

ddd defines the new maximum number of lines per page. It also sets the line spacing to 1/6 inch. MLN only works at logical start of page and for printers. The defined maximum number of lines is set for the entire duration of the connection unless redefined.

&P.VPA ddd

Vertical position absolute (three-digit decimal)

Effect on output:

The output text starts in the line designated by ddd.

&P.HPA ddd

Horizontal position absolute (three-digit decimal)

Effect on output:

Subsequent characters in the current line are output starting at the column designated by ddd.

&P.ASF d Automatic sheet feeding (sheet insertion/ejection control) d= decimal number

Effect on output:

- For d=0 at the end of the message a sheet ejection is triggered. When a form feed attachment is used: switchover to tractor.
- For d=1,2,3 the sheet ejection is combined with a sheet feed from the feed tray designated by d.
- For d=9 sheet insertion takes place from the form feed attachment (9013 Printer only).

&P.CAP Continue actual position

Effect on output:

No positioning to the start of the next line at the beginning of a message. Printing starts at the current position of the printer carriage. No resetting of logical control characters at end of message or start of message. Additionally, multiple PLUs and PLDs are allowed. However, only one PLU or PLD is reset at logical end of line.

To ensure that there is a defined initial state, the printer must be positioned to the start of a line with NL, NP or VPA when CAP is used for the first time within an output sequence. CAP is permitted only as the first character in a message. Overwriting may occur when CAP is used for the first time or after a reset sequence (e.g. HMI reset) if the cursor is not explicitly positioned on the start of a new line.

6. Logical control characters with special functions

&P.DEL Delete

Effect on output:

The character is removed from the output text and is not transmitted to the terminal.

&P.BS Backspace (only display terminals with APL capability, and printers)

Effect on output:

The next text character is superimposed on the preceding character (to form a compound character not contained in the character set).

A multiple BS is only allowed when a left margin has been set previously with LM.

Meaning for input:

The next text character and the preceding one are treated as a single unit.

Note

For the 9022 Printer, BS (backspace) following SO (shift out into character set extension) is ignored.

&P.SUB Substitute

Effect on output:

This logical control character and all other EBCDIC control characters (code < X'40') which are not logical control characters are represented on the terminal by the valid substitute character (macro TCHNG SUB=OUTIN).

Effect on input:

The valid substitute character has been detected in the terminal input and replaced (only if requested via the macro TCHNG SUB=OUTIN, see the manual 'BS2000 TIAM').

&P.ESC Escape

Effect on output:

Generates the control character ESC (EDCDIC code X'27'). This is transferred together with the next character in unchanged form. This permits the use, in line mode, of terminal device functions that are not logically supported (see descriptions of the respective terminals).

Note

This control character causes the column and line counts by VTSU-B to be suppressed (no overflow control).

Column counts are resumed by the control characters NL, NP or VPA, line counts (overflow control) are resumed by the control character NP or VPA.

&P.DC4 Treated like ESC by VTSU

Effect on output:

Generates the control character DC4 (EBCDIC X'3C'). This is transferred together with the following character to the terminal. This permits the use, in line mode, of terminal device functions that are not logically supported (cf. descriptions of the respective terminals).

The effect on column and line counting is as described in the note on the ESC control character.

&P.HT Horizontal tabulation

Effect on output:

Generates the control character HT (EBCDIC X'05'), which is passed unchanged to the terminal (cf. descriptions of the respective terminals).

The effect on column and line counting is as described in the note on the ESC control character.

&P.VT Vertical tabulation

Effect on output:

Generates the control character VT (EBCDIC X'0B'), which is passed unchanged to the terminal (cf. descriptions of the respective terminals).

The effect on column and line counting is as described in the note on the ESC control character.

&P.EXT TRA d II

Transparent output

Effect on output:

EXT TRA allows the user to transfer control character sequences unchanged (transparently) to data display terminals and printers.

With d, the user can select the devices to which the transparent character sequence is to be transferred:

d = X'00'

The control character sequence is transferred for all device types.

d = device type supplied in TSTAT/YINQUIRE:

The control character sequence is transferred only to the specified device type.

With ll, the user specifies the length of the transparent control character sequence. ll is specified in two bytes in either decimal (00-99) or binary form (X'0000'-X'7FFF'). The first five bytes (EXT TRA d ll) are not counted as part of the length specification. If the length is greater than the maximum device buffer, EXT is converted to SUB.

Note

If you use the logical control character EXT TRA you can specify any physical control character.

If you use EXT TRA in order to modify positioning, correct overflow control is no longer guaranteed.

&P.EXT RPT dd

The subsequent displayable or NULL character is repeated dd times (repeat symbol)

Effect on output

EXT RPT enables you to reduce the size of your buffer in the case of multiple repeats of a character.

dd is the number of times you want the subsequent character repeated. dd is specified in two bytes, either in decimal (00-99) or in binary notation (X'0000'-X'7FFF').

Notes on the logical control characters

- a) If a control character is illegal for a particular output device or in its current position, it is ignored together with any associated number following it, or it is replaced (see description of the individual control characters).
- b) If a logical control character that expects a number to follow it is in fact followed by bytes that do not form a legal number (no number or an impossible number), the control character is replaced by SUB and the subsequent characters are treated as text.
- c) If the number following the logical control character is only illegal in special cases (e.g. too large for the current line length), it is processed according to the procedure described under a).
- d) The control characters ASF, VPA and HPA have the effect of a logical end of line and act on the preceding control characters in the same way as NL (reset character set 2 etc.).

Effect of logical display control characters on data display terminals

Terminal	Effect of logical display control characters					
	NOR	EM1	EM2	EM3	EM4	DAR
8110 TTY	-	-	-	-	-	-
8150	-	-	-	-	-	-
8151	still	flashing	flashing	flashing	flashing	-
8152	roman	italics	italics	italics	italics	-
8160 } 8162 } OM 1	low int. roman still	low int. roman flashing	low int. italics still	high int. roman still	high int. italics still	blanked
8160 OM 2	high int. roman still	high int. roman flashing	high int. italics still	low int. roman still	low int. italics still	blanked
9748 } 9749 } OM 1 9750 } 9751 }	low int. roman still	low int. roman flashing	low int. under- scored still	high int. roman still	high int. under- scored still	blanked
9748 } 9749 } OM 2 9750 } 9751 }	high int. roman still	high int. roman flashing	high int. under- scored still	low int. roman still	low int. under- scored still	blanked
9752	yellow roman	yellow flashing	white roman	green roman	red roman	blanked
9755 } 9758 } OM 1	low int. roman	low int. roman flashing	low int. under- scored	high int. roman	high int. under- scored	blanked

TTY: PT80, T100, T1000;

OM: operating mode

-: control character is ignored

Terminal	Effect of logical display control characters					
	NOR	EM1	EM2	EM3	EM4	DAR
9755 } 9758 } OM 2	high int. roman	high int. roman flashing	high int. under- scored	low int. roman	low int. under- scored	blanked
9763, 9759 monochrome screen	low int. roman	low int. roman flashing	low int. under- scored	high int. roman	high int. under- scored	blanked
9763, 9759 OM 2, mono. screen	high int. roman	high int. roman flashing	high int. under- scored	low int. roman	low int. under- scored	blanked
9763 color screen	yellow roman	cyan roman	white roman	green roman	red roman	blanked
9763 OM 2 color screen	green roman	red under- scored	red under- scored	yellow roman	white under- scored	blanked
3270 OM 1	low int. roman	high int. roman	high int. roman	high int. roman	high int. roman	blanked
3270 OM 2	high int. roman	low int. roman	low int. roman	low int. roman	low int. roman	blanked
3279 OM 1	green	red	red	red	red	blue
3279 OM 2	green	white	white	white	white	green
8121 8122	normal	italics	italics	italics	italics	-
9001	normal	under- scored	under- scored	under- scored	under- scored	-

TTY: PT80, T100, T1000;

OM: operating mode

-: control character is ignored

Terminal	Effects of logical display control characters					
	NOR	EM1	EM2	EM3	EM4	DAR
9002	normal	italics	under-scored	italics	italics and under-scored	-
9003	normal	italics	red	italics	red and italics	-
9004	normal	shadow script	under-scored	bold	bold and under-scored	-
9013	normal	under-scored	under-scored	bold	bold and under-scored	-
9012	normal	under-scored	under-scored	bold	bold and under-scored	-
9011- 18/19	normal	italics 1)	under-scored	bold	bold and under-scored	-
9001- 31/8931	normal	italics 1)	under-scored	bold	bold and under-scored	-
9021	normal	italics	under-scored	bold	bold and under-scored	-
9022	normal	shadow script	under-scored	bold	bold and under-scored	-

TTY: PT80, T100, T1000;

OM: operating mode;

-: control character is ignored

1) Only for connection to a 9763 DDT or a BAM controller, otherwise underscored or no effect.

Terminal	Effect of logical display control characters				
	SO	SI	SPA	EPA	NUM
8110 TTY	-	-	-	-	-
8150	-	-	protected low int.	unpro- tected high int.	-
8151	-	-	protected low int.	unpro- tected high int.	-
8152	APL character set	first character set	protected low int.	unpro- tected high int.	-
8160	-	-	protected low int.	unpro- tected high int.	unprotected high intensity *
8162	second character set	first character set	protected low int.	unpro- tected high int.	unprotected high intensity *
9748 9749 9750/9751	-	-	protected low int.	unpro- tected high int.	unprotected high intensity *
9752	-	-	protected yellow	unpro- tected green	unprotected green *
9755 9756 9758	-	-	protected low int.	unpro- tected high int.	unprotected high intensity *
9763/9759 monochrome screen	-	-	protected low int.	unpro- tected high int.	unprotected high intensity *

TTY: PT80, T100, T1000;

OM: operating mode;

-: control character is ignored

* entries must be numeric

Terminal	Effect of logical display control characters				
	SO	SI	SPA	EPA	NUM
9763 color screen	-	-	protected yellow	unpro- tected green	unprotected green *
3270	- -	- -	protected low int.	unpro- tected high int.	unprotected high inten. *
3279 OM 1	-	-	white	red	red
3279 OM 2	-	-	white	white	white
812x	-		-	-	-
9001	-	-	-	-	-
9002 ₃₎	second character set	first character set	-	-	-
9003	second character set	first character set	-	-	-
9004	second character set	first character set	-	-	-
9013 ₃₎	character set ex- tension 1)	basic character set 2)	-	-	-
9012 ₃₎	character set ex- tension 1)	basic character set 2)	-	-	-

- 1) right half of ISO 8-bit code table, practical for teletex character set, for example
- 2) left half of ISO 8-bit code table.
- 3) depending on printer type (see printer manual)

-: control character is ignored

* entries must be numeric

Terminal	Effect of logical display control characters				
	SO	SI	SPA	EPA	NUM
9011-18/19	second character set	first character set	-	-	-
9001-31/8931	-	-	-	-	-
9021	secondary font	primary font	-	-	-
9022 ₃₎	character set extension 1)	basic character set 2)	-	-	-

- 1) right half of ISO 8-bit code table, practical for teletex character set, for example
- 2) left half of ISO 8-bit code table.

TTY: PT80, T100, T1000; -: control character is ignored

Applicability of the logical control characters to individual printers and data display terminals

Type	PLD PLU	LM	PTS PTX	VPA	HPA	ASF	MLL	MLN	BS	CAP	ESC DC4	HT	VT
9004	X	X	X	X	X	X	X	X	X	X	X	X	X
9001	X	-	-	-	-	-	X	X	-	X	X	X	X
9003	-	-	-	-	-	-	X	X	X	X	X	X	X
9002	-	-	-	X	X	-	X	X	X	X	X	X	X
9013	X	X	X	X	X	X	X	X	X	X	X	X	X
9012	X	X	X	X	X	X	X	X	X	X	X	X	X
9011 18/19	X	-	X	-	X	X	X	X	X	X	X	X	X
9001 31/8931	X	-	X	-	X	-	X	X	X	X	X	X	X
9022	X	X	X	X	X	X	X	X	X	X	X	X	X
9021	X	-	X	X	X	X	X	X	X	X	X	X	X
812x	-	-	-	-	-	-	X	X	X	X	X	-	-
816x 975x 974x	-	-	-	X	*	-	-	-	-	-	X	X	X
976x	-	-	-	X	*	-	-	-	-	-	X	X	X
3270	-	-	-	X	*	-	-	-	-	-	X	X	X
3279	-	-	-	X	*	-	-	-	-	-	X	X	X

- X - function is initiated
 - - logical control character is suppressed
 * - if preceded by a legal VPA, the logical control character is suppressed

Note

- The 9022 Printer ignores proportional type unless the current font supports proportional spacing.
- Proportional type is impractical with the 9013 Printer unless a suitable character set is selected.

- When used on the 9013 and 9002 Printers, the control character VPAdd causes the execution of a number of line feeds defined by ddd.
- When used on the 9002 Printer, the control character HPAdd either adds the number of blanks defined by ddd or defines the column as of which the subsequent characters are to be output. You can select either of these two options. The insertion of ddd blanks is the default.

Applicability of logical control characters for output

Terminal	CHS	LOC LOX	EXT DIM	EXT TRA	NLQ NLX	EXT DIS	EXT FLD	COL	EXT RPT
9763	X	X	X	X	-	X	X	X	X
975x 9748 9749 816x 3270	-	1)	-	X	-	X	X	-	X
9001 9002 9003 9004 9013	-	1)	-	X	-	-	-	-	X
9012	-	1)	-	X	-	-	-	-	X
9011- 18/19	-	1)	-	X	X	-	-	-	X
9001- 31/8931	-	1)	-	X	X	-	-	-	X
9022	-	1)	-	X	-	-	-	-	X
9021	-	1)	-	X	-	-	-	-	X

- X - function is initiated
- - logical control character is suppressed
- 1) substitution takes place (see LOC and LOX control characters).

Note

On the 9763 Data Display Terminal, only the screen formats supplied with TSTAT/YINQUIRE are supported for the logical control character EXT DIM.

Effect of output attributes

Terminal	FL	UND	BLK	RIN	INV
8110	-	-	-	-	-
815x	-	-	-	-	-
8160	flashing	italics	blanked	low int.	-
9750	flashing	under- scored/ inverse ¹⁾	blanked	low int.	-
9755	flashing	under- scored/ inverse ²⁾	blanked	low int.	-
9758 9756	flashing	under- scored inverse ²⁾	blanked	low int.	inverse
9763 9759	flashing	under- scored inverse ²⁾	blanked	low int.	inverse
3270	-	-	blanked and not printable	low int.	-

-: control character is ignored

1) set by jumper or ROM

2) selectable by SIDATA

Effect of field attributes

Terminal	PNS	PRS	NUF	MOD	MAR	PRT	ASK
8110	-	-	-	-	-	-	-
815x	-	-	-	-	-	-	-
8160	protected not send- able	protected send- able	numeric	premod- ified	mark- able	print- able	-
9750	protected not able	protected send-	numeric	premod- ified	mark- able	print- able	-
9755	protected not send- able	protected send- able	numeric	premod- ified	mark- able	print- able	-
9758 9756	protected not send- able	protected send- able	numeric	premod- ified	mark- able	print- able	-
9763 9759	protected not send- able	protected send- able	numeric	premod- ified	mark- able	print- able	-
3270	protected not sendable	-	numeric	premod- ified	select- able ¹⁾	print- able ²⁾	protected automat. tabulator

-: control character is ignored

- 1) the first character in the field is a destination character
- 2) on a 3270 terminal, a non-printable field is automatically blanked

Character spacing and line spacing for the individual printers (in inches)

Printer	HMI1	HMI2	HMI3	VMI1	VMI2	VMI3
9004	1/10	1/12	1/15	1/6	1/8	1/12
9001	1/10	1/12	1/17	1/6	1/8	1/12
9003	1/10	1/12	1/15	1/6	1/8	1/12
9002	1/10	1/10	1/10	1/6	1/6	1/6
9013	1/10	1/12	1/15 1st ch.set only	1/6	1/8	1/12
9012	1/10	1/12	1/12	1/6	1/8	1/12
9011 18/19	1/10	1/12	1/15	1/6	1/8	1/12
9001 31/8931	1/10	1/12	1/15	1/6	1/8	1/12
9022	1/10	1/12	1/15	1/6	1/8	1/12
9021	1/10	1/12	1/15	1/6	1/8	1/12
812x	1/10	1/10	1/10	1/6	1/6	1/6

Note

- On the 9011-31/8931 Printers, bold (EM3, EM4) is not executed after HM13.
- On the 9021 printer, HMI3 is executed with 1/15" only when as suitable font has been selected; otherwise the next smallest character spacing is used.

Effect of reset at connect time on logically supported functions

Printer	HMI	VMI	PTS	SO	CHS	ASF	EM1-4	NLQ	CR
9001	1/10	S	-	-	CS1	-	R	-	S
9004	S	S	R	R	-	ASF1	R	-	-
9013	S	S	R	R	S	ASF _n , n(S)	R	-	S
9012	M	M	M	R	M	ASF _n , n(M)	R	M	M
9011 18/19	M	M	M	R	M	Tract ASF _n , n(M)	EM1: M EM2: R EM3: M	M	M
9001 31/8931	S	S	R	-	CS1	-	R	S	-
9022	S	S	R	R	R	ASF1	R	-	-
9021	S	S	R	R	R	ASF1	R	-	-
VTSU default	1/10	1/6	R	¹⁾ R	CS1	-	R	R	CR

S : Reset to switch setting
M : Reset to menu setting
R : Reset
- : No logical support for function
CS: Character set
n(S): n defined by switch setting
n(M): n defined by menu selection

1) Left half of the ISO 8-bit code table for printers with 8-bit character sets; otherwise basic character set.

Notes

- VTSU default
It is assumed for the logical support of these functions that the VTSU default values are set by menu or switch settings. Any differences from the default settings are the responsibility of the user.
- Automatic sheet feed
For the 9011-18/19 Printer, the menu setting for automatic sheet feed (ASF) simply means selection of tractor or sheet feeder.

Example: Macro expansion of VTCSET

```

      BVTCSET  START
      VTCSET  LOG
1  *
1  *           VIRTUAL TERMINAL CONTROL CHARACTER SET
1  *
1  *           IDLKG ALIGN=C
1  *
1  *           LOGICAL RECORD DELIMITERS
1  *
1  LOGNL      EQU   X'15'           LOGICAL LINE END (CONT NEXT LINE)
1  LOGNP      EQU   X'0C'           LOGICAL PAGE END (CONT NEXT PAGE)
1  LOGCL      EQU   X'0D'           LOGICAL LINE END (CONT SAME LINE)
1  LOGVPA     EQU   X'29'           LOG VERTICAL POS ABSOLUT (CONT LINE N)
1  LOGHPA     EQU   X'2A'           LOG HORIZONT POS ABSOLUT (CONT COL N)
1  LOGASF     EQU   X'21'           LOG SHEED FEDDING FROM CASETTE N D1
1  LOGCAP     EQU   X'20'           CONTINUE ACTUAL POSITION AT MSG BEGIN
1  *
1  *           LOGICAL UNIT DELIMITERS
1  *
1  LOGEM1     EQU   X'1D'           EMPHASIZED LAYOUT 1
1  LOGEM2     EQU   X'1F'           EMPHASIZED LAYOUT 2
1  LOGEM3     EQU   X'13'           EMPHASIZED LAYOUT 3
1  LOGEM4     EQU   X'14'           EMPHASIZED LAYOUT 4
1  LOGNOR     EQU   X'1E'           NORMAL LAYOUT
1  LOGDAR     EQU   X'12'           DARK LAYOUT
1  LOGPLD     EQU   X'2B'           PARTIAL LINE DOWN
1  LOGPLU     EQU   X'2C'           PARTIAL LINE UP
1  *
1  LOGSO      EQU   X'0E'           SHIFT OUT TO 2ND CHARACTER SET
1  LOGSI      EQU   X'0F'           SHIFT IN TO NORMAL CHARACTER SET
1  *
1  LOGSPA     EQU   X'36'           START PROTECTED AREA
1  LOGEPA     EQU   X'08'           END PROTECTED AREA
1  LOGNUM     EQU   X'11'           START NUMERIC (UNPROTECTED) AREA
1  *
1  LOGCHS     EQU   X'06'           CHARACTER SET D1D2
1  LOGCOL     EQU   X'17'           COLOUR CHOICE
1  LOGLOC     EQU   X'09'           LOCAL ATTRIBUTE START S1
1  LOGLOX     EQU   X'0A'           LOCAL ATTRIBUTE EXIT S1
1  *
1  LOGVMI     EQU   X'24'           VERTICAL MOVEMENT INDICATOR D1
1  LOGHMI     EQU   X'23'           HORIZONTAL MOVEMENT INDICATOR D1
1  LOGLM      EQU   X'38'           LEFT MARGIN D1D2D3
1  LOGPTS     EQU   X'1A'           PROPORTIONAL TYPING START
1  LOGPTX     EQU   X'1B'           PROPORTIONAL TYPING END
1  LOGMLL     EQU   X'33'           MAXIMAL LINE LENGTH
1  LOGMLN     EQU   X'35'           MAXIMAL LINE NUMBER (ON PAGE)
1  LOGNLQ     EQU   X'39'           NEAR LETTER QUALITY START
1  LOGNLX     EQU   X'3B'           NEAR LETTER QUALITY EXIT
1  *
1  *           SPECIAL FUNCTIONS
1  *
1  LOGDEL     EQU   X'07'           DELETE
1  LOGBS      EQU   X'16'           BACKSPACE
1  LOGSUB     EQU   X'3F'           SUBSTITUTE
1  *

```

```

1 *          DELIMITER EXTENSION
1 *
1 LOGEXT    EQU    X'3E'          DELIMITER EXTENSION BYTE
1 *
1 *          EXTENDED LOGICAL DELIMITERS
1 *
1 LOGTRA    EQU    C'T'          TRANSPARENT OUTPUT X11L2
1 LOGDIM    EQU    C'D'          DIMENSION OF SCREEN D1D2D3D4D5
1 LOGRPT    EQU    C'R'          REPEAT NEXT CHARACTER NN TIMES
1 LOGDIS    EQU    C'I'          SET DISPLAY ATTRIBUTES
1 LOGRS     EQU    X'00'         RESET
1 LOGFL     EQU    X'01'         FLASHING
1 LOGUND    EQU    X'02'         UNDERSCORED
1 LOGBLK    EQU    X'04'         BLANKED
1 LOGRIN    EQU    X'08'         REDUCED INTENSITY
1 LOGINV    EQU    X'10'         INVERSE
1 LOGFLD    EQU    C'F'         SET FIELD CHARACTERISTICS
1 LOGINP    EQU    X'00'         INPUT
1 LOGPNS    EQU    X'01'         PROTECTED NOT SENDABLE
1 LOGPRS    EQU    X'20'         PROTECTED SENDABLE
1 LOGNUF    EQU    X'02'         NUMERIC
1 LOGMOD    EQU    X'04'         PRE-MODIFIED
1 LOGMAR    EQU    X'08'         MARKABLE
1 LOGPRT    EQU    X'10'         PRINTABLE
1 LOGASK    EQU    X'40'         AUTOMATIC SKIP
1 *
1 *
1 *          PHYSICAL UNIT DELIMITERS
1 *
1 LOGESC    EQU    X'27'         ESCAPE X
1 LOGDC4    EQU    X'3C'         DC4 X
1 LOGHT     EQU    X'05'         HORIZONTAL TABULATION
1 LOGVT     EQU    X'0B'         VERTICAL TABULATION
1 *
1          * ,VTCSET      050      930731      53531028
          END

```


3.3 VTSU control block: VTSUCB

The **VTSUCB** enables you to set VTSU parameters for input and output independently of the access method.

In the case of **DCAM applications**, you can use the following call to couple the VTSUCB to the RPB.

```
YRPB      OPTCD=VTSUCB,VCBADR=relexp
```

The VTSUCB is evaluated by the macros YCHANGE, YINQUIRE, YSEND, YRECEIVE and YSENDREC.

The YCHANGE RPB=addr,OPTCD=VTSUCB,VCBADR≠0 macro causes DCAM to place the VTSUCB in its own data structure. There is only one VTSUCB per connection.

Note that with DCAM applications it is not possible to change between input and output (YSEND and YRECEIVE) and simultaneously change mode. The mode of the YRECEIVE macro is always defined by the preceding YSEND, which sets the terminal to the requested mode.

If you are using a **TIAM application**, you can link the VTSUCB to the input/ output interface with one of the following calls:

```
        WROUT  record, error[,PARMOD=31],VTSUCBA=vtsuch
or      WRTRD  record1,,record2,,[length],error[,PARMOD=31],VTSUCBA=vtsuch
or      RDATA  record,error,[length],[,A][,PARMOD=31],VTSUCBA=vtsuch
```

The parameters NOPOS, READ, CURPOS, UPDATE and AUTOTAB are new.

Operation	Parameters
VTSUCB	<pre> [MF={ D C L M }], [PREFIX=], [MACID=] [, MODE= (MIXED, inmode, outmode)] [, LOW= { YES NO }] [, BELL= { YES NO }] [, GETFC= { YES NO }] [, HCOPY= { YES NO }] [, LOCIN= { YES NO }] [, NOLOG= { YES NO }] [, RETINF= { xx *NONE }] [, IHDR= { YES NO }] [, OHDR= { YES NO }] [, CODETR= { YES NO }] [, SPECIN= { N I C }] [, CCSNAME= { *EXTEND ccsname }] [, HOM= { YES NO }] [, AUTOTAB= { STD YES NO }] [, NOPOS= { YES NO }] [, CURPOS= { YES NO }] [, READ= { UNPROT MODIFIED }] [, UPDATE= { YES NO }] [, MODE=LINE] [, BELL= { YES NO }] [, GETFC= { YES NO }] [, HCOPY= { YES NO }] [, LOW= { YES NO }] [, NOLOG= { YES NO }] [, RETINF= { xx *NONE }] [, SPECIN= { N I C }] [, CCSNAME= { *EXTEND ccsname }] [, HOM= { YES NO }] [, NOPOS= { YES NO }] </pre>

Operation	Parameters
	<pre> [,MODE=EXTEND][,BELL={ YES NO }][,GETFC={ YES NO }][,LOCIN={ YES NO }] [,LOW={ YES NO }][,CCSNAME={ *EXTEND ccsname }] [,AUTOTAB={ STD YES NO }][,HCOPY={ YES NO }][,CURPOS={ YES NO }] [,READ={ UNPROT MODIFIED }][,UPDATE={ YES NO }] [,MODE=INFO][,BELL={ YES NO }][,GETFC={ YES NO }][,LOW={ YES NO }] [,NOLOG={ YES NO }][,CCSNAME={ *EXTEND ccsname }] [,MODE=PHYS][,IHDR={ YES NO }][,LOW={ YES NO }][,OHDR={ YES NO }] [,CCSNAME={ *EXTEND ccsname }][,CODETR={ YES NO }] [,MODE=FORM][,LOW={ YES NO }][,CCSNAME={ *EXTEND ccsname }] [,MODE=TRANS] [,MODE=CHIP] </pre>

Description of parameters**MF**

- =D** generates a DSECT.
- =C** generates the layout of the VTSUCB in the current data structure, naming each field and issuing all equates. The standard header is not filled.
- =L** generates a parameter list; the fields are filled in accordance with the specified macro keyword parameters. The names and equates are not issued. The standard header is filled automatically.
- =M** A VTSUCB generated using MF=L is modified with the keyword parameters of the parameter list. Only the fields addressed by the keyword parameters are filled. The remaining fields remain unchanged. Before this call, the DSECT must have been generated for the VTSUCB using MF=D and placed on the VTSUCB generated with MF=L by means of a USING statement. The same prefix must be used as for the associated DSECT.

PREFIX

- = x** specifies the first character to be prefixed to the names defined with MF=D or MF=C Default: PREFIX=Y.

MACID

- =xxx** specifies the 2nd - 4th characters of the names defined with MF=C. Default: MACID=VTC.

MODE

- =LINE** The current terminal is to be handled as a virtual line or page terminal. The message can be structured with the aid of logical control characters (see 'Logical control characters: VTCSET', page 55).
- In the case of TIAM applications, no other control characters are permitted. If any occur, they are converted to a substitute character defined by the user (see the TIAM User Guide, description of the `/MODIFY-TERMINAL-OPTIONS SUBSTITUTE-CHARACTER=` command. If SYSOUT is not a terminal, only the logical control characters NL and NP are interpreted, e.g. for printer output in batch mode. In input, the device-specific message header is not returned.

=EXTEND (For 9749, 975x, 9763, 816x Data Display Terminals and 3270 terminals only, printers process EXTEND internally as line mode).

The current terminal is to be handled as a virtual line or page terminal. By default, the output is protected and at low intensity. The message may be structured with logical control characters (see 'Logical control characters: VTCSET', page 55). The keys RU, EFZ, AFZ and LSP are disabled.

In the case of TIAM applications, areas in which the user may enter data begin with EPA, DAR or NUM and end with SPA. With 3270 terminals it must be noted that logical control characters take up space on the screen. However, a number of consecutive logical control characters require only one character position.

In the case of TIAM applications NUL is treated as a valid character in input/output. It is sent from the program to the terminal and vice versa. With 3270 terminals it must be noted that null characters are not transmitted to the host computer. VTSU-B pads fields returned in shortened form by adding null characters until the original length is reached. The user thus always receives output-length fields.

In the case of TIAM applications the beginning of an output message is displayed at the start of the line following the cursor position. Before the first text character is displayed, the screen is cleared from the cursor position if the message does not begin with VPA.

If the end of the screen is reached during output, output continues from the start of the screen. Continuation is always unprotected as far as the next start of field. Screen overflow control is disabled.

In the case of DCAM applications, the control character NL is not permitted in the input if EXTEND is specified (FDBK 04 00 48). In the case of TIAM applications, if the control character NL is detected in an input message processing continues and return code X'2C' is returned at the WRTRD interface, or X'0018' in the MAIN RC of the VTSUCB.

- =INFO** Messages can be displayed in a special information line (system status line) without the risk of destroying important data.
- This operand is intended primarily for application programs sending "asynchronous" messages to terminals when the information currently displayed is not known.
- The message is displayed:
- on terminals with a hardware system line: always protected in a hardware system line (e.g. DDT 9749, 9750, 9763)
 - TIAM applications only:
 - on terminals with no hardware system line: protected, in 24th line of the screen. This must, however, be preceded by output with `MODE=PHYS` or `MODE=FORM`, and `TCHNG INFOLIN=YES` must have been set.
 - in all other cases: as a normal line-mode message.
- If the message is longer than one screen line, it is split up and output line by line.
- In the case of TIAM applications, the system observes the wait time specified in the command
`/MODIFY-TERMINAL-OPTIONS OVERFLOW-CONTROL=TIME`
- DCAM applications do not automatically reset the system line. With TIAM applications the system line is automatically reset after the first input that follows an output in the system line.
- If `MODE=INFO`, the input is treated as a line-mode input.
- =PHYS** The messages are to be output at the terminal "physically", i.e. without being edited by the system or read in physically from the terminal. This allows special device functions to be executed for which the `LINE` or `FORM` mode is insufficient.
- =FORM** The application program uses the software component `FHS` or format handling, which also edits the message in a form suitable for output on a particular terminal.
- =(MIXED,inmode,outmode)**
- The application program combines different modes for input and output. Only `MODE=LINE`, `EXTEND`, `FORM` and `PHYS` may be combined. `MODE=CHIP`, `INFO` or `TRANS` must not be combined. The default value for the `LOW` parameter for input and output depends on the specified input mode. If `MODE=PHYS`, then `LOW=YES`, if `MODE=LINE`, `EXTEND` or `FORM`, then `LOW=NO`. All

other parameter values are set in accordance with the specified input/output modes.

- =TRANS** The output data is to be transmitted "transparently", i.e. it may consist of any binary characters (5, 7 or 8 bits per character depending on the device code), and is not converted on the transmission path. If the transmission path has not been generated as a "potentially transparent" path, the output is rejected with the return code X'04'.
- =CHIP** The output message is forwarded to the chip card terminal by the device protocol (810 protocol). The message must have been generated in expanded mode. If the chip card terminal cannot be accessed, the output is rejected with the return code X'81' or X'82'.
- For input, the system checks whether the message has come from the chip card terminal and removes the device protocol. The input message is preceded by the function key code as the first byte. Input messages which do not come from the chip card terminal are converted into the short code K14.
- CCSNAME** Specifies the name of the coded character set to be used for this message. Specify the code name of the EBCDIC variant. The name of the corresponding ISO code variant is automatically rejected. The name must not be more than 8 bytes in length. Note that if the character set in use is changed, the screen is automatically cleared prior to the new output.
- =ccsname** Explicitly specifies the coded character set name. The options are:
- | | |
|---------|--|
| ccsname | Name of code of user's choice.
Only EBCDIC codes are supported. |
| blank | If no name is specified, the standard mode is adopted by default. The standard mode is either a 7-bit mode or an 8-bit mode activated with the <code>MODIFY-TERMINAL-OPTIONS</code> command. |
- =*EXTEND** The extended standard code is automatically used.
For BS2000 V10.0A the extended standard code is the system standard code.
For BS2000/OSD-BC V1.0 the extended standard code is the user standard code.

CODETR	Specifies for physical output whether the message is to be converted from or to the specified code. This parameter is of assistance only for output on printers that process ESCAPE sequences not coded in conformance with the EBCDIC core. These special ESCAPE sequences are ignored by VTSU.
= <u>YES</u>	VTSU converts the message from or to the specified code. VTSU recognizes and skips the standard ESCAPE sequences.
= <u>NO</u>	The control characters SO/SI are evaluated. VTSU does not implement any further conversion.
BELL	
= <u>YES</u>	An acoustic alarm sounds when an output takes place (only on 9749, 975x, 9763, 816x Data Display Terminals and 3270 terminals with special hardware option).
= <u>NO</u>	No acoustic alarm sounds when an output takes place.
HOM	(816x, 9749, 975x, 9763 DDTs and 3270 display terminals only.)
= <u>YES</u>	The message is to be output in unstructured, homogeneous form, i.e. the entire message is regarded as an output unit. The length of the message is restricted by the size of the system's output buffer. Effect for 816x, 975x, 9763 DDTs and 3270 terminals in operating mode 1 (TIAM applications only): If a character in an output message is modified, the entire message can be returned to the originator, if it is not explicitly structured by means of logical control characters.
= <u>NO</u>	The message is to be output in structured, heterogeneous form, i.e. a logical line is regarded as an output unit. Effect for 816x, 975x, 9763 DDTs and 3270 terminals in operating mode 1 (TIAM applications only): Individual logical lines can be modified separately and thus returned to originator on a selective basis.
GETFC	
= <u>YES</u>	The logical function key code representing the key which initiated the transmission at the terminal is transmitted as the first character in the message.
= <u>NO</u>	No function key code is transferred.
HCOPY	
= <u>YES</u>	The message output to a data display terminal is simultaneously output to a hardcopy unit (printer) connected to the terminal.

In the case of TIAM applications, a hard copy is produced only if a hardcopy unit was generated for the data display terminal at connection setup or by means of the MODIFY-TERMINAL-OPTIONS command (see TIAM User Guide). For 3270 terminals, the hardcopy unit must be assigned (generated) at connection setup.

If HCOPY=YES is specified without the EXTEND mode for TIAM applications and the message contains the logical control character SPA, EPA, NUM or DAR (if DARPRINTABLE=N), only the last, unprotected portion of the message is printed, not the entire message. If OVERFLOW-CONTROL=NO is also specified, it may happen that only part of the output is printed on the hardcopy unit. The OVERFLOW-CONTROL operand is described under the MODIFY-TERMINAL-OPTIONS command in the TIAM User Guide. Setting HCOPY=YES automatically selects operating mode 2 for the output.

=NO

The output message is not printed out simultaneously at a hardcopy unit (printer) connected to the terminal.

IHDR

=YES

The entire message prefix is passed to the application program (default value for MDE=PHYS).

On 3270 terminals, the message header comprises the application ID (AID) and the two-byte cursor position.

=NO

The message prefix is not passed to the application program.

LOCIN

Input of local attributes. This operand is relevant only to data display terminals which support local attributes (e.g. the 9763 Data Display Terminal).

=YES

Local attributes in the input message are passed to the user as logical control characters (see 'Logical control characters: VTCSET' page 55). If the local attributes were set by LOC EM1-4 or LOC EXT DIS x the logical control character passed on to the user is always LOC EM1-4.

=NO

Local attributes are removed from the input message and are not passed on to the user.

LOW	If MODE=MIXED, the default value of the LOW parameter for input and output depends on the specified input mode. LOW=YES if the input mode is MODE=PHYS, whereas LOW=NO if the input mode is MODE=LINE, EXTEND or FORM.
=YES	Lowercase letters are passed to the application program (default option for MODE=PHYS).
= <u>NO</u>	All lowercase letters are passed to the application program as upper case.
NOLOG	
=YES	Logical control characters are not interpreted. All characters less than X'40' in the EBCDI code are replaced by SUB (smudge character). Only printable characters are accepted.
= <u>NO</u>	All logical control characters are interpreted and special physical control characters are accepted (see VTCSET, e.g. ESC,DC4). Other characters < X'40' are replaced by SUB. Printable characters are accepted.
OHDR	
=YES	The message includes a user-specific header which the system prefixes to the output text. The length of the message header +1 must be specified in binary in byte 5 of the message. If you are using a TIAM application, bear in mind for outputs to 8160, 975x and 9763 Data Display Terminals and their locally connected printers that the system uses (MODE=LINE) or FHS (MODE=FORM) with parameters (PAG) and does not employ message headers (PARAM0, PARAM1). The differences between the two methods of operation are described in the user guides for the data display terminals and printers.
= <u>NO</u>	The message prefix/PAGs (see above) are prefixed to the output text by the system.
RETINF	Applies to printer terminals only Return information is expected from the printer terminal.
=xx	Any two printable characters which are to be returned with the return information. Quotation marks returned as part of this information must be duplicated (e.g. RETINF=""). The return information is 4 bytes long and has the following structure:
	<pre> Byte 0 Identification (X'41' positive/ X'42' negative) Bytes 1-2 RETINF bytes Byte 3 Information on printer status (printer-specific) </pre>
= <u>*NONE</u>	No return information is transmitted.

SPECIN	Request for special input. If special input is requested, the SPECIN parameter must have been specified in the output message requesting this input.
= <u>N</u>	Normal input from the terminal.
=I	The data is input from the ID card reader. The input data may consist of the ID card information or the short code K14. This specification is only possible with 9749, 975x, 9763, 816x and 3270 terminals for which an ID card reader has been defined. At 3270 display terminals, unlike TRANSDATA devices, data can be input from a defined ID card reader at any time. If input from an ID card reader is requested, any other input is converted to K14.
=C	The input data is confidential and must not be visible on the terminal. This is achieved by blanking, by clearing the screen (the screen format is reset to 24x80), or by overwriting the input line (on printer terminals).
NOPOS	(For printers only) The position at which a message is output is defined for line mode and mixed mode. In mixed mode, this parameter is not accepted unless the output mode is MODE=LINE. The parameter is ignored under all other circumstances.
=YES	The output message begins at the start of the current line.
= <u>NO</u>	The output message begins at the start of the next line.
READ	The physical read mode is defined for the extended line mode and for mixed mode. In mixed mode this parameter is accepted only if EXTEND is the value for both input and output mode. The parameter is ignored under all other circumstances. Note that DCAM applications with extended line mode require the same physical read mode (UNPROT or MODIFIED) for both macros when a YSEND is followed by a YRECEIVE.
= <u>UNPROT</u>	All unprotected fields, including those that were not modified, are returned. In order to ascertain which values have been changed, you must compare the data you receive with your original data.
=MODIFIED	Only the modified fields are returned. The position on the screen is prefixed to each modified field in the user buffer (see descriptions of the logical control characters VPA and HPA).

- CURPOS** Defines whether the cursor position is returned following the input for extended line mode and mixed mode. In mixed mode this parameter is accepted only if EXTEND is the value for both input and output mode. The parameter is ignored under all other circumstances.
- =YES** The fields YVTCPOSL and YVTCPOSC are returned with the current cursor position (lines and columns) after the input.
- =NO** No information on the cursor position is returned. The fields YVTCPOSL and YVTCPOSC are undefined.
- UPDATE** For form output in extended mode and mixed mode, you can define whether the entire screen will be updated or only the modified lines. A modified line is a line in which either an existing field has been updated or a new field has been created. In mixed mode this parameter is accepted only if EXTEND is the value for both input and output mode. The parameter is ignored under all other circumstances.
- =YES** Only modified lines are updated.
If you create a new field, make sure that the end of the new field has the output attribute 'blanked' and the field attribute 'protected'. When a new field is created, binary zeros are output as far as the next start of field. The output attribute 'blanked' suppresses the output of binary zeros and blanks are output instead. The field attribute 'protected' prevents the subsequent field from being overwritten by the new field.
Note that when the screen is updated the same coded character set name (CCSNAME) must be used as when the original screen was generated. If there is a discrepancy, the original screen is deleted and only the updated lines are output.
Note, too, that updating does not implicitly reset field attributes. For example, a field assigned the attribute 'premodified' retains this attribute even after the update. This means, therefore, that attributes must be explicitly reset.
- =NO** The entire screen is updated for the first logical new page.

AUTOTAB	Defines the automatic tabulator from an unprotected field to the next unprotected field. In mixed mode this parameter is accepted only if EXTEND is the value for both input and output mode. The parameter is ignored under all other circumstances.
= <u>STD</u>	Processing depends on the value of the EXPROPOS parameter.
=YES	As soon as you enter a character at the end of an unprotected field, the cursor automatically jumps from this unprotected field to the next unprotected field (even if EXPROPOS=Y).
	<i>Note</i>
	<ul style="list-style-type: none">– On 3270 terminals the cursor can always be moved to protected fields with the aid of the arrow keys. If, however, AUTOTAB=YES, the cursor automatically jumps from one input field to the next as soon as a character is entered at the end of the original field.– The EXPROPOS parameter is ignored by 3270 terminals.
=NO	The cursor does not move on automatically (even if EXPROPOS=N).

Return information

VTSU enters error messages in the VTSUCB standard header. The standard header also contains errors which are not associated directly with the VTSUCB. A special return code supplied by the access methods indicates whether the VTSUCB contains error messages.

Error information in the VTSUCB standard header

SUBCODE		MAINCODE		Meaning
2	1	2	1	
X'00'	X'00'	X'00'	X'00'	Successful processing *
X'58'	X'00'	X'00'	X'08'	Error in RETINF byte Parameter ignored
X'02'	X'00'	X'00'	X'08'	Mode not valid for current terminal Substitute representation
X'00'	X'01'	X'FF'	X'FF'	Error in UNIT or FUNCT
X'xx'	X'01'	X'00'	X'04'	Parameter error in VTSUCB (see meaning of SUBCODE 2)
X'40'	X'01'	X'00'	X'04'	Header length of user-specific message not valid
X'50'	X'01'	X'00'	X'04'	User buffer length for input not valid
X'60'	X'01'	X'00'	X'04'	Requested XHCS function not available. XHCS not loaded.
X'61'	X'01'	X'00'	X'04'	XHCS function requested for 7-bit data terminals.
X'62'	X'01'	X'00'	X'04'	XHCS function not supported
X'80'	X'01'	X'00'	X'04'	MODE parameter not valid for command type
X'86'	X'01'	X'00'	X'04'	CCSNAME of VTSUCB incompatible with devices
X'1E'	X'01'	X'00'	X'04'	Invalid CCS name
X'00'	X'03'	X'FF'	X'FF'	Error in VERSION
X'xx'	X'20'	X'00'	X'04'	Internal error (for diagnostics)
X'00'	X'40'	X'00'	X'0C'	Output message truncated
X'00'	X'40'	X'00'	X'10'	Input message truncated
X'00'	X'40'	X'00'	X'18'	Extended line mode Input message truncated
X'02'	X'40'	X'00'	X'04'	Mode not valid for current terminal No substitute representation
X'10'	X'40'	X'00'	X'20'	Limited information
X'81'	X'40'	X'00'	X'04'	No chip card terminal available for terminal
X'82'	X'40'	X'00'	X'04'	Chip card terminal available but not accessible
X'83'	X'40'	X'00'	X'04'	Data display terminal rejects message for chip card terminal

* In conjunction with 8-bit code processing the meaning can vary (see the 'XHCS' manual).

Meaning of the individual fields:

- SUBCODE 1 indicates the error class
 - X'00': Successful processing
Additional information is provided on minor corrections to the input parameters by VTSU.
 - X'01': Parameter error
Function was not executed
 - X'03': Error in VTSUCB version
Function was not executed
 - X'20': Internal error (system error)
Function was not executed
 - X'40': see table
- SUBCODE 2 Error information used for diagnostics only.

Meaning of SUBCODE 2:

- Where SUBCODE 1 = X'00' for diagnostics only
- Where SUBCODE 1 = X'01' (parameter error)
In SUBCODE 2, the first incorrect parameter in the VTSUCB is displayed. Only the parameters relevant to the selected MODE, however, are checked. Other parameters are always ignored, and no return code is issued.
 - 08: Error in length specification in VTSUCB
 - 10: Error in MODE parameter
 - 11: Error in HCOPY parameter
 - 12: Error in BELL parameter
 - 13: Error in NOLOG parameter
 - 14: Error in READ parameter
 - 15: Error in HOM parameter
 - 16: Error in RETINF parameter
 - 17: Error in LOCIN parameter
 - 18: Error in OHDR parameter
 - 19: Error in CODETR parameter
 - 1A: Error in IHDR parameter
 - 1B: Error in LOW parameter
 - 1C: Error in SPECIN parameter
 - 1D: Error in GETFC parameter
 - 1E: Error in CCSNAME parameter
 - 1F: Error in CURPOS parameter
 - 20: Error in UPDATE parameter
 - 22: Error in AUTOTAB parameter
 - 23: Error in NOPOS parameter

- Where SUBCODE 1 = X'20' (internal error) for diagnostics
- Where SUBCODE 1 = X'40'
 MAINCODE 1 = X'04' function was not executed (see table)
 MAINCODE 1 ≠ X'04' see table
- MAINCODE 1 Error information for the application program. The application program can use this information to identify operating errors.
- MAINCODE 2 is not currently assigned and is set to the value X'00'. If the standard header contains errors relating to the identification of the product itself (e.g. wrong version), the value is set to X'FF'.

Example, dummy section (DSECT) for the VTSU control block

```

    BVTSUCB  START
    VCBSECT  VTSUCB MF=D,PREFIX=Y
1  VCBSECT  IDLKG VER=050,SECT=D
2          *,VERSION 050
2  VCBSECT  DSECT
1          FHDR MF=(C,YVTC),EQUATES=NO
2          DS      0A
2  YVTCFHE  DS      0XL8          0  GENERAL OPERAND LIST HEADER
2  *
2  YVTCIFID DS      0A          0  INTERFACE IDENTIFIER
2  YVTCFCTU DS      AL2          0  FUNCTION UNIT NUMBER
2  *
2  *
2  *
2  *
2  YVTCFCT  DS      AL1          2  FUNCTION NUMBER
2  YVTCFCTV DS      AL1          3  FUNCTION INTERFACE VERSION NUMBER
2  *
2  YVTCRET  DS      0A          4  GENERAL RETURN CODE
2  YVTCsRET DS      0AL2        4  SUB RETURN CODE
2  YVTCsR2  DS      AL1         4  SUB RETURN CODE 2
2  YVTCsR1  DS      AL1         5  SUB RETURN CODE 1
2  YVTCmRET DS      0AL2        6  MAIN RETURN CODE
2  YVTCmR2  DS      AL1         6  MAIN RETURN CODE 2
2  YVTCmR1  DS      AL1         7  MAIN RETURN CODE 1
2  YVTCFHL  EQU     8           8  GENERAL OPERAND LIST HEADER LENGTH
2  *
1  *
1  *  SUBCODE 1 VALUES
1  *
1  YVTCECPM EQU     X'01'       ERROR CLASS PARAMETER ERROR
1  YVTCECIN EQU     X'20'       ERROR CLASS INTERNAL ERROR
1  YVTCECSP EQU     X'40'       ERROR CLASS SPECIAL ERROR
1  *
1  *  SUBCODE 2 VALUES
1  *
1  YVTCERLN EQU     X'08'       ERROR IN LENGTH OF VTSUCB
1  YVTCERMO EQU     X'10'       ERROR IN MODE PARAMETER
1  YVTCERHC EQU     X'11'       ERROR IN HARDCOPY PARAMETER

```


1	YVTCERBE	EQU	X'12'	ERROR IN BELL PARAMETER
1	YVTCERNO	EQU	X'13'	ERROR IN NOLOG PARAMETER
1	YVTCERRD	EQU	X'14'	ERROR IN READ PARAMETER
1	YVTCERHO	EQU	X'15'	ERROR IN HOM PARAMETER
1	YVTCERRE	EQU	X'16'	ERROR IN RETINF PARAMETER
1	YVTCERLO	EQU	X'17'	ERROR IN LOCIN PARAMETER
1	YVTCEROH	EQU	X'18'	ERROR IN OUTPUT HEADER PARAMETER
1	YVTCERCO	EQU	X'19'	ERROR IN CODETR PARAMETER
1	YVTCERIH	EQU	X'1A'	ERROR IN INPUT HEADER PARAMETER
1	YVTCERLW	EQU	X'1B'	ERROR IN LOWER INPUT PARAMETER
1	YVTCERSP	EQU	X'1C'	ERROR IN SPECIAL INPUT PARAMETER
1	YVTCERGE	EQU	X'1D'	ERROR IN GET FUNCTION CODE PARAMETER
1	YVTCERCC	EQU	X'1E'	INVALID CCSNAME
1	YVTCERCP	EQU	X'1F'	ERROR IN CURPOS PARAMETER
1	YVTCERUP	EQU	X'20'	ERROR IN UPDATE PARAMETER
1	YVTCERWA	EQU	X'21'	ERROR IN WARINFO PARAMETER
1	YVTCERAT	EQU	X'22'	ERROR IN AUTOTAB PARAMETER
1	YVTCERNP	EQU	X'23'	ERROR IN NOPOS PARAMETER
1	*			
1	YVTCERXH	EQU	X'60'	XHCS FCT REQUESTED BUT XHCS NOT LOADED
1	YVTCER7B	EQU	X'61'	XHCS FCT REQUESTED FOR 7-BIT TERMINAL
1	YVTCERBS	EQU	X'62'	XHCS FCT SUPPORT ONLY FROM BS2000 V10
1	*			
1	YVTCERM1	EQU	X'80'	MODE NOT VALID FOR COMMAND TYPE
1	YVTCERM2	EQU	X'81'	MODE=CHIP USED BUT NO CKT ANNOUNCED
1	YVTCERC1	EQU	X'82'	CKT NOT AVAILABLE FROM TERMINAL
1	YVTCERC2	EQU	X'83'	OTHER ERROR CODE FROM DSS BY CKT-MSG
1	YVTCERX2	EQU	X'86'	VTSUCB CCSN INCOMPATIBLE WITH DEVICE
1	*			
1	YVTCERO1	EQU	X'40'	HEADER LENGTH OF OUTPUT MSG NOT VALID
1	YVTCERI1	EQU	X'50'	USER BUFFER LEN FOR INPUT NOT VALID
1	*			
1	* MAINCODE VALUES FOR SUBCODE 1 = X'40'			
1	*			
1	YVTCMRPM	EQU	X'04'	WRONG PARAMETER FOR DEVICE
1	YVTCMROT	EQU	X'0C'	OUTPUT TRUNCATION
1	YVTCMRIT	EQU	X'10'	INPUT TRUNCATION
1	YVTCMRNL	EQU	X'18'	NL IN EXT LINE INPUT MESSAGE
1	*			
1	* MAINCODE VALUES FOR SUBCODE 1 = X'00'			
1	*			
1	YVTCMRCO	EQU	X'08'	CORRECTED ERROR
1	*			
1	*			
1	YVTCLEN	DS	H	LENGTH OF VTSUCB
1	*			
1	YVTCINM	DS	C	INPUT MODE FOR MODE=MIXED
1	YVTCOUTM	DS	C	OUTPUT MODE FOR MODE=MIXED
1	*			
1		DS	XL4	RETURN INFO (NOT YET USED)
1	*			
1	YVTCMODE	DS	C	MODE OF MESSAGE
1	YVTCLINE	EQU	C'L'	LINE MODE
1	YVTCEXT	EQU	C'E'	EXTENDED LINE
1	YVTCINFO	EQU	C'I'	INFO LINE MESSAGE
1	YVTCPHYS	EQU	C'P'	PYHSICAL MODE
1	YVCTTRAN	EQU	C'T'	TRANSPARENT MODE
1	YVTCFORM	EQU	C'F'	FORM MODE

1	YVTCCHIP	EQU	C 'C'	CHIPCARD MODE (FOR CKT)
1	YVTCMIXD	EQU	C 'M'	MIXED MODE
1	*			
1	YVTCHC	DS	C	HARCOPY FUNCTION
1	YVTCHCN	EQU	C 'N'	NO HARDCOPY
1	YVTCHCY	EQU	C 'Y'	LOCAL/CENTRAL HARDCOPY
1	*			
1	YVTCBEL	DS	C	BELL FUNCTION
1	YVTCBELN	EQU	C 'N'	NO BELL
1	YVTCBELY	EQU	C 'Y'	BELL AFTER OUTPUT
1	*			
1	YVTCNLG	DS	C	NO LOG CHARS TO INTERPRET FUNCTION
1	YVTCNLGN	EQU	C 'N'	LOGICAL CHARACTERS TO INTERPRET
1	YVTCNLGY	EQU	C 'Y'	NO LOGICAL CHARACTERS TO INTERPRET
1	*			
1	YVTCRBYT	DS	CL2	RETURN INFO BYTES
1	*			
1	YVTCRIN	DS	C	RETURN INFORMATION FUNCTION
1	YVTCRINN	EQU	C 'N'	NO RETURN INFORMATION
1	YVTCRINY	EQU	C 'Y'	RETURN INFORMATION REQUIRED
1	*			
1	YVTCLOC	DS	C	INPUT OF LOCAL CHARACTERS
1	YVTCLOCN	EQU	C 'N'	NO LOCAL CHARACTERS REQUIRED
1	YVTCLOCY	EQU	C 'Y'	LOCAL CHARACTERS REQUIRED
1	*			
1	YVTCOHD	DS	C	OUTPUT HEADER FUNCTION
1	YVTCOHDN	EQU	C 'N'	NO OUTPUT HEADER IN USER MSG
1	YVTCOHDY	EQU	C 'Y'	OUTPUT HEADER IN USER MESSAGE
1	*			
1	YVTCCTR	DS	C	CODE TRANSLATION FUNCTION
1	YVTCCTRN	EQU	C 'N'	NO CODE TRANSLATION DONE BY VTSU
1	YVTCCTRY	EQU	C 'Y'	CODE TRANSLATION TO/FROM CCS REQ.
1	*			
1	YVTCIHD	DS	C	INPUT HEADER FUNCTION
1	YVTCIHDN	EQU	C 'N'	NO INPUT HEADER REQUIRED
1	YVTCIHDY	EQU	C 'Y'	INPUT HEADER REQUIRED
1	*			
1	YVTCLOW	DS	C	LOWER CHARACTERS FUNCTION
1	YVTCLOWN	EQU	C 'N'	TRANSLATE LOWER CHARACTERS
1	YVTCLOWY	EQU	C 'Y'	RETAIN LOWER CHARACTERS
1	*			
1	YVTCSPIN	DS	C	SPECIAL INPUT FUNCTION
1	YVTCNSPI	EQU	C 'N'	NO SPECIAL INPUT
1	YVTCIDIN	EQU	C 'I'	INPUT FROM ID-CARD READER
1	YVTCCOIN	EQU	C 'C'	CONFIDENTIAL INPUT
1	*			
1	YVTCFC	DS	C	FUNCTION CODE
1	YVTCFCN	EQU	C 'N'	NO FUNCTION CODE REQUIRED
1	YVTCFCY	EQU	C 'Y'	FUNCTION CODE REQUIRED
1	*			
1	YVTCOM	DS	C	HOMOGENEOUS OUTPUT
1	YVTCOMN	EQU	C 'N'	NO HOMOGENEOUS OUTPUT REQUIRED
1	YVTCOMY	EQU	C 'Y'	HOMOGENEOUS OUTPUT REQUIRED
1	*			
1	YVTCNOP	DS	C	OUTPUT ON SAME LINE
1	YVTCNOPN	EQU	C 'N'	OUTPUT STARTS ON NEXT LINE
1	YVTCNOPY	EQU	C 'Y'	OUTPUT STARTS ON CURRENT LINE
1	*			

```

1 YVTCCNA      DS      CL8           CODED CHARACTER SET NAME
1 *
1 YVTCCUR      DS      C           CURSOR POSITION REQUESTED
1 YVTCCURN     EQU     C'N'         CURSOR POSITION NOT RETURNED
1 YVTCCURY     EQU     C'Y'         CURSOR POSITION GIVEN AFTER INPUT
1 *
1 YVTCPOSL     DS      XL1         CURSOR POSITION (LINE)
1 YVTCPOSC     DS      XL1         CURSOR POSITION (COLUMN)
1 *
1 YVTCREAD     DS      C           READ MODE (EXTENDED LINE MODE)
1 YVTCRDUN     EQU     C'U'         READ UNPROTECTED
1 YVTCRDMO     EQU     C'M'         READ MODIFIED
1 *
1 YVTCUPD      DS      C           SCREEN UPDATE IN EXTENDED LINE MODE
1 YVTCUPDN     EQU     C'N'         NO SCREEN UPDATE -> REFRESH
1 YVTCUPDY     EQU     C'Y'         SCREEN UPDATE
1 *
1 YVTCWAR      DS      C           WAR BYTE REQUESTED
1 YVTCWARN     EQU     C'N'         NO INFO ABOUT WAR BYTE
1 YVTCWARY     EQU     C'Y'         VALUE OF WAR BYTE TO RETURN
1 *
1 YVTCWARI     DS      XL1         RETURNED WAR BYTE VALUE
1 *
1 YVTCAT       DS      C           AUTOMATIC TABULATION
1 YVTCATS      EQU     C'S'         STANDARD AUTOMATIC TABULATION
1 YVTCATN      EQU     C'N'         AUTOMATIC TABULATION NOT REQUESTED
1 YVTCATY      EQU     C'Y'         AUTOMATIC TABULATION REQUESTED
1 *
1 YVTC#        EQU     *-YVTCFHE    LENGTH OF DSECT
1 * ,VTSUCB    050    921231    53531040
END

```


4 COBOL interface

This chapter describes how logical data display terminals support COBOL applications. The data structures provided for this purpose are **VTSUBAS**, **VTSUMONC**, **VTSUPERP**, **TIAMCTRC** and **VTSUCBC**.

The data structures **VTSUBAS**, **VTSUMONC** and **VTSUPERP** contain the status information supplied by VTSU. The **YINQUIRE** macro enables you to query the status information. Note that status information can be queried only for DCAM(NEA) transport service applications. The status information that can be queried is as follows:

- Description of the data display terminal and the character sets, e.g. type and number of logically addressable character sets, screen format
- Description of the peripherals connected to the terminal, e.g. ID card reader, chip card terminal
- Basic terminal information, e.g. type of data display terminal, number of colors

The **TIAMCTRC** data structure contains logical control characters. These logical control characters are converted into the device control characters for the specific line terminal. Logical control characters are:

- record control characters such as new line, new page, characters per line, etc.
- Display control characters for message display, such as highlight, select character set, line spacing, character spacing, etc.
- Miscellaneous functions such as delete, backspace, etc.

The **VTSUCBC** data structure allows you to define VTSU parameters for input and output independently of the access method. These I/O parameters correspond to the edit options, but you should note that new edit options will be available only via the **VTSUCB**. DCAM applications can include the **VTSUCB** in **YSEND**, **YRECEIVE** and **YSENDREC** macros, while TIAM applications can include it in **WROUT**, **WRTRD** and **RDATA** macros.

All data structures are in the **\$TSOS.SYSLIB.VTSU-B.110** library and can be copied into the application program by means of a copy macro.

4.1 Status information



This section applies to DCAM(NEA) transport service applications only.

The YINQUIRE macro can be used to query status information. The data structures at your disposal for this purpose are:

- VTSUBAS for basic information on the data terminal
- VTSUMONC for the description of the data display terminal and the character sets.
- VTSUPERP for the description of the peripherals connected to the terminal

Note that DCAM supplies the information on partner characteristics for DCAM-COBOL applications. Consequently, the status information on partner characteristics is described in the manual DCAM COBOL Macros.

4.1.1 Basic information: VTSUBAS

The copy member VTSUBAS contains the basic terminal information supplied by VTSU. It can be copied into the application program with **COPY VTSUBAS**.

```

*          VTSUBAS    999          921011    53531050          *
*****
* VTSUBAS   : V999
*****
*
*   30 STATUS-BASIC.
*
*                               PERIPHERIE
*
*   32 BASIC-PRESENT              PIC X.
*
*       88 BASIC-PRESENT-YES      VALUE "Y".
*                               STATUS INFO FROM TERMINAL
*   88 BASIC-PRESENT-NO          VALUE "N".
*                               NO STATUS INFO FROM TERMINAL
*   32 STATUS-POSSIBLE            PIC X.
*
*       88 STATUS-POSSIBLE-YES    VALUE "Y".
*                               STAT INF FROM TERMINAL POSSIBLE
*   88 STATUS-POSSIBLE-NO        VALUE "N".
*                               NO STAT INF FROM TERMINAL POSS
*
*   32 PRINT-DVTYP                PIC X(8).
*                               PRINTABLE DEVICE TYPE
*
*   32 LOCAL-HARDCOPY             PIC X.
*
*       88 LOCAL-HARDCOPY-YES     VALUE "Y".
*       88 LOCAL-HARDCOPY-NO     VALUE "N".
*
*   32 ID-CARD-READER            PIC X.
*
*       88 ID-CARD-READER-YES    VALUE "Y".
*       88 ID-CARD-READER-NO    VALUE "N".
*
*   32 SCREEN-COLOURS           PIC X.
*
*       88 SCREEN-COLOURS-NO     VALUE "N".
*       88 SCREEN-COLOURS-4     VALUE "4".
*       88 SCREEN-COLOURS-8     VALUE "8".
*
*   32 FILLER                     PIC X(3).
*
*   32 PHYS-LINES                 PIC X(4).
*                               NUMBER OF PHYSICAL LINES
*
*   32 PHYS-COLUMNS             PIC X(4).
*                               NUMBER OF PHYSICAL COLUMNS
*

```

```
*      32  TERMINAL-TYPE                PIC X.
*
*      88  TERMINAL-TYPE-7-BIT         VALUE "7".
*      88  TERMINAL-TYPE-8-BIT         VALUE "8".
*
*      32  CURRENT-CHAR-SET-NAME       PIC X(8).
*
*      32  SUPP-CHAR-SET-NUMBER        PIC X.
*
*      32  VARIANT-1                   PIC X.
*      32  VARIANT-2                   PIC X(2).
*      32  VARIANT-3                   PIC X(2).
*      32  VARIANT-4                   PIC X(2).
*      32  VARIANT-5                   PIC X(2).
*      32  VARIANT-6                   PIC X(2).
*      32  VARIANT-7                   PIC X(2).
*      32  VARIANT-8                   PIC X(2).
*      32  VARIANT-9                   PIC X(2).
*      32  VARIANT-10                  PIC X(2).
*      32  VARIANT-11                  PIC X(2).
*      32  VARIANT-12                  PIC X(2).
*      32  VARIANT-13                  PIC X(2).
*      32  VARIANT-14                  PIC X(2).
*      32  VARIANT-15                  PIC X(2).
*      32  VARIANT-16                  PIC X(2).
*
*      32  FILLER                       PIC X(2).
*
*      32  ACTIVE-CHAR-SET-NAME        PIC X(8).
*
*      32  OPERATING-MODE              PIC X.
*
*      88  READ-MODIFIED                VALUE "M".
*      88  READ-UNPROTECTED            VALUE "U".
*
*      32  LOGICAL-LINE-END-CHAR       PIC X.
*
*      32  SUBSTITUTE-CHAR             PIC X.
*
*      32  PERMANENT-HARDCOPY          PIC X.
*
*      88  PERMANENT-HARDCOPY          VALUE "Y".
*      88  NOT-PERMANENT-HARDCOPY     VALUE "N".
```


Description of fields**BASIC-PRESENT** Terminal status

- "Y" Status message from the terminal available.
"N" No status message from the terminal available.

STATUS-POSSIBLE Terminal status

- "Y" Status message from the terminal is possible.
"N" No status message from the terminal is expected.

PRINT-DVTYP Terminal type, printable

- "TYP00" Unknown device type
"SS-8103" 8103 Printer
"DSS-X.29" X.29 display terminal
"DSS-8150" 8150 Data Display Terminal
"RECHNER" Terminal/computer
"SS-8102" 8102 Printer Terminal
"DSS-8151" 8151 Data Display Terminal
"DSS-8152" 8152 Data Display Terminal
"SS-8110" 8110 Printer Terminal
"DRS-8121" 8121 Printer
"FS100" 100 Teleprinter
"FS100-E" 100-E Teleprinter
"DRS90037" 90037 Integrated Printer Terminal Controller
"DRS-8122" 8122 Printer
"DSS-8162" 8162 Data Display Terminal
"DSS-8160" 8160 Data Display Terminal
"DRS-8124" 8124 Printer
"AP" Application program
"SST-X.29" X.29 printer terminal
"DSS-9750" 9750 Data Display Terminal
"DRS-9003" 9003 Printer
"DSS-9770" 9770 Terminal
"DRS-9002" 9002 Printer
"DSS-3974" 3974 Display Unit
"DSS-9751" 9751 Data Display Terminal
"DSS-9752" 9752 Data Display Terminal
"DSS-9753" 9753 Data Display Terminal
"DRS-9001" 9001 Printer
"DSS-9731" 9731 Graphics Workstation
"DSS9770R" 9770R Terminal

"DRS-9004" 9004 Printer
"DSS-9754" 9754 Data Display Terminal
"DSS-9755" 9755 Data Display Terminal
"DSS-9763" 9763 Data Display Terminal
"DRS-9012" 9012 Printer
"DRS-9013" 9013 Printer
"DSS-3270" 3270 display terminal
"DRS-0131" 9001-31 Printer
"DRS-0189" 9001-8931 Printer
"DRS-9022" 9022 Electronic Page Printer
"DRS-1118" 9011-18 Printer
"DRS-1119" 9011-19 Printer
"DRS-3287" 3287 Printer
"TCP-IP" TCP-IP application
"DRS-9021" 9021 Printer
"DRS-9014" 9014 Printer
"DRS-9026" 9026 Printer (HDLC, 9025-compatible)
"DSS-FE" Front-end data display terminal (FHS-DOORS)

LOCAL-HARDCOPY Local hardcopy unit

"Y" Local hardcopy unit is connected
"N" No local hardcopy unit is connected

ID-CARD-READER ID card reader

"Y" ID card reader is connected
"N" No ID card reader is connected

SCREEN-COLOURS Number of colors at the terminal

"N" no colors
"4" 4 colors
"8" 8 colors

PHYS-LINES Physical number of lines, taken from the generation or default value.

PHYS-COLUMNS Physical number of characters per line, taken from the generation or default value.

TERMINAL-TYPE

Type of terminal

"7" Terminal operates in 7-bit mode only

"8" Terminal operates in 7-bit or 8-bit mode

CURRENT-CHAR-SET-NAME

Name of the extended standard character set This value is supplied only if the terminal supports 8-bit mode.

SUPP-CHAR-SET-NUMBER

Number of 8-bit characters sets supported

VARIANT-1

1st supported character set

VARIANT-2

2nd supported character set

VARIANT-3

3rd supported character set

VARIANT-4

4th supported character set

VARIANT-5

5th supported character set

VARIANT-6

6th supported character set

VARIANT-7

7th supported character set

VARIANT-8

8th supported character set

VARIANT-9

9th supported character set

VARIANT-10

10th supported character set

VARIANT-11

11th supported character set

VARIANT-12

12th supported character set

VARIANT-13

13th supported character set

VARIANT-14

14th supported character set

VARIANT-15

15th supported character set

VARIANT-16

16th supported character set

ACTIVE-CHAR-SET-NAME

Name of the activated extended character set

This value is supplied only if the terminal supports 8-bit mode.

OPERATING-MODE

physical read mode

"M" Only modified fields are read

"U" Only unprotected fields are read

LOGICAL-LINE-END-CHAR

Logical line end character for data display terminals without the equivalent hardware function

SUBSTITUTE-CHAR

Substitute character for characters lower than X'40' that are not logical control characters.

PERMANENT-HARDCOPY

Permanent hardcopy

"Y" All outputs are logged by additional hardcopy.

"N" Outputs are not logged by additional hardcopy.

4.1.2 Peripherals: VTSUPERP

The copy member VTSUPERP contains the description of the peripherals. It can be copied into the application program by means of **COPY VTSUPERP**.

```

*          VTSUPERP   999          921011   53531048          *
*****
*  VTSUPERP :  V999
*****
*
*  30 STATUS-PERIPHERIE.
*
*          PERIPHERIE
*
*  32 PERPH-PRESENT          PIC X.
*
*      88 PERPH-PRESENT-YES          VALUE "Y".
*          STATUS INFO FROM TERMINAL
*      88 PERPH-PRESENT-NO          VALUE "N".
*          NO STATUS INFO FROM TERMINAL
*
*  32 FILLER          PIC X(2).
*
*  32 LOC-HARDCOPY          PIC X.
*
*      88 LOC-HARDCOPY-YES          VALUE "Y".
*      88 LOC-HARDCOPY-NO          VALUE "N".
*
*  32 FILLER          PIC X(2).
*
*  32 ID-CARDREADER          PIC X.
*
*      88 ID-CARDREADER-YES          VALUE "Y".
*      88 ID-CARDREADER-NO          VALUE "N".
*
*  32 CHIPCARD-TERMINAL          PIC X.
*
*      88 CHIPCARD-TERMINAL-YES          VALUE "Y".
*      88 CHIPCARD-TERMINAL-NO          VALUE "N".
*

```

Description of fields**PERH-PRESENT** Terminal status

- "Y" Status message from the terminal
- "N" No status message from the terminal

LOC-HARDCOPY Local hardcopy unit

- "Y" Local hardcopy unit connected
- "N" No local hardcopy unit connected

ID-CARDREADER ID card reader

- "Y" ID card reader connected
- "N" No ID card reader connected

CHIPCARD-TERMINAL Chip card terminal

- "Y" Chip card terminal connected
- "N" No chip card terminal connected

4.1.3 Data display terminal and character sets: VTSUMONC

The copy member VTSUMONC contains the description of the terminal type and the character sets. It can be copied into the application program by means of **COPY VTSUMONC**.

```

*          VTSUMONC  999          921011  53531046          *
*****
*  VTSUMONC :  V999
*****
*
*  30 STATUS-MONITOR-AND-CHARSETS.
*
*
*  32 MONCS-PRESENT          PIC X.
*                               STAT INF PRES
*  88 MONCS-PRESENT-YES     VALUE "Y".
*                               STATUS INFO PRESENT
*  88 MONCS-PRESENT-NO     VALUE "N".
*                               NO STATUS INFO
*
*  32 MONTYPE              PIC X.
*                               MONITOR TYPE
*  88 MONOCHROM-SCREEN     VALUE "M".
*                               MONOCHROM SCREEN
*  88 COLOUR-SCREEN        VALUE "C".
*                               COLOURED SCREEN
*  88 PRINTER              VALUE "P".
*                               PRINTER DEVICE
*
*  32 FAT                  PIC X.
*                               STAT INF PRES
*  88 FAT-YES              VALUE "Y".
*                               NEW FIELD ATTR POSS
*  88 FAT-NO               VALUE "N".
*                               NO NEW FIELD ATTRIBUTES
*
*  32 FILLER                PIC X(1).
*                               RESERVED
*
*  32 DIM1                  PIC X.
*                               DIMENSION 24 X 80
*  88 DIM1-YES            VALUE "Y".
*                               DIM 24 X 80 POSSIBLE
*  88 DIM1-NO             VALUE "N".
*                               DIM 24 X 80 NOT POSS
*
*  32 DIM2                  PIC X.
*                               DIMENSION 32 X 80
*  88 DIM2-YES            VALUE "Y".
*                               DIM 32 X 80 POSSIBLE
*  88 DIM2-NO             VALUE "N".
*                               DIM 32 X 80 NOT POSS
*
*  32 DIM3                  PIC X.
*                               DIMENSION 43 X 80

```

```

      88 DIM3-YES                VALUE "Y".
*                               DIM 43 X 80 POSSIBLE
      88 DIM3-NO                VALUE "N".
*                               DIM 43 X 80 NOT POSS
*
32 DIM4                        PIC X.
*                               DIMENSION 27 X 132
      88 DIM4-YES              VALUE "Y".
*                               DIM 27 X 132 POSSIBLE
      88 DIM4-NO              VALUE "N".
*                               DIM 27 X 132 NOT POSS
*
32 FILLER                      PIC X(4).
*                               RESERVED
*
32 ZEG-NUMBER                  PIC 9(4) COMP.
*                               NUM OF US CHAR SETS
*
32 ZEG0-TYPE                   PIC X.
*                               TYP OF CHARSET 0
      88 ZEG0-SINGLE-PLANE     VALUE "S".
*                               SINGLE PLANE CHARS
      88 ZEG0-TRIPLE-PLANE   VALUE "T".
*                               TRIPLE PLANE CHARS
      88 ZEG0-NOT-USABLE     VALUE "N".
*                               NOT USABLE CHAR SET
*
32 ZEG0-STATUS                 PIC X.
*                               STATUS OF CHARSET 0
      88 ZEG0-NOT-LOADED     VALUE "0".
*                               CHARSET NOT LOADED
      88 ZEG0-DSS-LOADASSIGN VALUE "1".
*                               CHARSET USED BY DSS
      88 ZEG0-DVA-LOADED     VALUE "2".
*                               CHARSET LOADED
      88 ZEG0-DVA-LOADASSIGN VALUE "3".
*                               CHARS LOADED+ASSIGNED
*
32 ZEG1-TYPE                   PIC X.
*                               TYP OF CHARSET 1
      88 ZEG1-SINGLE-PLANE     VALUE "S".
*                               SINGLE PLANE CHARS
      88 ZEG1-TRIPLE-PLANE   VALUE "T".
*                               TRIPLE PLANE CHARS
      88 ZEG1-NOT-USABLE     VALUE "N".
*                               NOT USABLE CHAR SET
*
32 ZEG1-STATUS                 PIC X.
*                               STATUS OF CHARSET 1
      88 ZEG1-NOT-LOADED     VALUE "0".
*                               CHARSET NOT LOADED
      88 ZEG1-DSS-LOADASSIGN VALUE "1".
*                               CHARSET USED BY DSS
      88 ZEG1-DVA-LOADED     VALUE "2".
*                               CHARSET LOADED
      88 ZEG1-DVA-LOADASSIGN VALUE "3".
*                               CHARS LOADED+ASSIGNED
*

```



```

*      32 ZEG2-TYPE                PIC X.
*                                     TYP OF CHARSET 2
*      88 ZEG2-SINGLE-PLANE        VALUE "S".
*                                     SINGLE PLANE CHARS
*      88 ZEG2-TRIPLE-PLANE       VALUE "T".
*                                     TRIPLE PLANE CHARS
*      88 ZEG2-NOT-USABLE         VALUE "N".
*                                     NOT USABLE CHAR SET
*
*      32 ZEG2-STATUS              PIC X.
*                                     STATUS OF CHARSET 2
*      88 ZEG2-NOT-LOADED         VALUE "0".
*                                     CHARSET NOT LOADED
*      88 ZEG2-DSS-LOADASSIGN     VALUE "1".
*                                     CHARSET USED BY DSS
*      88 ZEG2-DVA-LOADED         VALUE "2".
*                                     CHARSET LOADED
*      88 ZEG2-DVA-LOADASSIGN     VALUE "3".
*                                     CHARS LOADED+ASSIGNED
*
*      32 ZEG3-TYPE                PIC X.
*                                     TYP OF CHARSET 3
*      88 ZEG3-SINGLE-PLANE        VALUE "S".
*                                     SINGLE PLANE CHARS
*      88 ZEG3-TRIPLE-PLANE       VALUE "T".
*                                     TRIPLE PLANE CHARS
*      88 ZEG3-NOT-USABLE         VALUE "N".
*                                     NOT USABLE CHAR SET
*
*      32 ZEG3-STATUS              PIC X.
*                                     STATUS OF CHARSET 3
*      88 ZEG3-NOT-LOADED         VALUE "0".
*                                     CHARSET NOT LOADED
*      88 ZEG3-DSS-LOADASSIGN     VALUE "1".
*                                     CHARSET USED BY DSS
*      88 ZEG3-DVA-LOADED         VALUE "2".
*                                     CHARSET LOADED
*      88 ZEG3-DVA-LOADASSIGN     VALUE "3".
*                                     CHARS LOADED+ASSIGNED
*
*      32 ZEG4-TYPE                PIC X.
*                                     TYP OF CHARSET 4
*      88 ZEG4-SINGLE-PLANE        VALUE "S".
*                                     SINGLE PLANE CHARS
*      88 ZEG4-TRIPLE-PLANE       VALUE "T".
*                                     TRIPLE PLANE CHARS
*      88 ZEG4-NOT-USABLE         VALUE "N".
*                                     NOT USABLE CHAR SET
*
*      32 ZEG4-STATUS              PIC X.
*                                     STATUS OF CHARSET 4
*      88 ZEG4-NOT-LOADED         VALUE "0".
*                                     CHARSET NOT LOADED
*      88 ZEG4-DSS-LOADASSIGN     VALUE "1".
*                                     CHARSET USED BY DSS
*      88 ZEG4-DVA-LOADED         VALUE "2".
*                                     CHARSET LOADED
*      88 ZEG4-DVA-LOADASSIGN     VALUE "3".

```

```
*                                     CHARS LOADED+ASSIGNED
*
32  ZEG5-TYPE                          PIC X.
*                                     TYP OF CHARSET 5
      88  ZEG5-SINGLE-PLANE              VALUE "S".
*                                     SINGLE PLANE CHARS
      88  ZEG5-TRIPLE-PLANE             VALUE "T".
*                                     TRIPLE PLANE CHARS
      88  ZEG5-NOT-USABLE               VALUE "N".
*                                     NOT USABLE CHAR SET
*
32  ZEG5-STATUS                          PIC X.
*                                     STATUS OF CHARSET 5
      88  ZEG5-NOT-LOADED               VALUE "0".
*                                     CHARSET NOT LOADED
      88  ZEG5-DSS-LOADASSIGN           VALUE "1".
*                                     CHARSET USED BY DSS
      88  ZEG5-DVA-LOADED               VALUE "2".
*                                     CHARSET LOADED
      88  ZEG5-DVA-LOADASSIGN           VALUE "3".
*                                     CHARS LOADED+ASSIGNED
*
32  ZEG6-TYPE                          PIC X.
*                                     TYP OF CHARSET 6
      88  ZEG6-SINGLE-PLANE              VALUE "S".
*                                     SINGLE PLANE CHARS
      88  ZEG6-TRIPLE-PLANE             VALUE "T".
*                                     TRIPLE PLANE CHARS
      88  ZEG6-NOT-USABLE               VALUE "N".
*                                     NOT USABLE CHAR SET
*
32  ZEG6-STATUS                          PIC X.
*                                     STATUS OF CHARSET 6
      88  ZEG6-NOT-LOADED               VALUE "0".
*                                     CHARSET NOT LOADED
      88  ZEG6-DSS-LOADASSIGN           VALUE "1".
*                                     CHARSET USED BY DSS
      88  ZEG6-DVA-LOADED               VALUE "2".
*                                     CHARSET LOADED
      88  ZEG6-DVA-LOADASSIGN           VALUE "3".
*                                     CHARS LOADED+ASSIGNED
*
32  ZEG7-TYPE                          PIC X.
*                                     TYP OF CHARSET 7
      88  ZEG7-SINGLE-PLANE              VALUE "S".
*                                     SINGLE PLANE CHARS
      88  ZEG7-TRIPLE-PLANE             VALUE "T".
*                                     TRIPLE PLANE CHARS
      88  ZEG7-NOT-USABLE               VALUE "N".
*                                     NOT USABLE CHAR SET
*
32  ZEG7-STATUS                          PIC X.
*                                     STATUS OF CHARSET 7
      88  ZEG7-NOT-LOADED               VALUE "0".
*                                     CHARSET NOT LOADED
      88  ZEG7-DSS-LOADASSIGN           VALUE "1".
*                                     CHARSET USED BY DSS
      88  ZEG7-DVA-LOADED               VALUE "2".
```

```
*                               CHARSET LOADED
      88  ZEG7-DVA-LOADASSIGN    VALUE "3".
*                               CHARS LOADED+ASSIGNED
```

Description of fields

MONCS-PRESENT Terminal status

- "Y" Status message from the terminal available. The following messages are taken from the status report.
- "N" No status message from the terminal available. The following values are the VTSU default values.

MONTYPE

Type of monitor

- "M" Terminal with monochrome screen
- "C" Terminal with color screen
- "P" Printer

FAT New field attributes

- "Y" Field attributes can be used.
- "N" Field attributes cannot be used.

DIM1 Screen dimensions 24 lines x 80 characters

- "Y" The format is supported.
- "N" The format is not supported.

DIM2 Screen dimensions 32 lines x 80 characters

- "Y" The format is supported.
- "N" The format is not supported.

DIM3 Screen dimensions 43 lines x 80 characters

- "Y" The format is supported.
- "N" The format is not supported.

DIM4 Screen dimensions 27 lines x 132 characters

"Y" The format is supported.

"N" The format is not supported.

ZEG-NUMBER Number of addressable character sets (max. 8). The character sets are numbered ZEG0 through ZEG7. The two following specifications ZEGx-TYPE and ZEGx-STATUS apply to each of the 8 character sets; x stands for the character set number in each case, e.g. ZEG5-TYPE = type of character set no. 5

ZEGx-TYPE Type of character set x

"S" Character set x is a loadable monochrome character set.

"T" Character set x is a loadable color character set.

"N" Character set x is a non-loadable character set

ZEGx-STATUS Status of character set x

"0" Character set x is not loaded.

"1" Character set x has been assigned by the display terminal.

"2" Character set x has been loaded by the dp system.

"3" Character set x has been loaded and assigned by the dp system.

4.2 Logical control characters: TIAMCTRC

The TIAMCTRC data structure contains logical control characters for input and output in line mode (virtual line terminal) and the logical printer control characters. You can copy the data structure into the application program by means of **COPY TIAMCTRC**.

The symbolic names of these logical control characters are shaded gray in the following listing. These symbolic names must be used so that logical control characters can be inserted in line mode output messages or located in line mode input messages.

The control characters **SELECT-COLOR**, **REPEAT-SYMBOL**, **DISPLAY-ATTR** and **FIELD-CHARACTERISTICS** are new.

```

COPY TIAMCTRC

*          TIAMCTRC   001          921231   53531052          *
*****
*
*          TIAMCTRC   V923          *
*
*                   V050          VTSU-B V11          *
*****
*
01  LINE-MODE-CONTROL-CHARACTERS.
*
*          LOGICAL RECORD DELIMITERS
*
41  CC-NEW-LINE          PIC 9(4)  COMP SYNC  VALUE 21.
41  CC-NEW-LINE-RED          REDEFINES CC-NEW-LINE.
   42  FILLER          PIC X.
   42  NEW-LINE          PIC X.
41  CC-NEW-PAGE          PIC 9(4)  COMP SYNC  VALUE 12.
41  CC-NEW-PAGE-RED          REDEFINES CC-NEW-PAGE.
   42  FILLER          PIC X.
   42  NEW-PAGE          PIC X.
41  CC-CONT-SAME-LINE    PIC 9(4)  COMP SYNC  VALUE 13.
41  CC-CONT-SAME-LINE-RED REDEFINES CC-CONT-SAME-LINE.
   42  FILLER          PIC X.
   42  CONT-SAME-LINE    PIC X.
41  CC-CONT-LINE-N      PIC 9(4)  COMP SYNC  VALUE 41.
41  CC-CONT-LINE-N-RED  REDEFINES CC-CONT-LINE-N.
   42  FILLER          PIC X.
   42  CONT-LINE-N      PIC X.
41  CC-CONT-COL-N      PIC 9(4)  COMP SYNC  VALUE 42.
41  CC-CONT-COL-N-RED  REDEFINES CC-CONT-COL-N.
   42  FILLER          PIC X.
   42  CONT-COL-N      PIC X.
41  CC-SHEET-FEED-N     PIC 9(4)  COMP SYNC  VALUE 33.
41  CC-SHEET-FEED-N-RED REDEFINES CC-SHEET-FEED-N.
   42  FILLER          PIC X.
   42  SHEET-FEED-N     PIC X.
41  CC-CONT-ACT-POS     PIC 9(4)  COMP SYNC  VALUE 32.
41  CC-CONT-ACT-POS-RED REDEFINES CC-CONT-ACT-POS.
   42  FILLER          PIC X.

```

```

      42 CONT-ACT-POS      PIC X.
*
* LOGICAL UNIT DELIMITERS
*
41 CC-EMPH-LAYOUT1      PIC 9(4) COMP SYNC VALUE 29.
41 CC-EMPH-LAYOUT1-RED  REDEFINES CC-EMPH-LAYOUT1.
   42 FILLER             PIC X.
   42 EMPH-LAYOUT1      PIC X.
41 CC-EMPH-LAYOUT2      PIC 9(4) COMP SYNC VALUE 31.
41 CC-EMPH-LAYOUT2-RED  REDEFINES CC-EMPH-LAYOUT2.
   42 FILLER             PIC X.
   42 EMPH-LAYOUT2      PIC X.
41 CC-EMPH-LAYOUT3      PIC 9(4) COMP SYNC VALUE 19.
41 CC-EMPH-LAYOUT3-RED  REDEFINES CC-EMPH-LAYOUT3.
   42 FILLER             PIC X.
   42 EMPH-LAYOUT3      PIC X.
41 CC-EMPH-LAYOUT4      PIC 9(4) COMP SYNC VALUE 20.
41 CC-EMPH-LAYOUT4-RED  REDEFINES CC-EMPH-LAYOUT4.
   42 FILLER             PIC X.
   42 EMPH-LAYOUT4      PIC X.
41 CC-NORMAL-LAYOUT     PIC 9(4) COMP SYNC VALUE 30.
41 CC-NORMAL-LAYOUT-RED REDEFINES CC-NORMAL-LAYOUT.
   42 FILLER             PIC X.
   42 NORMAL-LAYOUT     PIC X.
41 CC-DARK-LAYOUT       PIC 9(4) COMP SYNC VALUE 18.
41 CC-DARK-LAYOUT-RED   REDEFINES CC-DARK-LAYOUT.
   42 FILLER             PIC X.
   42 DARK-LAYOUT       PIC X.
41 CC-PART-LINE-UP      PIC 9(4) COMP SYNC VALUE 44.
41 CC-PART-LINE-UP-RED  REDEFINES CC-PART-LINE-UP.
   42 FILLER             PIC X.
   42 PART-LINE-UP      PIC X.
41 CC-PART-LINE-DOWN    PIC 9(4) COMP SYNC VALUE 43.
41 CC-PART-LINE-DOWN-RED REDEFINES CC-PART-LINE-DOWN.
   42 FILLER             PIC X.
   42 PART-LINE-DOWN    PIC X.
*
41 CC-SECOND-CHAR-SET   PIC 9(4) COMP SYNC VALUE 14.
41 CC-SECOND-CHAR-SET-RED REDEFINES CC-SECOND-CHAR-SET.
   42 FILLER             PIC X.
   42 SECOND-CHAR-SET   PIC X.
41 CC-NORMAL-CHAR-SET   PIC 9(4) COMP SYNC VALUE 15.
41 CC-NORMAL-CHAR-SET-RED REDEFINES CC-NORMAL-CHAR-SET.
   42 FILLER             PIC X.
   42 NORMAL-CHAR-SET   PIC X.
*
41 CC-START-PROT-AREA   PIC 9(4) COMP SYNC VALUE 54.
41 CC-START-PROT-AREA-RED REDEFINES CC-START-PROT-AREA.
   42 FILLER             PIC X.
   42 START-PROT-AREA   PIC X.
41 CC-END-PROT-AREA     PIC 9(4) COMP SYNC VALUE 8.
41 CC-END-PROT-AREA-RED REDEFINES CC-END-PROT-AREA.
   42 FILLER             PIC X.
   42 END-PROT-AREA     PIC X.
41 CC-START-NUM-DATA    PIC 9(4) COMP SYNC VALUE 17.
41 CC-START-NUM-DATA-RED REDEFINES CC-START-NUM-DATA.
   42 FILLER             PIC X.
   42 START-NUM-DATA    PIC X.

```

```

41 CC-START-CHARSET-N PIC 9(4) COMP SYNC VALUE 6.
41 CC-START-CHARSET-N-RED REDEFINES CC-START-CHARSET-N.
   42 FILLER PIC X.
   42 START-CHARSET-N PIC X.
41 CC-SELECT-COLOUR PIC 9(4) COMP SYNC VALUE 23.
41 CC-SELECT-COLOUR-RED REDEFINES CC-SELECT-COLOUR.
   42 FILLER PIC X.
   42 SELECT-COLOUR PIC X.
*
41 DIMENSION-24-80 PIC X(6) VALUE "D24080".
41 DIMENSION-32-80 PIC X(6) VALUE "D32080".
41 DIMENSION-43-80 PIC X(6) VALUE "D43080".
41 DIMENSION-27-132 PIC X(6) VALUE "D27132".
*
41 CC-START-LOCAL-ATTR PIC 9(4) COMP SYNC VALUE 9.
41 CC-START-LOCAL-ATTR-RED REDEFINES CC-START-LOCAL-ATTR.
   42 FILLER PIC X.
   42 START-LOCAL-ATTR PIC X.
41 CC-END-LOCAL-ATTR PIC 9(4) COMP SYNC VALUE 10.
41 CC-END-LOCAL-ATTR-RED REDEFINES CC-END-LOCAL-ATTR.
   42 FILLER PIC X.
   42 END-LOCAL-ATTR PIC X.
*
41 CC-VERT-MOVE-IND PIC 9(4) COMP SYNC VALUE 36.
41 CC-VERT-MOVE-IND-RED REDEFINES CC-VERT-MOVE-IND.
   42 FILLER PIC X.
   42 VERT-MOVE-IND PIC X.
41 CC-HORIZ-MOVE-IND PIC 9(4) COMP SYNC VALUE 35.
41 CC-HORIZ-MOVE-IND-RED REDEFINES CC-HORIZ-MOVE-IND.
   42 FILLER PIC X.
   42 HORIZ-MOVE-IND PIC X.
41 CC-LEFT-MARGIN PIC 9(4) COMP SYNC VALUE 56.
41 CC-LEFT-MARGIN-RED REDEFINES CC-LEFT-MARGIN.
   42 FILLER PIC X.
   42 LEFT-MARGIN PIC X.
41 CC-START-PROP-TYPE PIC 9(4) COMP SYNC VALUE 26.
41 CC-START-PROP-TYPE-RED REDEFINES CC-START-PROP-TYPE.
   42 FILLER PIC X.
   42 START-PROP-TYPE PIC X.
41 CC-END-PROP-TYPE PIC 9(4) COMP SYNC VALUE 27.
41 CC-END-PROP-TYPE-RED REDEFINES CC-END-PROP-TYPE.
   42 FILLER PIC X.
   42 END-PROP-TYPE PIC X.
41 CC-MAX-LINE-LEN PIC 9(4) COMP SYNC VALUE 51.
41 CC-MAX-LINE-LEN-RED REDEFINES CC-MAX-LINE-LEN.
   42 FILLER PIC X.
   42 MAX-LINE-LEN PIC X.
41 CC-MAX-LINE-NUM PIC 9(4) COMP SYNC VALUE 53.
41 CC-MAX-LINE-NUM-RED REDEFINES CC-MAX-LINE-NUM.
   42 FILLER PIC X.
   42 MAX-LINE-NUM PIC X.
41 CC-START-NEAR-LETT PIC 9(4) COMP SYNC VALUE 57.
41 CC-START-NEAR-LETT-RED REDEFINES CC-START-NEAR-LETT.
   42 FILLER PIC X.
   42 START-NEAR-LETT PIC X.
41 CC-END-NEAR-LETT PIC 9(4) COMP SYNC VALUE 59.
41 CC-END-NEAR-LETT-RED REDEFINES CC-END-NEAR-LETT.
   42 FILLER PIC X.

```

```

      42 END-NEAR-LETT    PIC X.
*
*   SPECIAL FUNCTIONS
*
41  CC-EXTEND-LOG-CHAR   PIC 9(4)  COMP SYNC  VALUE 62.
41  CC-EXTEND-LOG-CHAR-RED  REDEFINES CC-EXTEND-LOG-CHAR.
      42 FILLER          PIC X.
      42 EXTEND-LOG-CHAR   PIC X.
41  CC-DELETE-CHAR      PIC 9(4)  COMP SYNC  VALUE 7.
41  CC-DELETE-CHAR-RED   REDEFINES CC-DELETE-CHAR.
      42 FILLER          PIC X.
      42 DELETE-CHAR      PIC X.
41  CC-BACKSPACE        PIC 9(4)  COMP SYNC  VALUE 22.
41  CC-BACKSPACE-RED    REDEFINES CC-BACKSPACE.
      42 FILLER          PIC X.
      42 BACKSPACE        PIC X.
41  CC-SUBSTITUTE       PIC 9(4)  COMP SYNC  VALUE 63.
41  CC-SUBSTITUTE-RED   REDEFINES CC-SUBSTITUTE.
      42 FILLER          PIC X.
      42 SUBSTITUTE       PIC X.
*  TRANSPARENT OUTPUT
      41 TRANSPARENT-OUTPUT PIC X  VALUE "T".
*
*   PHYSICAL UNIT DELIMITERS
*
41  CC-PHYS-ESC         PIC 9(4)  COMP SYNC  VALUE 39.
41  CC-PHYS-ESC-RED     REDEFINES CC-PHYS-ESC.
      42 FILLER          PIC X.
      42 PHYS-ESC        PIC X.
41  CC-PHYS-DC4        PIC 9(4)  COMP SYNC  VALUE 60.
41  CC-PHYS-DC4-RED    REDEFINES CC-PHYS-DC4.
      42 FILLER          PIC X.
      42 PHYS-DC4       PIC X.
41  CC-PHYS-HT         PIC 9(4)  COMP SYNC  VALUE 5.
41  CC-PHYS-HT-RED     REDEFINES CC-PHYS-HT.
      42 FILLER          PIC X.
      42 PHYS-HT        PIC X.
41  CC-PHYS-VT         PIC 9(4)  COMP SYNC  VALUE 11.
41  CC-PHYS-VT-RED     REDEFINES CC-PHYS-VT.
      42 FILLER          PIC X.
      42 PHYS-VT        PIC X.
*
      41 CC-REPEAT-SYMBOL PIC X  VALUE "R".
*
*   DISPLAY ATTRIBUTES
*
41  CC-DISPLAY-ATTR    PIC X  VALUE "I".
41  CC-FLASHING        PIC 9(4)  COMP SYNC  VALUE 1.
41  CC-FLASHING-RED    REDEFINES CC-FLASHING.
      42 FILLER          PIC X.
      42 FLASHING        PIC X.
41  CC-UNDERSCORED    PIC 9(4)  COMP SYNC  VALUE 2.
41  CC-UNDERSCORED-RED REDEFINES CC-UNDERSCORED.
      42 FILLER          PIC X.
      42 UNDERSCORED    PIC X.
41  CC-BLANKED         PIC 9(4)  COMP SYNC  VALUE 4.
41  CC-BLANKED-RED     REDEFINES CC-BLANKED.
      42 FILLER          PIC X.

```



```

      42 BLANKED          PIC X.
41  CC-REDUCED-INTENSITY PIC 9(4) COMP SYNC VALUE 8.
41  CC-REDUCED-INT-RED  REDEFINES CC-REDUCED-INTENSITY.
      42 FILLER          PIC X.
      42 REDUCED-INTENSITY PIC X.
41  CC-INVERSE-MODE     PIC 9(4) COMP SYNC VALUE 16.
41  CC-INVERSE-MODE-RED REDEFINES CC-INVERSE-MODE.
      42 FILLER          PIC X.
      42 INVERSE-MODE    PIC X.
41  CC-RESET           PIC 9(4) COMP SYNC VALUE 0.
41  CC-RESET-RED       REDEFINES CC-RESET.
      42 FILLER          PIC X.
      42 RESET           PIC X.
*
41  CC-FIELD-CHARACTERISTICS PIC X VALUE "F".
41  CC-PROT-NOT-SENDABLE PIC 9(4) COMP SYNC VALUE 1.
41  CC-PROT-NOT-SEND-RED  REDEFINES CC-PROT-NOT-SENDABLE.
      42 FILLER          PIC X.
      42 PROTECTED-NOT-SENDABLE PIC X.
41  CC-PROT-SENDABLE PIC 9(4) COMP SYNC VALUE 32.
41  CC-PROT-SENDABLE-RED REDEFINES CC-PROT-SENDABLE.
      42 FILLER          PIC X.
      42 PROTECTED-SENDABLE PIC X.
41  CC-NUMERIC         PIC 9(4) COMP SYNC VALUE 2.
41  CC-NUMERIC-RED     REDEFINES CC-NUMERIC.
      42 FILLER          PIC X.
      42 NUMERIC         PIC X.
41  CC-PREMODIFIED     PIC 9(4) COMP SYNC VALUE 4.
41  CC-PREMODIFIED-RED REDEFINES CC-PREMODIFIED.
      42 FILLER          PIC X.
      42 PREMODIFIED     PIC X.
41  CC-MARKABLE        PIC 9(4) COMP SYNC VALUE 8.
41  CC-MARKABLE-RED    REDEFINES CC-MARKABLE.
      42 FILLER          PIC X.
      42 MARKABLE        PIC X.
41  CC-PRINTABLE       PIC 9(4) COMP SYNC VALUE 16.
41  CC-PRINTABLE-RED   REDEFINES CC-PRINTABLE.
      42 FILLER          PIC X.
      42 PRINTABLE       PIC X.
41  CC-INPUT           PIC 9(4) COMP SYNC VALUE 0.
41  CC-INPUT-RED       REDEFINES CC-INPUT.
      42 FILLER          PIC X.
      42 INPUT           PIC X.
41  CC-AUTOSKIP        PIC 9(4) COMP SYNC VALUE 64.
41  CC-AUTOSKIP-RED    REDEFINES CC-AUTOSKIP.
      42 FILLER          PIC X.
      42 AUTOSKIP        PIC X.
*
*
```

How to use these control characters

Before each output item which is to be controlled with one of these control characters, an item with a length of 1 or more bytes, depending on the control character, should be defined with PIC X (see data structure TIAMCTRC). The desired character can then be transferred to this item from the data structure LINE-MODE-CONTROL-CHARACTERS (see example below) by means of a MOVE statement.

Meaning of the control characters in TIAMCTRC:

NEW-LINE	<p>The logical end-of-line character is output and the cursor set to the start of the next line. Special forms of display are reset to normal (normal, standard color, 1st character set, unprotected)</p> <p>For more information, see the control character NL in the VTCSET Assembler macro.</p>
NEW-PAGE	<p>The logical end-of-line character is output, a new page initialized and the cursor set to the start of the page. The screen is set to the standard format. Special forms of display are reset (normal, standard color, standard character set, unprotected).</p> <p>For more information, see the control character NP in the VTCSET Assembler macro.</p>
CONT-SAME-LINE	<p>The logical end-of-line character is output and the cursor set to the start of the current line (no line feed). Special forms of display are reset to normal. This applies only to printers and printer terminals.</p> <p>For more information, see the control character CL in the VTCSET Assembler macro.</p>
CONT-LINE-N	<p>The cursor is set to the beginning of line n (absolute). n is specified as a 3-digit printable decimal character following the control character.</p> <p>For more information, see the control character VPA in the VTCSET Assembler macro.</p>
EMPH-LAYOUT1	<p>The subsequent text characters are highlighted on the monitor (see table at end of this section).</p> <p>For more information, see the control character EM1 in the VTCSET Assembler macro.</p>

EMPH-LAYOUT2	<p>The subsequent text characters are highlighted on the monitor (see table at end of this section).</p> <p>For more information, see the control character EM2 in the VTCSET Assembler macro.</p>
EMPH-LAYOUT3	<p>The subsequent text characters are highlighted on the monitor (see table at end of this section).</p> <p>For more information, see the control character EM3 in the VTCSET Assembler macro.</p>
EMPH-LAYOUT4	<p>The subsequent text characters are highlighted on the monitor (see table at end of this section).</p> <p>For more information, see the control character EM4 in the VTCSET Assembler macro.</p>
NORMAL-LAYOUT	<p>The subsequent text characters are displayed normally on the monitor.</p> <p>For more information, see the control character NOR in the VTCSET Assembler macro.</p>
DARK-LAYOUT	<p>The subsequent characters are blanked out on the monitor and are not printable.</p> <p>Note that support for this character is continued for the sake of compatibility only. Use instead the control-character string:</p> <pre>EXTENDED-LOG-CHAR DISPLAY-ATTR BLANKED text</pre> <p>For more information, see the control character DAR in the VTCSET Assembler macro.</p>
SECOND-CHAR-SET	<p>The subsequent text characters are displayed using character set 2 (only terminals with 2nd character set or with APL capability, and printers).</p> <p>For more information, see the control character SO in the VTCSET Assembler macro.</p>
NORMAL-CHAR-SET	<p>Subsequent text characters are displayed on the terminal in character set 1.</p> <p>For more information, see the control character SI in the VTCSET Assembler macro.</p>

START-PROT-AREA	<p>Subsequent text characters are output on the monitor at low intensity and protected, i.e. they cannot be overwritten and transferred to the host computer.</p> <p>For more information, see the control character SPA in the VTCSET Assembler macro.</p>
END-PROT-AREA	<p>Subsequent text characters are output to the terminal as unprotected characters and are displayed at normal intensity.</p> <p>For more information, see the control character EPA in the VTCSET Assembler macro.</p>
START-NUM-DATA	<p>Only numerics can be entered in the unprotected field. Numerics are digits and the characters . * / + -.</p> <p>For more information, see the control character NUM in the VTCSET Assembler macro.</p>
START-CHARSET-N	<p>Subsequent text characters are displayed in a selectable character set. The character set is specified by a 2-digit decimal number (00-07) input after the control character. START-CHARSET-N affects only 9763 Data Display Terminals.</p> <p>For more information, see the control character CHS in the VTCSET Assembler macro.</p>
START-LOCAL-ATTR	<p>Creates local attributes on 9763 Data Display Terminals. The attribute specified after the control code is only created locally.</p> <p>For more information, see the control character LOC in the VTCSET Assembler macro.</p>
END-LOCAL-ATTR	<p>The local attributes are reset to the last valid non-local value.</p> <p>For more information, see the control character LOX in the VTCSET Assembler macro.</p>
DELETE-CHAR	<p>This character is removed from the output text and is not transmitted to the terminal.</p> <p>For more information, see the control character DEL in the VTCSET Assembler macro.</p>
BACKSPACE	<p>The next text character is superimposed on its predecessor. In this way a character not contained in the character set can be created (only on terminals with APL capability and on printers).</p> <p>For more information, see the control character BS in the VTCSET Assembler macro.</p>

SUBSTITUTE	<p>This control character is represented at the terminal by the valid substitute character.</p> <p>For more information, see the control character SUB in the VTCSET Assembler macro.</p>																
PHYS-ESC	<p>This character, together with the next one, is transferred in unchanged form (physical control character). Column and line counts are suspended.</p> <p>For more information, see the control character ESC in the VTCSET Assembler macro.</p>																
PHYS-DC4	<p>This character, together with the next one, is transferred in unchanged form (physical control character). Column and line counts are suspended.</p> <p>For more information, see the control character DC4 in the VTCSET Assembler macro.</p>																
PHYS-HT	<p>This character, which is transferred in unchanged form, effects a horizontal tab. Column and line counts are suspended.</p> <p>For more information, see the control character HT in the VTCSET Assembler macro.</p>																
PHYS-VT	<p>This character, which is transferred in unchanged form, effects a vertical tab. Column and line counts are suspended.</p> <p>For more information, see the control character VT in the VTCSET Assembler macro.</p>																
SELECT-COLOR	<p>SELECT-COLOR selects a color for the field on a 9763 Color Data Display Terminal. A two-digit decimal number (00-07) after the control character specifies the desired color. SELECT-COLOR affects only the 9763 Color Data Display Terminal. It creates a start of field.</p> <table><tr><td>00</td><td>standard color</td><td>04</td><td>green</td></tr><tr><td>01</td><td>blue</td><td>05</td><td>cyan</td></tr><tr><td>02</td><td>red</td><td>06</td><td>yellow</td></tr><tr><td>03</td><td>magenta</td><td>07</td><td>white</td></tr></table> <p>For more information, see the control character COL in the VTCSET Assembler macro.</p>	00	standard color	04	green	01	blue	05	cyan	02	red	06	yellow	03	magenta	07	white
00	standard color	04	green														
01	blue	05	cyan														
02	red	06	yellow														
03	magenta	07	white														

EXTEND-LOG-CHAR Logical code extension character. This logical control character can be used only in conjunction with the DIMENSION, TRANSPARENT-OUTPUT, REPEAT-SYMBOL, DISPLAY-ATTR or FIELD-CHARACTERISTICS control characters. In all other circumstances, EXTEND-LOG-CHAR is converted to SUBSTITUTE-CHAR.

The logical code extension character EXTENDED-LOG-CHAR must precede the control characters listed below.

DIMENSION-24-80 The standard (default) screen format 24x80 is set.
DIMENSION-32-80 Screen format 32x80 is set.
DIMENSION-43-80 Screen format 43x80 is set.
DIMENSION-27-132 Screen format 27x132 is set.

Note that selection of a screen format (DIMENSION) on a 9763 Data Display Terminal is effective only after NEW-PAGE. For more information, see the control character DIM in the VTCSET Assembler macro.

TRANSPARENT-OUTPUT

The control character string is passed without change (transparently) to terminal and printer.

For more information, see the control character TRA in the VTCSET Assembler macro.

REPEAT-SYMBOL The subsequent displayable or NULL character is repeated. A two-digit decimal value after the control character specifies the number of repeats.

For more information, see the control character RPT in the VTCSET Assembler macro.

DISPLAY-ATTR Output attributes

For more information, see the control character DIS in the VTCSET Assembler macro.

FIELD-CHARACTERISTICS

Field attributes

For more information, see the control character FLD in the VTCSET Assembler macro.

The control characters listed below must be preceded by the logical code extension character EXTENDED-LOG-CHAR and the output attribute DISPLAY-ATTR. The subsequent text characters are then emphasized by the device-specific highlight mode on the monitor. On a 9763 Color Data Display Terminal, the attributes 'flashing' 'underscored', 'high intensity' and combinations thereof are mapped to colors, depending on SIDATA. Emphasized layout is reset by the logical record control characters or by the display control character NORMAL-LAYOUT. Any characteristics not supported by the data display terminal are ignored.

FLASHING	The output attribute 'flashing' is assigned to the subsequent field.
UNDERScoreD	The output attribute 'underscored/italics' is assigned to the subsequent field.
BLANKED	The output attribute 'blanked' is assigned to the subsequent field.
REDUCED-INTENSITY	The output attribute 'reduced intensity' is assigned to the subsequent field.
INVERSE-MODE	The output attribute 'inverse' is assigned to the subsequent field. Inverse is ignored for terminals older than the 9763 DDT.
RESET	The attributes FLASHING, UNDERScoreD, BLANKED, REDUCED-INTENSITY and INVERSE-MODE are reset. Note that RESET does not have the same effect as NORMAL-LAYOUT, which automatically activates the 'low intensity' attribute.

You can specify one attribute or a sequence of attributes.

e.g. EXTENDED-LOG-CHAR DISPLAY-ATTR UNDERScoreD+FLASHING text

The subsequent text appears underscored and flashing.

The control characters listed below must be preceded by the logical code extension character EXTENDED-LOG-CHAR and the field attribute FIELD-CHARACTERISTICS. FIELD-CHARACTERISTICS creates a start of field. This field is non-printable by default. If emphasized layout was selected beforehand, it is reset to low intensity for a protected field and high intensity for all other fields. When field characteristics are combined the output attribute 'low intensity' is retained. Other output attributes must be set explicitly by means of the control characters DISPLAY-ATTR or EMPH-LAYOUT1 - EMPH-LAYOUT4. Bear in mind that the field attributes must always precede the output attributes.

PROTECTED-NOT-SENDABLE

The field characteristics 'protected' and 'not-sendable' are assigned to the subsequent field.

PROTECTED-SENDABLE

The field characteristics 'protected' and 'sendable' are assigned to the subsequent field.

NUMERIC

The field characteristic 'numeric' is assigned to the subsequent field.

PREMODIFIED

The field characteristic 'premodified' is assigned to the subsequent field.

MARKABLE

The field characteristic 'markable' is assigned to the subsequent field.

PRINTABLE

The field characteristic 'printable' is assigned to the subsequent field.

INPUT

The subsequent field is an unprotected input field having the field characteristics 'not numeric', 'not markable' and 'not printable'.

AUTOSKIP

The field characteristic 'protected field with automatic tabulator' is assigned to the subsequent field (3270 terminals only).

You can specify one field characteristics or a sequence of characteristics.

e.g. EXTENDED-LOG-CHAR FIELD-CHARACTERISTICS NUMERIC+MARKABLE text

The subsequent text is numeric and markable.

The control characters listed below are only for printers and are not used by TIAM.

CONT-COL-N The cursor is positioned to column n of the current line; n is specified as a three-digit printable decimal number following the control character. In extended line mode the control character CONT-COL-N is not processed unless it is specified immediately after a valid CONT-LINE-N.

For more information, see the control character HPA in the VTCSET Assembler macro.

SHEET-FEED-N Sheet feed or ejection at the printer is controlled:

n=0 sheet ejection at message end and with form feed attachment switchover to tractor
n=1, 2, 3 sheet ejection and sheet feed from tray n
n=9 sheet feed from form feed attachment (9013 Printer only)

(is replaced by NEW-PAGE if the latter has already occurred in the message); n is specified by a printable digit after the control character.

For more information, see the control character ASF in the VTCSET Assembler macro.

CONT-ACT-POS Start of the printout at the current carriage position (not on the next line).
Logical control characters are not reset at the start or end of the message. To ensure that the status is defined initially, positioning to the start of line is required when CONT-ACT-POS is used several times in an output sequence (NEW-LINE, NEW-PAGE or CONT-LINE-N). CONT-ACT-POS is only admissible as the first character in a message. Data may be overwritten when CONT-ACT-POS is used for the first time unless the carriage is explicitly positioned at the start of a new line.

For more information, see the control character CAP in the VTCSET Assembler macro.

PART-LINE-UP The printer is backspaced half a line. A further PART-LINE-UP can only be issued after PART-LINE-DOWN or in a new line. PART-LINE-UP is reset by PART-LINE-DOWN or at the logical end of line.

For more information, see the control character PLU in the VTCSET Assembler macro.

- PART-LINE-DOWN** The printer is spaced forward half a line. A further PART-LINE-DOWN can only be issued after PART-LINE-UP or in a new line. PART-LINE-DOWN is reset by PART-LINE-UP or at the logical end of line.
- For more information, see the control character PLD in the VTCSET Assembler macro.
- VERT-MOVE-IND** The printable character *d* after the control character defines the line spacing
(only after NEW-PAGE or SHEET-FEED-N):
- | | | |
|-----|---------------------|---------------------------|
| d=1 | normal line spacing | (1/6 inch; default value) |
| d=2 | narrow line spacing | (1/8 inch) |
| d=3 | half-line spacing | (1/12 inch) |
- For more information, see the control character VMI in the VTCSET Assembler macro.
- HORIZ-MOVE-IND** The printable character *d* after the control character defines the character spacing (only after NEW-LINE, NEW-PAGE, SHEET-FEED-N, CONT-LINE-N or at the start of message):
- | | | |
|-----|------------------|----------------------------|
| d=1 | normal spacing | (1/10 inch; default value) |
| d=2 | condensed type A | (1/12 to 1/15 inch) |
| d=3 | condensed type B | (1/15 to 1/17 inch) |
- For more information, see the control character HMI in the VTCSET Assembler macro.
- LEFT-MARGIN** A three-digit printable decimal number after the control character defines the column from which all subsequent lines are to start. A useful line length of 10 characters must remain. LEFT-MARGIN is only allowed at the start of a message or after NEW-LINE, NEW-PAGE, SHEET-FEED-N or CONT-LINE-N and is active until the end of the message if no new definition is made.
- For more information, see the control character LM in the VTCSET Assembler macro.
- START-PROP-TYPE** The subsequent characters are printed with proportional spacing. Column counting is stopped. START-PROP-TYPE is active until END-PROP-TYPE or until the end of the message.
- For more information, see the control character PTS in the VTCSET Assembler macro.

END-PROP-TYPE	Printing with proportional spacing is stopped. For more information, see the control character PTX in the VTCSET Assembler macro.
MAX-LINE-LEN	The maximum number of characters per line is redefined by a threedigit decimal number after the control character. It applies for the entire duration of the connection or until it is reset. MAX-LINE-LEN is allowed only after NEW-PAGE or SHEET-FEED-N. For more information, see the control character MLL in the VTCSET Assembler macro.
MAX-LINE-NUM	The maximum number of lines per page is redefined by a threedigit decimal number after the control character. It applies for the entire duration of the connection or until it is reset. MAX-LINE-NUM is allowed only after NEW-PAGE or SHEET-FEED-N. For more information, see the control character MLN in the VTCSET Assembler macro.
START-NEAR-LETT	Start letter quality. For more information, see the control character NLQ in the VTCSET Assembler macro.
END-NEAR-LETT	Stop letter quality. For more information, see the control character NLX in the VTCSET Assembler macro.

Example

```
.
.
.
WORKING-STORAGE SECTION.
*
  COPY TIAMINFO.
*
  COPY TIAMCTRC.
*
01  OUTPUT-AREA.
    41  MSGLENGTH PIC 9(5)  COMP SYNC  VALUE 71.
    41  FILLER    PIC X.
    41  TXT1      PIC X(22) VALUE
          "THIS IS AN EXAMPLE.".
    41  STZ1      PIC X.
    41  TXT2      PIC X(20) VALUE "THIS TEXT IS ".
    41  STZ2      PIC X.
    41  TXT3      PIC X(9)  VALUE "OUTPUT IN".
    41  STZ3      PIC X.
    41  TXT4      PIC X(12) VALUE " LINE MODE.".
*
*
PROCEDURE DIVISION.
  MOVE NEW-LINE      TO STZ1  IN OUTPUT-AREA.
  MOVE EMPH-LAYOUT1 TO STZ2  IN OUTPUT-AREA.
  MOVE NORMAL-LAYOUT TO STZ3  IN OUTPUT-AREA.
.
.
.
```

Terminal	Effect of logical display control characters					
	NORMAL LAYOUT	EMPH-LAYOUT1	EMPH-LAYOUT2	EMPH-LAYOUT3	EMPH-LAYOUT4	DARK-LAYOUT
8110 TTY	-	-	-	-	-	-
8150	-	-	-	-	-	
8151	still	flashing	flashing	flashing	flashing	-
8152	roman	italics	italics	italics	italics	-
8160 } 8162 } OM 1	low int. roman still	low int. roman flashing	low int. italics still	high int. roman still	high int. italics still	blanked
8160 OM 2	high int. roman still	high int. roman flashing	high int. italics still	low int. roman still	low int. italics still	blanked
9748 } 9749 } OM 1 9750 } 9751 }	low int. roman still	low int. roman flashing	low int. under- scored still	high int. roman still	high int. under- scored still	blanked
9748 } 9749 } OM 2 9750 } 9751 }	high int. roman still	high int. roman flashing	high int. under- scored still	low int. roman still	low int. under- scored still	blanked
9752	yellow roman	yellow flashing	white roman	green roman	red roman	blanked
9755 } 9756 } OM 1 9758 }	low int. roman	low int. roman flashing	low int. under- scored	high int. roman	high int. under- scored	blanked
9755 } 9756 } OM 2 9758 }	high int. roman	high int. roman flashing	high int. under- scored	low int. roman	low int. under- scored	blanked

TTY: PT80, T100, T1000;

OM: operating mode

-: control character is ignored

Terminal	Effect of logical display control characters					
	NORMAL LAYOUT	EMPH-LAYOUT1	EMPH-LAYOUT2	EMPH-LAYOUT3	EMPH-LAYOUT4	DARK-LAYOUT
9763 / 9759 OM 1 monochrome screen	low int. roman	low int. roman flashing	low int. under- scored	high int. roman	high int. under- scored	blanked
9763 / 9759 OM 2 monochrome screen	high int. roman	high int. roman flashing	high int. under- scored	low int. roman	low int. under- scored	blanked
9763 OM 1 color screen	yellow	cyan	white	green	red	blanked
9763 OM 2 color screen	green	red	red	yellow	white	blanked
3270 OM 1	low int. roman	high int. roman	high int. roman	high int. roman	high int. roman	blanked
3270 OM 2	high int. roman	low int. roman	low int. roman	low int. roman	low int. roman	blanked
3279 OM 1	green	red	red	red	red	blue
3279 OM 2	green	white	white	white	white	green
8121 8122	normal	italics	italics	italics	italics	-

TTY: PT80, T100, T1000;

OM: operating mode

-: control character is ignored

Terminal	Effect of logical display control characters					
	NORMAL-LAYOUT	EMPH-LAYOUT1	EMPH-LAYOUT2	EMPH-LAYOUT3	EMPH-LAYOUT4	DARK-LAYOUT
9001	normal	under-scored	under-scored	under-scored	under scored	-
9002	normal	italics	under-scored	italics	ital. and under-scored	-
9003	normal	italics	red	italics	red and italics	-
9004	normal	shadow	under-scored	bold	bold and under-scored	-
9013	normal	under-scored	under-scored	bold	bold and under-scored	-
9012	normal	under-scored	under-scored	bold	bold and under-scored	-
9011- 18/19	normal	italics 1)	under-scored	bold	bold and under-scored	-
9001-31/8931	normal	italics 1)	under-scored	bold	bold and under-scored	-
9021	normal	italics	under-scored	bold	bold and under-scored	-
9022	normal	shadow	under-scored	bold	bold and under-scored	-

TTY PT80, T100, T1000; OM operating mode - control character is ignored

1) Only when connected to a 9763 DDT or BAM controller, otherwise underscored or no effect.

Terminal	Effect of logical display control characters				
	SECOND-CHAR-SET	NORMAL-CHAR-SET	START-PROT-AREA	END-PROT-AREA	START-NUM-DATA
8110 TTY	-	-	-	-	-
8150	-	-	protected low int.	unprotected high int.	-
8151	-	-	protected low int.	unprotected high int.	-
8152	APL character set	first character set	protected low int.	unprotected high int.	-
8160	-	-	protected low int.	unprotected high int.	unprotected high int. *
8162	second character set	first character set	protected low int.	unprotected high int.	unprotected high int. *
9748 9749 9750 9751	-	-	protected low int.	unprotected high int.	unprotected high int. *
9752	-	-	protected yellow	unprotected green	unprotected green *
9755 9756 9758	-	-	protected low int.	unprotected high int.	unprotected high int. *
9763 monochrome screen	-	-	protected low int.	unprotected high int.	unprotected high int. *
9763 color screen	-	-	protected yellow	unprotected green	unprotected green *
3270	-	-	protected low int.	unprotected high int.	unprotected high int. *
3279 OM 1	-	-	white	red	red
3279 OM 2	-	-	white	white	white

* numeric values only

Terminal	Effect of logical display control characters				
	SECOND-CHAR-SET	NORMAL-CHAR-SET	START-PROT-AREA	END-PROT-AREA	START-NUM-DATA
812x	-		-	-	-
9001	-	-	-	-	-
9002 3)	second character set	first character set	-	-	-
9003	second character set	first character set	-	-	-
9004	second character set	first character set	-	-	-
9013 3)	character set extension 1)	basic character set 2)	-	-	-

- 1) Right half of the ISO 8-bit code table; useful e.g. for teletex character set
- 2) Left half of ISO 8-bit code table.
- 3) Dependent on printer type (see description of printer in user guide)

TTY: PT80, T100, T1000; OM: operating mode; -: control character is ignored

Terminal	Effect of logical display control characters				
	SECOND-CHAR-SET	NORMAL-CHAR-SET	START-PROT-AREA	END-PROT-AREA	START-NUM-DATA
9012	character set extension 1)	basic character set 2)	-	-	-
9011-18/19	second character set	first character set	-	-	-
9001-31/8931	-	-	-	-	-
9021	second character set	first character set	-	-	-
9022	character set extension 1)	basic character set 2)	-	-	-

1) Right half of the ISO 8-bit code table; useful e.g. for teletex character set

2) Left half of ISO 8-bit code table.

TTY: PT80, T100, T1000;

OM: operating mode;

-: control character is ignored

Applicability of the logical control characters to individual printers and data display terminals

\ Ter- Control\minal charact.\	9004	9001	9003	9002	9012 9013	9011 -18/ 19	9001 -31/ 8931	9022	9021	812x	816x	9748 9749	975x	976x	3270	3279
PART-LINE-DOWN PART-LINE-UP	X	X	-	-	X	X	X	X	X	-	-	-	-	-	-	-
LEFT-MARGIN	X	-	-	-	X	-	-	X	-	-	-	-	-	-	-	-
START-PROP-TYPE END-PROP-TYPE	X	-	-	-	X	X	X	X	X	-	-	-	-	-	-	-
CONT-LINE-N	X	-	-	X	X	-	-	X	X	-	X	X	X	X	X	X
CONT-COL-N	X	-	-	X	X	X	X	X	X	-	*	*	*	*	*	*
SHEET-FEED-N	X	-	-	-	X	X	-	X	X	-	-	-	-	-	-	-
MAX-LINE-LEN	X	X	-	X	X	X	X	X	X	X	-	-	-	-	-	-
MAX-LINE-NUM	X	X	-	X	X	X	X	X	X	X	-	-	-	-	-	-
BACKSPACE	X	-	-	X	X	X	X	X	X	X	-	-	-	-	-	-
CONT-ACT-POS	X	X	-	X	X	X	X	X	X	X	-	-	-	-	-	-
PHYS-ESC PHYS-DC4	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
PHYS-HT PHYS-VT	X	X	X	X	X	X	X	X	X	-	X	X	X	X	X	X

- X → function is initiated
- → logical control character is suppressed
- * → if preceded by a legal CONT-LINE-N, the logical control character is suppressed

Note

- The 9022 Printer ignores proportional type unless the current font supports proportional spacing.
- Proportional type is impractical with the 9013 Printer unless a suitable character set is selected.
- When used on the 9013 and 9002 Printers, the control character CONT-LINE-N causes the execution of a number of line feeds defined by n.
- When used on the 9002 Printer, the control character CONT-COL-N either adds the number of blanks defined by n or defines the column as of which the subsequent characters are to be output. You can select either of these two options. The insertion of n blanks is the default.

Applicability of logical control characters for output

\ Terminal Control\ charact.\	9736	975x	9748 9749	816x	3270	9001 9002 9003	9004 9013	9012	9011 -18/ 19	9001 -31/ 8931	9022	9021
START-CHARSET-N	X	-	-	-	-	-	-	-	-	-	-	-
START-LOCAL-ATTR END-LOCAL-ATTR	X	1)	1)	1)	1)	1)	1)	1)	1)	1)	1)	1)
DIMENSION	X	-	-	-	-	-	-	-	-	-	-	-
TRANSPARENT-OUTPUT	X	X	X	X	X	X	X	X	X	X	X	X
START-NEAR-LETT END-NEAR-LETT	-	-	-	-	-	-	-	-	X	X	-	-
DISPLAY-ATTR	X	X	X	X	X	-	-	-	-	-	-	-
FIELD- CHARACTERISTICS	X	X	X	X	X	-	-	-	-	-	-	-
SELECT-COLOR	X	-	-	-	-	-	-	-	-	-	-	-
REPEAT-SYMBOL	X	X	X	X	X	X	X	X	X	X	X	X

- X → function is initiated
- → logical control character is suppressed
- 1) substitution takes place (see LOC and LOX control characters).

Note

On the 9763 Data Display Terminal, only the screen formats supplied with TSTAT/YINQUIRE are supported for the logical control character DIMENSION.

Effect of output attributes

Control \ Terminal charact. \	8110	815x	8160	9750	9755	9758 9756	9763 9759	3270
FLASHING	-	-	flashing	flashing	flashing	flashing	flashing	-
UNDERScoreD	-	-	italics	under- scored inv. 1)	under- scored Inv. 2)	under- scored Inv. 2)	under- scored Inv. 2)	-
BLANKED	-	-	blanked	blanked	blanked	blanked	blanked	blanked not printable
REDUCED-INTENSITY	-	-	low int.	low int.	low int.	low int.	low int.	low int.
INVERSE-MODE	-	-	-	-	-	inverse	inverse	-

-: control character is ignored

1) set by jumper or ROM

2) selectable by SIDATA

Effect of field attributes

\ Term- Control\ charact.\	8110	815x	8160	9750	9755	9758 9756	9763 9759	3270
PROTECTED-NOT SENDABLE	-	-	protected not sendable	protected not sendable	protected not sendable	protected not sendable	protected not sendable	protected not sendable
PROTECTED- SENDABLE	-	-	protected sendable	protected sendable	protected sendable	protected sendable	protected sendable	-
NUMERIC	-	-	numeric	numeric	numeric	numeric	numeric	numeric
PREMODIFIED	-	-	pre- modified	pre- modified	pre- modified	pre- modified	pre- modified	pre- modified
MARKABLE	-	-	mark- able	mark- able	mark- able	mark- able	mark- able	select- able ¹⁾
PRINTABLE	-	-	printable	printable	printable	printable	printable	printable ₂₎
AUTOSKIP	-	-	-	-	-	-	-	protected automat. tabulator

-: control character is ignored

- 1) the first character in the field is a destination character
- 2) on a 3270 terminal, a non-printable field is automatically blanked

Character spacing and line spacing for the individual printers (in inches)

Printer	HORIZ-MOVE-IND			VERT-MOVE-IND		
	d=1	d=2	d=3	d=1	d=2	d=3
9004	1/10	1/12	1/15	1/6	1/8	1/12
9001	1/10	1/12	1/17	1/6	1/8	1/12
9003	1/10	1/12	1/15	1/6	1/8	1/12
9002	1/10	1/10	1/10	1/6	1/6	1/6
9013	1/10	1/12	1/15 1st ch.set only	1/6	1/8	1/12
9012	1/10	1/12	1/12	1/6	1/8	1/12
9011 18/19	1/10	1/12	1/15	1/6	1/8	1/12
9001 31/8931	1/10	1/12	1/15	1/6	1/8	1/12
9022	1/10	1/12	1/15	1/6	1/8	1/12
9021	1/10	1/12	1/15	1/6	1/8	1/12
812x	1/10	1/10	1/10	1/6	1/6	1/6

Note

- On the 9011 Printer, bold is not executed after HORIZ-MOVE-IND d=3.
- On the 9021 Printer, HORIZ-MOVE-IND d=3 is executed only if a suitable character set is selected; otherwise the next narrowest character set is selected.

Effect of reset at connect time on logically supported functions

\ Term- Control\ inal charact.\	9001	9004	9013	9012	9011 -18/ 19	9001 -31/ 8931	9022	9021	VTSU STD
HORIZ-MOVE-IND	1/10	S	S	M	M	S	S	S	1/10
VERT-MOVE-IND	S	S	S	M	M	S	S	S	1/6
START-PROP-TYPE	-	R	R	M	M	R	R	R	R
SECOND-CHAR-SET	-	R	R	R	R	-	R	R	1)R
START-CHARSET-N	CS1	-	S	M	M	ZV1	R	R	ZV1
SHEET-FEED-N	-	SHEET- FEED-1	SHEET- FEED-n n(S)	SHEET- FEED-n n(M)	Tract SHEET- FEED-n n(M)	-	SHEET- FEED-1	SHEET- FEED-1	-
EMPH-LAYOUT1-4	R	R	R	R	EM1:M EM2:R EM3:M	R	R	R	R
START-NEAR-LETT	-	-	-	M	M	S	-	-	R
CR	S	-	S	M	M	-	-	-	CR

S : Reset to switch setting
M : Reset to menu setting
R : Reset
- : No logical support for function
CS: Character set
n(S): n defined by switch setting
n(M): n defined by menu selection

1) Left half of the ISO 8-bit code table for printers with 8-bit character sets; otherwise basic character set.

Note

- VTSU default
It is assumed for the logical support of these functions that the VTSU default values are set by menu or switch settings. Any differences from the default settings are the responsibility of the user.
- Automatic sheet feed
For the 9011-18/19 Printer, the menu setting for automatic sheet feed (SHEET-FEED-N) simply means selection of tractor or sheet feeder.

4.3 VTSU control block: VTSUCBC

The **VTSUCBC** data structure allows you to define VTSU parameters for input and output independently of the access method.

For **DCAM applications** you can include the VTSUCB in the YSEND and YRECEIVE macros, but note that in the command structure the VTSUCB field must be set to 'YES'. The VTSUCB is not evaluated unless EDIT='SYS' is used for the connection. DCAM copies the VTSUCB into its own data structure. If the parameters in the VTSUCB remain unchanged, it is not necessary to specify VTSUCB in every macro. If the VTSUCB is to be included, DCAM expects the VTSUCB as the fourth parameter; e.g.

```
CALL YSEND USING APP-NAME,CONN-NAME,BEF-NAME,VTSUCB,BEREICH1
```

Note

Note that with DCAM applications it is not possible to change between input and output (YSEND and YRECEIVE) and simultaneously change mode. The mode of the YRECEIVE macro is always defined by the preceding YSEND, which sets the terminal to the requested mode.

For **TIAM applications** you can include the VTSUCB in the RDATA, WROUT and WRTRD macros, for example

```
CALL "WROUT" USING TIAM-CONTROL-INFO user area 1 VTSUCB.
```

Parameters set to values other than the defaults must be set in this VTSUCB with MOVE.

The parameters NO-POSITION, READ-MODE, CURSOR-POSITION, SCREEN-UPDATE and AUTOMATIC-TABULATION are new.

The data structure VTSU-CONTROL-BLOCK is copied to the application program by means of the call:

COPY VTSUCBC

This copy member is located in the library \$TSOS.SYSLIB.VTSU-B.110.

COPY VTSUCBC

```

*          VTSUCBC      050          921209          53531054          *
*****
*  VTSUCBC   :  V050
*****
*
*  1 VTSUCB.
*
*    31 STANDARD-HEADER.
*      32 FUNCTION-UNIT          PIC 9(4)  COMP VALUE 49.
*      32 FUNCTION-VERS          PIC 9(4)  COMP VALUE 260.
*      32 RETURN-CODE.
*        33 SUBCODES              PIC S9(4)  COMP VALUE -1.
*        33 SUBCODES-RED          REDEFINES SUBCODES.
*          34 SUBCODE-2            PIC X.
*          34 SUBCODE-1            PIC X.
*        33 MAINCODE              PIC S9(4)  COMP VALUE -1.
*
*    31 BLOCKLEN                  PIC 9(4)  COMP VALUE 48.
*                                  LENGTH OF VTSUCB
*
*    31 INPUT-MODUS                PIC X.
*                                  INPUT MODE FOR MIXED MODE
*
*    31 OUTPUT-MODUS              PIC X.
*                                  OUTPUT MODE FOR MIXED MODE
*
*    31 FILLER                      PIC X(4).
*                                  RESERVED FOR RETURN-INFO
*
*
*    31 MODUS                      PIC X
*                                  VALUE "L".
*                                  MODE OF MESSAGE
*      88 LINE-MODE                 VALUE "L".
*      88 EXTENDED-LINE             VALUE "E".
*      88 INFO                       VALUE "I".
*                                  INFO LINE MESSAGE
*      88 PHYS                       VALUE "P".
*      88 TRANS                      VALUE "T".
*      88 FORM                       VALUE "F".
*      88 CHIP                       VALUE "C".
*                                  CHIP CARD MODE
*      88 MIXED                     VALUE "M".
*                                  MIXED MODE
*
*
*    31 HARDCOPY                   PIC X
*                                  VALUE "N".
*                                  HARDCOPY FUNCTION
*      88 NO-HC                     VALUE "N".
*                                  NO HARDCOPY
*      88 YES-HC                    VALUE "Y".
*                                  LOCAL/CENTRAL HARDCOPY
*
*
*    31 BELL                       PIC X
*                                  VALUE "N".

```

```

*
*           88 NO-BELL                BELL FUNCTION
*                                           VALUE "N".
*           88 YES-BELL               NO BELL
*                                           VALUE "Y".
*                                           BELL AFTER OUTPUT
*
*           31 NO-LOG-CHARS          PIC X           VALUE "N".
*                                           NO LOG CHARS TO INTERPRET FCT
*           88 NO-NOLOG               VALUE "N".
*                                           LOGICAL CHARACTERS TO INTERPRET
*           88 YES-NOLOG              VALUE "Y".
*                                           NO LOGICAL CHAR TO INTERPRET
*
*           31 RETURN-BYTES          PIC X(2).
*                                           RETURN INFO BYTES
*
*           31 RET-INFO               PIC X           VALUE "N".
*                                           RETURN INFORMATION FUNCTION
*           88 NO-RIN                 VALUE "N".
*                                           NO RETURN INFORMATION
*           88 YES-RIN                VALUE "Y".
*                                           RETURN INFORMATION REQUIRED
*
*           31 LOCAL-CHARS           PIC X           VALUE "N".
*                                           INPUT OF LOCAL CHARACTERS
*           88 NO-LOC                  VALUE "N".
*                                           NO LOCAL CHARACTERS REQUIRED
*           88 YES-LOC                 VALUE "Y".
*                                           LOCAL CHARACTERS REQUIRED
*
*           31 OUTPUT-HEADER         PIC X           VALUE "N".
*                                           88 NO-OHDR                VALUE "N".
*                                           NO OUTPUT HEADER IN USER
*           88 YES-OHDR               VALUE "Y".
*                                           OUTPUT HEADER IN USER MSG
*
*           31 CODE-TRANSLATION      PIC X           VALUE SPACE.
*                                           CODE TRANSLATION FUNCTION
*           88 NO-TRANSLATION         VALUE "N".
*                                           NO CODE TRANSL. TO/FROM CCS
*           88 YES-TRANSLATION        VALUE "Y".
*                                           CODE TRANSLATION REQUIRED
*
*           31 INPUT-HEADER           PIC X           VALUE "Y".
*                                           INPUT HEADER FUNCTION
*           88 NO-IHDR                 VALUE "N".
*                                           NO INPUT HEADER REQUIRED
*           88 YES-IHDR                VALUE "Y".
*                                           INPUT HEADER REQUIRED
*
*           31 LOWER-CHARS           PIC X           VALUE "N".
*                                           LOWER CHARACTERS FUNCTION
*           88 NO-LOW                  VALUE "N".
*                                           TRANSLATE LOWER CHARACTER
*           88 YES-LOW                 VALUE "Y".
*                                           RETAIN LOWER CHARACTERS
*
*

```

```

*      31 SPECIAL-INPUT  PIC X          VALUE "N".
*                                     SPECIAL INPUT FUNCTION
*      88 NO-SPEC-INPUT          VALUE "N".
*                                     NO SPECIAL INPUT
*      88 ID-CARD-READER        VALUE "I".
*                                     INPUT FROM ID-CARD READER
*      88 CONFIDENTIAL          VALUE "C".
*                                     CONFIDENTIAL INPUT
*
*      31 FUNCTION-CODE  PIC X          VALUE "N".
*                                     FUNCTION CODE
*      88 NO-FC          VALUE "N".
*                                     NO FUNCTION CODE REQUIRED
*      88 YES-FC         VALUE "Y".
*                                     FUNCTION CODE REQUIRED
*
*      31 HOM-OUTPUT     PIC X          VALUE "N".
*                                     HOMOGENEOUS OUTPUT
*      88 NO-HOM        VALUE "N".
*                                     NO HOM. OUTPUT REQUIRED
*      88 YES-HOM       VALUE "Y".
*                                     HOM. OUTPUT REQUIRED
*
*      31 NO-POSITION   PIC X          VALUE "N".
*                                     OUTPUT ON CURRENT LINE
*      88 NO-NOPOS     VALUE "N".
*                                     OUTPUT ON NEXT LINE
*      88 YES-NOPOS    VALUE "Y".
*                                     OUTPUT ON CURRENT LINE
*
*      31 CCSNAME              PIC X(8) VALUE SPACES.
*
*      31 CURSOR-POSITION PIC X          VALUE "N".
*
*      88 NO-CURSOR-POS      VALUE "N".
*                                     CURSOR POSITION NOT REQUESTED
*      88 YES-CURSOR-POS    VALUE "Y".
*                                     CURSOR POSITION REQUESTED
*
*      31 CURSOR-POS-LINE  PIC X.
*                                     CURSOR POSITION IN LINE
*      31 CURSOR-POS-COLUMN PIC X.
*                                     CURSOR POSITION IN COLUMN
*
*      31 READ-MODE        PIC X          VALUE "U".
*
*      88 READ-MODIFIED    VALUE "M".
*
*      88 READ-UNPROTECTED VALUE "U".
*
*      31 SCREEN-UPDATE  PIC X          VALUE "N".
*                                     SCREEN UPDATE BY EXT.LINE MSG
*      88 NO-SCREEN-UPDATE VALUE "N".
*
*      88 YES-SCREEN-UPDATE VALUE "Y".
*
*      31 WARBYTE-INFO   PIC X          VALUE "N".
*                                     WAR BYTE VALUE REQUESTED
*      88 NO-WARBYTE-INFO VALUE "N".
*
*

```

```

      88  YES-WARBYTE-INFO          VALUE "Y" .
*
      31  WAR-BYTE-VALUE            PIC X .
*
      31  AUTOMATIC-TABULATION PIC X  VALUE "S" .
*                                     AUTOMATIC TABULATION REQUEST
      88  STD-AUTOTAB              VALUE "S" .
*
      88  NO-AUTOTAB               VALUE "N" .
*
      88  YES-AUTOTAB              VALUE "Y" .
*
*
```

Description of fields

STANDARD-HEADER

VTSU enters error messages in the VTSUCB standard header. The standard header also contains error messages which are not associated directly with the VTSUCB. A special return code supplied by the access methods indicates whether the VTSUCB contains error messages.

The error information contained in the standard header of the VTSUCB is described in the ASSEMBLER section, beginning on page 102.

MODE

- "L" The current terminal is to be handled as a virtual line or page terminal. The message can be structured with the aid of logical control characters.
- "E" This specification supports the use of protected and unprotected fields with the aid of the logical control characters END-PROT-AREA, START-NUM-DATA, START-PROT-AREA and DARK-LAYOUT (extended line mode). Output of text is protected and displayed at low intensity as the default option. The message can be structured by logical control characters. The keys RU, EFZ, AFZ and LSP are locked. The control character NL is not permitted in input. MODE=E can only be specified for 9749, 975x, 9763, 816x and 3270 terminals. Printers process EXTEND internally as line mode.
- "I" Messages can be displayed in a special information line without the risk of destroying important data. This operand is intended primarily for application programs sending "asynchronous" messages to terminals when the information currently displayed is not known.

On terminals with a hardware system line (e.g. 9749, 9750 and 9752 DDTs), the message is always protected. In all other cases, it is displayed as a normal line-mode message. If the message is longer than one screen line, it is split up and output line by line.

DCAM applications do not automatically reset the system line.

With TIAM applications the system line is automatically reset after the first input that follows an output in the system line.

- "P" The messages are to be output at the terminal "physically", i.e. without being edited by the system, or read in physically from the terminal. This allows special device functions to be executed for which the LINE or FORM mode is insufficient.
- "T" The output data is to be transmitted "transparently". It may consist of any binary characters (5, 7 or 8 bits per character depending on the device code), and is not converted on the transmission path.
- "F" The current terminal is to work in format mode. The application program uses the software component FHS or format handling, which also edits the message in a form suitable for output on a particular terminal.
- "C" The output message is passed to the chip card terminal by the appropriate device protocol. The message must be generated in expanded mode (see the manuals "Data Display Terminals, Functional Description" and "Data Display Terminals, Code Tables"). If the chip card terminal is not addressable, output is rejected with a return code.

On input, the message is checked as to whether it has been transmitted from the chip card terminal. If so, the device protocol is removed, and the first byte preceding the input message is set to the function key code. Input messages not originating from a chip card terminal are converted to the short code K14.

- "M" The application program mixes different modes for input (INPUT-MODE) and output (OUTPUT-MODE). Only MODE="L", "E", "F" and "P" may be combined in this way. MODE="C", "I" or "T" must not be combined.
The default value for LOWER-CHARS for input and output depends on the specified input mode. If MODE="P", then LOWER-CHARS is "YES", if MODE="L", "E" or "F", LOWER-CHARS is "NO". All other parameter values are defined in accordance with the specified input/output modes.

HARDCOPY is evaluated only if MODE="L" or "M"

- "N" The output message is not output simultaneously at the terminal and at a hardcopy unit (printer) connected to the terminal.
- "Y" The message output to a data display terminal is simultaneously output to a hardcopy unit (printer) connected to the terminal.

BELL is evaluated only if MODE="L","E","I" or "M"

- "N" No acoustic alarm sounds when an output takes place.
- "Y" An acoustic alarm sounds when an output takes place (only on 9749, 975x, 9763, 816x and 3270 display terminals with a special hardware option).

NO-LOG-CHARS is evaluated only if MODE="L","I" or "M"

- "N" All logical control characters are interpreted and special physical control characters are accepted. Other characters less than X'40' are replaced by SUB (smudge character). Printable characters are accepted.
- "Y" Logical control characters are not interpreted. All characters less than X'40' are replaced by SUB. Only printable characters are accepted.

RETURN-BYTES

This item must contain two printable characters if return information is required from the printer terminal (RET-INFO="Y").

RET-INFO is evaluated only if MODUS="L" or "M"

- "N" No printer return information is required.
- "Y" Return information is required by the printer terminal. The two characters specified in the item RETURN-BYTES are returned with the return information.

LOCAL-CHARS

- "N" Local attributes are removed from the input message.
- "Y" If the input message contains local attributes, they are transmitted as logical control characters. If the local attributes were set by means of START-LOCAL-ATTR EMPH-LAYOUT1-4 or EXTENDED-LOG-CHAR DISPLAY-ATTR, the logical control character passed to the user is always START-LOCAL-ATTR EMPH-LAYOUT1-4. This can only be specified for data display terminals which support local attributes such as character set (e.g. the 9763 Data Display Terminal).

OUTPUT-HEADER is evaluated only if MODE="P" or "M"

- "N" No user-specific header is included. The message header created by the system is prefixed to the output text.
- "Y" The message includes a user-specific header which the system prefixes to the output text.

CODE-TRANSLATION is evaluated only if MODE="P" or "M"

Specifies for physical output whether the message is to be converted from or to the specified code. This parameter is of assistance only for output on printers that process ESCAPE sequences not coded in conformance with the EBCDIC core. These special ESCAPE sequences are ignored by VTSU.

- "Y" VTSU converts the message from or to the specified code. VTSU recognizes and skips the standard ESCAPE sequences.
- "N" The control characters SO/SI are evaluated. VTSU does not implement any further conversion.

INPUT-HEADER is evaluated only if MODE="P" or "M"

- "N" The message header is not passed to the application program.
- "Y" The message header is passed to the application program.

LOWER-CHARS is evaluated only if MODE="L","E","I","P","F" or "M"

If MODE="M", the default value for LOWER-CHARS for input and output depends on the specified input mode. If MODE="P", LOWER-CHARS "YES"; if MODE="L","E","I" or "F", LOWER-CHARS "NO".

- "N" All lowercase letters are passed to the application program as upper case.
- "Y" Lowercase letters are passed to the application program (default option for MODE="P").

SPECIAL-INPUT is evaluated only if MODE="L" or "M"

This parameter must have been specified in the output message requesting special input.

"N" Normal input from the terminal.

"I" The data is input from the ID card reader. The input data may consist of the ID card information or the short code K14. This specification is only possible with 9749, 975x, 9763, 816x and 3270 terminals for which an ID card reader has been defined. At 3270 display terminals, data can be input from a defined ID card reader at any time. If input from an ID card reader is requested, any other input is converted to K14.

"C" The input data is confidential and must not be visible at the terminal. This is achieved by blanking, by clearing the screen (the screen format is reset to 24x80), or by overwriting the input line (on printer terminals).

HOM-OUTPUT (816x, 9749, 975x ,9763 and 3270 display terminals only)

"N" The message is to be output in structured, heterogeneous form, i.e. a logical line is to be regarded as an output unit.

"Y" The message is to be output in unstructured, homogeneous form, i.e. the entire message is regarded as an output unit. The length of the message is restricted by the size of the system's output buffer.

FUNCTION-CODE is evaluated only if MODE="L", "E", "I" or "M"

"N" No function key code is to be transferred.

"Y" The logical function key code representing the key which initiated the transmission at the terminal is transmitted as the first character in the message.

NO-POSITION is evaluated only if MODE="L" or "M"

This parameter applies only to printers. In mixed mode this parameter is accepted only if the output mode is MODE="L". Under all other circumstances the parameter is ignored.

"N" The output message begins at the start of the next line.

"Y" The output message begins at the start of the current line.

CCSNAME is evaluated only if MODE="L","E","I","P","F" or "M"

Specifies the name of the coded character set to be used for this message. Specify the code name of the EBCDIC variant. The name of the corresponding ISO code variant is automatically rejected. The name must not be more than 8 bytes in length. Note that if the character set in use is changed, the screen is automatically cleared prior to the new output.

"ccsname"

Explicitly specifies the coded character set name. The options are:

ccsname Name of code of user's choice. Only EBCDIC codes are supported.

blank If no name is specified, standard mode is adopted by default. This is either a 7-bit mode or an 8-bit mode activated by means of the
MODIFY-TERMINAL-OPTIONS command.

=*EXTEND The extended standard code is automatically used.

For BS2000 V10.0A the extended standard code is the system standard code.

For BS2000/OSD-BC V1.0 the extended standard code is the user standard code.

CURSOR-POSITION is evaluated only if MODE="E" or "M"

Defines whether the cursor position is returned following the input.

In mixed mode this parameter is accepted only if EXTEND is the value for both input and output mode. The parameter is ignored under all other circumstances.

"N" No information on the cursor position is returned. The fields CURSOR-POS-LINE and CURSOR-POS-COLUM are undefined.

"Y" The fields CURSOR-POS-LINE and CURSOR-POS-COLUM are returned with the current cursor position (lines and columns) after the input.

READ-MODE is evaluated only if MODE="E" or "M"

Defines the physical read mode.

In mixed mode this parameter is accepted only if EXTEND is the value for both input and output mode. The parameter is ignored under all other circumstances.

"M" Only the modified fields are returned. The position on the screen is prefixed to each modified field in the user buffer (see descriptions of the logical control characters VERT-MOVE-IND and HORZ-MOVE-IND).

"U" All unprotected fields, including those that were not modified, are returned. In order to ascertain which values have been changed, you must compare the data you receive with your original data.

SCREEN-UPDATE is evaluated only if MODE="E" or "M"

For form output (FHS, SDF,...), you can define whether the entire screen will be updated or only the modified lines. In mixed mode this parameter is accepted only if EXTEND is the value for both input and output mode. The parameter is ignored under all other circumstances.

"N" The entire screen is updated for the first logical new page.

"Y" Only modified lines are updated. Once the text has been output the remainder of the line is padded with protected blanks.

AUTOMATIC-TABULATION is evaluated only if MODE="E" or "M"

Defines the automatic tabulator from an unprotected field to the next unprotected field. In mixed mode this parameter is accepted only if EXTEND is the value for both input and output mode. The parameter is ignored under all other circumstances.

"S" Processing depends on the value of the EXPROPOS parameter.

"N" The cursor does not jump automatically (even if EXPROPOS=N).

"Y" As soon as you enter a character at the end of an unprotected field, the cursor automatically jumps from this unprotected field to the next unprotected field (even if EXPROPOS=Y).

Note

- On 3270 terminals the cursor can always be moved to protected fields with the aid of the arrow keys. If, however, AUTOMATIC-TABULATION is "Y" the cursor automatically jumps from one input field to the next as soon as a character is entered at the end of the original field.
- The EXPROPOS parameter is ignored by 3270 terminals.

5 FORTRAN interface



The following sections are applicable to TIAM applications only.

The definitions of the FORTRAN data structure and the FORTRAN data fields correspond to those of the COBOL data structure and the COBOL data fields. This chapter, therefore, deals only with details of the FORTRAN language and syntax as they apply to these data structures. You will find detailed information on the data structures and data fields in the COBOL and ASSEMBLER descriptions.

Support for virtual terminals in FORTRAN is provided by the data structures FORCTRC for the logical control characters and VTSUCBF for the VTSU control block.

These data structures are available as include elements in the \$TSOS.SYSLIB.VTSU-B.110 library and can be copied into your application program.

The VTSU interface for FORTRAN is available as of VTSU V10.1A.

5.1 Logical control characters: FORCTRC

The FORCTRC data structure is copied into the application program by means of the following %INCLUDE statement. It contains the logical control characters for input and output in line mode, along with the printer control characters.

```
%INCLUDE $TSOS.SYSLIB.VTSU-B.110(FORCTRC)
```

The structure of FORCTRC is shown below. For detailed information on the individual parameters, see the COBOL description as of page 125 or the ASSEMBLER description as of page 55.

```
*****
*          FORCTRC          050          921231          VTSU-B          U *
*****
*
*          COPYRIGHT (C) SIEMENS NIXDORF INFORMATIONSSYSTEME AG 1991 *
*          ALL RIGHTS RESERVED *
*
*****
*
*
* LOGICAL RECORD DELIMITERS
*
CHARACTER*1  NEWLINE
INTEGER*1    NEWLINEINT          / 21 /
CHARACTER*1  NEWPAGE
INTEGER*1    NEWPAGEINT         / 12 /
CHARACTER*1  CONTSAMELINE
INTEGER*1    CONTSAMELINEINT    / 13 /
CHARACTER*1  CONTLINEN
INTEGER*1    CONTLINENINT       / 41 /
CHARACTER*1  CONTCOLN
INTEGER*1    CONTCOLNINT        / 42 /
CHARACTER*1  SHEETFEEDN
INTEGER*1    SHEETFEEDNINT      / 33 /
CHARACTER*1  CONTACTPOS
INTEGER*1    CONTACTPOSINT      / 32 /
*
* LOGICAL UNIT DELIMITERS
*
CHARACTER*1  EMPHLYOUT1
INTEGER*1    EMPHLYOUT1INT      / 29 /
CHARACTER*1  EMPHLYOUT2
INTEGER*1    EMPHLYOUT2INT      / 31 /
CHARACTER*1  EMPHLYOUT3
INTEGER*1    EMPHLYOUT3INT      / 19 /
CHARACTER*1  EMPHLYOUT4
INTEGER*1    EMPHLYOUT4INT      / 20 /
CHARACTER*1  NORMALLAYOUT
INTEGER*1    NORMALLAYOUTINT    / 30 /
CHARACTER*1  DARKLAYOUT
INTEGER*1    DARKLAYOUTINT      / 18 /
CHARACTER*1  PARTLINEUP
```

	INTEGER*1	PARTLINEUPINT	/ 44 /
	CHARACTER*1	PARTLINEDOWN	
	INTEGER*1	PARTLINEDOWNINT	/ 43 /
*			
	CHARACTER*1	SECCHARSET	
	INTEGER*1	SECCHARSETINT	/ 14 /
	CHARACTER*1	NORMCHARSET	
	INTEGER*1	NORMCHARSETINT	/ 15 /
*			
	CHARACTER*1	STARTPROTA	
	INTEGER*1	STARTPROTAINT	/ 54 /
	CHARACTER*1	ENDPROTAREA	
	INTEGER*1	ENDPROTAREAINT	/ 8 /
	CHARACTER*1	STARTNUMDATA	
	INTEGER*1	STARTNUMDATAINT	/ 17 /
*			
	CHARACTER*1	STARTCHSETN	
	INTEGER*1	STARTCHSETNINT	/ 6 /
	CHARACTER*1	SELECTCOLOUR	
	INTEGER*1	SELECTCOLOURINT	/ 23 /
	CHARACTER*6	DIMENSION2480	/ 'D24080' /
	CHARACTER*6	DIMENSION3280	/ 'D32080' /
	CHARACTER*6	DIMENSION4380	/ 'D43080' /
	CHARACTER*6	DIMENSION27132	/ 'D27132' /
*			
	CHARACTER*1	TRANSPARENTOUT	/ 'T' /
	CHARACTER*1	REPEATSYMBOL	/ 'R' /
*			
	CHARACTER*1	STARTLOCATTR	
	INTEGER*1	STARTLOCATTRINT	/ 9 /
	CHARACTER*1	ENDLOCATTR	
	INTEGER*1	ENDLOCATTRINT	/ 10 /
*			
	CHARACTER*1	VERTMOVEIND	
	INTEGER*1	VERTMOVEINDINT	/ 36 /
	CHARACTER*1	HORIZMOVEIND	
	INTEGER*1	HORIZMOVEINDINT	/ 35 /
	CHARACTER*1	LEFTMARGIN	
	INTEGER*1	LEFTMARGININT	/ 56 /
	CHARACTER*1	STARTPROPT	
	INTEGER*1	STARTPROPTINT	/ 26 /
	CHARACTER*1	ENDPROPTYPE	
	INTEGER*1	ENDPROPTYPEINT	/ 27 /
	CHARACTER*1	MAXLINELEN	
	INTEGER*1	MAXLINELENINT	/ 51 /
	CHARACTER*1	MAXLINENUM	
	INTEGER*1	MAXLINENUMINT	/ 53 /
	CHARACTER*1	STARTNELETT	
	INTEGER*1	STARTNELETTINT	/ 57 /
	CHARACTER*1	ENDNEARLETT	
	INTEGER*1	ENDNEARLETTINT	/ 59 /
*			
*			
*		SPECIAL FUNCTIONS	
	CHARACTER*1	EXTLOGCHAR	
	INTEGER*1	EXTLOGCHARINT	/ 62 /
	CHARACTER*1	DELETECHAR	
	INTEGER*1	DELETECHARINT	/ 7 /

```

CHARACTER*1 BACKSPACE
INTEGER*1 BACKSPACEINT / 22 /
CHARACTER*1 SUBSTITUTE
INTEGER*1 SUBSTITUTEINT / 63 /

```

*
*
*

PHYSICAL UNIT DELIMITERS

```

CHARACTER*1 PHYSESC
INTEGER*1 PHYSESCINT / 39 /
CHARACTER*1 PHYSDC4
INTEGER*1 PHYSDC4INT / 60 /
CHARACTER*1 PHYSHT
INTEGER*1 PHYSHTINT / 5 /
CHARACTER*1 PHYSVT
INTEGER*1 PHYSVTINT / 11 /

```

*
*
*

DISPLAY ATTRIBUTES

```

CHARACTER*1 DISPLAYATTR / 'I' /
CHARACTER*1 FLASHING
INTEGER*1 FLASHINGINT / 1 /
CHARACTER*1 UNDERSCORED
INTEGER*1 UNDERSCOREDINT / 2 /
CHARACTER*1 BLANKED
INTEGER*1 BLANKEDINT / 4 /
CHARACTER*1 REDUCEDINTENS
INTEGER*1 REDUCEDINT / 8 /
CHARACTER*1 INVERSEMODE
INTEGER*1 INVERSEMODEINT / 16 /
CHARACTER*1 RESET
INTEGER*1 RESETINT / 0 /

```

*
*
*

FIELD CHARACTERISTICS

```

CHARACTER*1 FIELDCHARACT / 'F' /
CHARACTER*1 PROTNOTSEND
INTEGER*1 PROTNOTSENDINT / 1 /
CHARACTER*1 PROTSENDABLE
INTEGER*1 PROTSENDINT / 32 /
CHARACTER*1 NUMERIC
INTEGER*1 NUMERICINT / 2 /
CHARACTER*1 PREMODIFIED
INTEGER*1 PREMODIFIEDINT / 4 /
CHARACTER*1 MARKABLE
INTEGER*1 MARKABLEINT / 8 /
CHARACTER*1 PRINTABLE
INTEGER*1 PRINTABLEINT / 16 /
CHARACTER*1 INPUT
INTEGER*1 INPUTINT / 0 /
CHARACTER*1 AUTOSKIP
INTEGER*1 AUTOSKIPINT / 64 /

```

*
*
*

```

EQUIVALENCE (NEWLINE ( 1: 1), NEWLINEINT)
EQUIVALENCE (NEWPAGE ( 1: 1), NEWPAGEINT)
EQUIVALENCE (CONTSAMELINE ( 1: 1), CONTSAMELINEINT)
EQUIVALENCE (CONTLINEN ( 1: 1), CONTLINENINT)

```



```

EQUIVALENCE (CONTCOLN      ( 1: 1), CONTCOLNINT)
EQUIVALENCE (SHEETFEEDN   ( 1: 1), SHEETFEEDNINT)
EQUIVALENCE (CONTACTPOS   ( 1: 1), CONTACTPOSINT)
EQUIVALENCE (EMPHLAYOUT1  ( 1: 1), EMPHLYOUT1INT)
EQUIVALENCE (EMPHLAYOUT2  ( 1: 1), EMPHLYOUT2INT)
EQUIVALENCE (EMPHLAYOUT3  ( 1: 1), EMPHLYOUT3INT)
EQUIVALENCE (EMPHLAYOUT4  ( 1: 1), EMPHLYOUT4INT)
EQUIVALENCE (NORMALLAYOUT ( 1: 1), NORMALLAYOUTINT)
EQUIVALENCE (DARKLAYOUT   ( 1: 1), DARKLAYOUTINT)
EQUIVALENCE (PARTLINEUP   ( 1: 1), PARTLINEUPINT)
EQUIVALENCE (PARTLINEDOWN ( 1: 1), PARTLINEDOWNINT)
EQUIVALENCE (SECCHARSET   ( 1: 1), SECCHARSETINT)
EQUIVALENCE (NORMCHARSET  ( 1: 1), NORMCHARSETINT)
EQUIVALENCE (STARTPROTA   ( 1: 1), STARTPROTAINT)
EQUIVALENCE (ENDPROTAREA  ( 1: 1), ENDPROTAREAINT)
EQUIVALENCE (STARTNUMDATA ( 1: 1), STARTNUMDATAINT)
EQUIVALENCE (STARTCHSETN  ( 1: 1), STARTCHSETNINT)
EQUIVALENCE (STARTLOCATTR ( 1: 1), STARTLOCATTRINT)
EQUIVALENCE (ENDLOCATTR   ( 1: 1), ENDLOCATTRINT)
EQUIVALENCE (VERTMOVEIND  ( 1: 1), VERTMOVEINDINT)
EQUIVALENCE (HORIZMOVEIND ( 1: 1), HORIZMOVEINDINT)
EQUIVALENCE (LEFTMARGIN   ( 1: 1), LEFTMARGININT)
EQUIVALENCE (STARTPROPT   ( 1: 1), STARTPROPTINT)
EQUIVALENCE (ENDPROPTYPE  ( 1: 1), ENDPROPTYPEINT)
EQUIVALENCE (MAXLINELEN   ( 1: 1), MAXLINELENINT)
EQUIVALENCE (MAXLINENUM   ( 1: 1), MAXLINENUMINT)
EQUIVALENCE (STARTNELETT  ( 1: 1), STARTNELETTINT)
EQUIVALENCE (ENDNEARLETT ( 1: 1), ENDNEARLETTINT)
EQUIVALENCE (EXTLOGCHAR   ( 1: 1), EXTLOGCHARINT)
EQUIVALENCE (DELETECHAR   ( 1: 1), DELETECHARINT)
EQUIVALENCE (BACKSPACE    ( 1: 1), BACKSPACEINT)
EQUIVALENCE (SUBSTITUTE    ( 1: 1), SUBSTITUTEINT)
EQUIVALENCE (PHYSESC      ( 1: 1), PHYSESCINT)
EQUIVALENCE (PHYSDC4      ( 1: 1), PHYSDC4INT)
EQUIVALENCE (PHYSHT       ( 1: 1), PHYSHTINT)
EQUIVALENCE (PHYSVT       ( 1: 1), PHYSVTINT)
EQUIVALENCE (SELECTCOLOUR ( 1: 1), SELECTCOLOURINT)
EQUIVALENCE (FLASHING     ( 1: 1), FLASHINGINT)
EQUIVALENCE (UNDERScoreD  ( 1: 1), UNDERScoreDINT)
EQUIVALENCE (BLANKED      ( 1: 1), BLANKEDINT)
EQUIVALENCE (REDUCEDINTENS ( 1: 1), REDUCEDINTENS)
EQUIVALENCE (INVERSEMODE  ( 1: 1), INVERSEMODEINT)
EQUIVALENCE (RESET        ( 1: 1), RESETINT)
EQUIVALENCE (PROTNOTSEND  ( 1: 1), PROTNOTSENDINT)
EQUIVALENCE (PROTSENDABLE ( 1: 1), PROTSENDABLE)
EQUIVALENCE (NUMERIC      ( 1: 1), NUMERICINT)
EQUIVALENCE (PREMODIFIED  ( 1: 1), PREMODIFIEDINT)
EQUIVALENCE (MARKABLE     ( 1: 1), MARKABLEINT)
EQUIVALENCE (PRINTABLE    ( 1: 1), PRINTABLEINT)
EQUIVALENCE (INPUT        ( 1: 1), INPUTINT)
EQUIVALENCE (AUTOSKIP     ( 1: 1), AUTOSKIPINT)

```

*

*

5.2 VTSU control block: VTSUCBF

The **TIAM** access method offers you a VTSUCB for FORTRAN applications. In **FORTRAN applications** you can include the VTSUCB in the RDATA, WROUT and WRTRD macros, for example:

```
EXTERNAL WROUT
CALL WROUT (TIAMCONTROLINFO, user area, VTSUCB)
```

Parameters having a value other than the default must be set in the VTSUCB by means of a statement.

The VTSU-CONTROL-BLOCK data structure is copied into the application program by means of the following %INCLUDE statement:

```
%INCLUDE $TSOS.SYSLIB.VTSU-B.110(VTSUCBF)
```

The structure of the VTSUCB is shown below. For detailed information on the individual parameters, see the COBOL description as of page 153 or the ASSEMBLER description as of page 89.

Data structure of the VTSUCB

```
*****
*      VTSUCBF          050          921209          VTSU-B          U *
*****
*
*      COPYRIGHT (C) SIEMENS NIXDORF INFORMATIONSSYSTEME AG 1992 *
*                  ALL RIGHTS RESERVED *
*
*****
*
*      CHARACTER*48      VTSUCB
*
*      CHARACTER*8      STANDARDHEADER
*      CHARACTER*8      STANDARDHEADER
*      INTEGER*2        FCTIONUNIT      /49/
*      INTEGER*1        FCTIONNBER     /1/
*      INTEGER*1        FCTIONVERS     /4/
*      CHARACTER*4      RETURNCODES
*      CHARACTER*2      SUBRETCODES
*      INTEGER*1        SUBCODE2       /-1/
*      INTEGER*1        SUBCODE1       /-1/
*      INTEGER*2        MAINRETCODE    /-1/
*
*      END STANDARD HEADER
*
*      INTEGER*2        BLOCKLEN       /48/
*
*      CHARACTER*1      INPUTMODE
*
*      INPUT MODE FOR MIXED MODE
*
*      CHARACTER*1      OUTPUTMODE
*
*      OUTPUT MODE FOR MIXED MODE
*
*      CHARACTER*1      MODE           /'L'/
*
*      MODE OF MESSAGE
```

```

*      CHARACTER*1      HARDCOPYFCT      /'N'/
*                                     HARDCOPY FUNCTION
*      CHARACTER*1      BELLFCT         /'N'/
*                                     BELL FUNCTION
*      CHARACTER*1      NOLOGCHARS      /'N'/
*                                     NO LOG CHARS TO INTERPRET
*      CHARACTER*2      RETURNBYTES     /'N'/
*                                     RETURN INFO BYTES
*      CHARACTER*1      RETINFO         /'N'/
*                                     RETURN INFORMATION FCT
*      CHARACTER*1      LOCALCHARS      /'N'/
*                                     INPUT OF LOCAL CHARACTERS
*      CHARACTER*1      OUTPUTHEADER    /'N'/
*                                     OUTPUT HEADER FUNCTION
*      CHARACTER*1      CODETRANSLATION /' '/
*                                     CODE TRANSLATION FUNCTION
*      CHARACTER*1      INPUTHEADER     /'Y'/
*                                     INPUT HEADER FUNCTION
*      CHARACTER*1      LOWERCHARS      /'N'/
*                                     LOWER CHARACTERS FUNCTION
*      CHARACTER*1      SPECIALINPUT    /'N'/
*                                     SPECIAL INPUT FUNCTION
*      CHARACTER*1      FCTIONCODE      /'N'/
*                                     FUNCTION CODE
*      CHARACTER*1      HOMOUTPUT       /'N'/
*                                     HOMOGENEOUS OUTPUT
*      CHARACTER*1      NOPOS           /'N'/
*                                     OUTPUT NEXT/CURRENT LINE
*      CHARACTER*8      CCSNAME         /'          '/
*
*      CHARACTER*1      CURSORPOSITION  /'N'/
*                                     CURSOR POSITION REQUESTED
*      INTEGER*1       CURSLINEPOS      /'N'/
*                                     CURSOR LINE POSITION
*      INTEGER*1       CURSCOLUMNPOS    /'N'/
*                                     CURSOR COLUMN POSITION
*      CHARACTER*1     READMODE         /'U'/
*                                     READ UNPROTECTED/MODIFIED
*      CHARACTER*1     SCREENUPDATE     /'N'/
*                                     SCREEN UPDATE/OVERWRITE
*      CHARACTER*1     WARBYTEINFO      /'N'/
*                                     WAR BYTE INFO REQUESTED
*      CHARACTER*1     WARBYTEVALUE     /'N'/
*                                     RETURNED WAR BYTE VALUE
*      CHARACTER*1     AUTOTAB          /'S'/
*                                     AUTOMATIC TABULATION REQUEST
* *****
*      CHARACTER*1     LINE             /'L'/
*                                     LINE MODE
*      CHARACTER*1     EXTEND           /'E'/
*                                     EXTENDED LINE MODE
*      CHARACTER*1     INFO             /'I'/
*                                     INFO LINE MESSAGE
*      CHARACTER*1     PHYS             /'P'/
*                                     PHYSICAL MODE
*      CHARACTER*1     TRANS           /'T'/
*                                     TRANSPARENT MODE

```

```

*      CHARACTER*1      FORM          /'F'/
*                               FORM MODE
*      CHARACTER*1      CHIP          /'C'/
*                               CHIP CARD MODE
*      CHARACTER*1      MIXED         /'M'/
*                               MIXED MODE
*
*      CHARACTER*1      NOHC          /'N'/
*                               NO HARDCOPY
*      CHARACTER*1      YESHC         /'Y'/
*                               LOCAL / CENTRAL HC
*      CHARACTER*1      NOBELL        /'N'/
*                               NO BELL
*      CHARACTER*1      YESBELL       /'Y'/
*                               BELL AFTER OUTPUT
*      CHARACTER*1      NONOLOG       /'N'/
*                               LOG. CHAR. TO INTERPRET
*      CHARACTER*1      YESNOLOG      /'Y'/
*                               NO LOG CHAR TO INTERPRET
*      CHARACTER*1      NORIN         /'N'/
*                               NO RETURN INFORMATION
*      CHARACTER*1      YESRIN        /'Y'/
*                               RETURN INFORM. REQUIRED
*      CHARACTER*1      NOLOC         /'N'/
*                               NO LOCAL CHAR REQUIRED
*      CHARACTER*1      YESLOC        /'Y'/
*                               LOCAL CHARACTERS REQUIRED
*      CHARACTER*1      NOOHDR        /'N'/
*                               NO OUTPUT HEADER IN USER
*      CHARACTER*1      YESOHDR       /'Y'/
*                               OUTPUT HEADER IN USER MSG
*      CHARACTER*1      NOTRANSlation /'N'/
*                               NO CODE TRANSL. TO FROM CCS
*      CHARACTER*1      YESTRANSlation /'Y'/
*                               CODE TRANSLATION REQUIRED
*      CHARACTER*1      NOIHDR        /'N'/
*                               NO INPUT HEADER REQUIRED
*      CHARACTER*1      YESIHDR       /'Y'/
*                               INPUT HEADER REQUIRED
*      CHARACTER*1      NOLOW         /'N'/
*                               TRANSLATE LOWER CHARACTERS
*      CHARACTER*1      YESLOW        /'Y'/
*                               RETAIN LOWER CHARACTERS
*      CHARACTER*1      NOSPECINPUT   /'N'/
*                               NO SPECIAL INPUT
*      CHARACTER*1      IDCARDREADER  /'I'/
*                               INPUT FROM ID-CARD READER
*      CHARACTER*1      CONFIDENTIAL  /'C'/
*                               CONFIDENTIAL INPUT
*      CHARACTER*1      NOFC          /'N'/
*                               NO FUNCTION CODE REQUIRED
*      CHARACTER*1      YESFC         /'Y'/
*                               FUNCTION CODE REQUIRED
*      CHARACTER*1      NOHOM         /'N'/
*                               NO HOM. OUTPUT REQUIRED
*      CHARACTER*1      YESHOM        /'Y'/
*                               HOM. OUTPUT REQUIRED
*      CHARACTER*1      NONOPOS       /'N'/

```

```

*                                     OUTPUT ON NEXT LINE
CHARACTER*1      YESNOPOS      /'Y'/
*
*                                     OUTPUT ON CURRENT LINE
CHARACTER*1      NOCURSOR      /'N'/
*
*                                     CURSOR POS. NOT REQUESTED
CHARACTER*1      YESCURSOR     /'Y'/
*
*                                     CURSOR POSITION REQUESTED
CHARACTER*1      READMODIFIED  /'M'/
*
*                                     READ MODIFIED MODE
CHARACTER*1      READUNPROTECTED /'U'/
*
*                                     READ UNPROTECTED MODE
CHARACTER*1      NOSCREENUPD   /'N'/
*
*                                     SCREEN OVERWRITE
CHARACTER*1      YESSCREENUPD  /'Y'/
*
*                                     SCREEN UPDATE
CHARACTER*1      NOWARINFO     /'N'/
*
*                                     NO WAR BYTE INFO
CHARACTER*1      YESWARINFO    /'Y'/
*
*                                     SCREEN OVERWRITE
CHARACTER*1      STDAUTOTAB    /'S'/
*
*                                     STANDARD TABULATION
CHARACTER*1      NOAUTOTAB     /'N'/
*
*                                     NO AUTOMATIC TABULATION
CHARACTER*1      YESAUTOTAB    /'Y'/
*
*                                     AUTOMATIC TABULATION
*****
*
EQUIVALENCE (VTSUCB ( 1: 8),STANDARDHEADER)
EQUIVALENCE (STANDARDHEADER (1:2),FCTIONUNIT)
EQUIVALENCE (STANDARDHEADER (3:3),FCTIONNBER)
EQUIVALENCE (STANDARDHEADER (4:4),FCTIONVERS)
EQUIVALENCE (STANDARDHEADER (5:8),RETURNCODES)
EQUIVALENCE (RETURNCODES (1:2),SUBRETCODES)
EQUIVALENCE (SUBRETCODES (1:1),SUBCODE2)
EQUIVALENCE (SUBRETCODES (2:2),SUBCODE1)
EQUIVALENCE (RETURNCODES (3:4),MAINRETCODE)
EQUIVALENCE (VTSUCB ( 9:10),BLOCKLEN)
EQUIVALENCE (VTSUCB (11:11),INPUTMODE)
EQUIVALENCE (VTSUCB (12:12),OUTPUTMODE)
EQUIVALENCE (VTSUCB (17:17),MODE)
EQUIVALENCE (VTSUCB (18:18),HARDCOPYFCT)
EQUIVALENCE (VTSUCB (19:19),BELLFCT)
EQUIVALENCE (VTSUCB (20:20),NOLOGCHARS)
EQUIVALENCE (VTSUCB (21:22),RETURNBYTES)
EQUIVALENCE (VTSUCB (23:23),RETINFO)
EQUIVALENCE (VTSUCB (24:24),LOCALCHARS)
EQUIVALENCE (VTSUCB (25:25),OUTPUTHEADER)
EQUIVALENCE (VTSUCB (26:26),CODETRANSLATION)
EQUIVALENCE (VTSUCB (27:27),INPUTHEADER)
EQUIVALENCE (VTSUCB (28:28),LOWERCHARS)
EQUIVALENCE (VTSUCB (29:29),SPECIALINPUT)
EQUIVALENCE (VTSUCB (30:30),FCTIONCODE)
EQUIVALENCE (VTSUCB (31:31),HOMOUTPUT)
EQUIVALENCE (VTSUCB (32:32),NOPOS)
EQUIVALENCE (VTSUCB (33:40),CCSNAME)
EQUIVALENCE (VTSUCB (41:41),CURSORPOSITION)
EQUIVALENCE (VTSUCB (42:42),CURSLINEPOS)
EQUIVALENCE (VTSUCB (43:43),CURSCOLUMNPOS)

```

```
EQUIVALENCE (VTSUCB (44:44), READMODE)
EQUIVALENCE (VTSUCB (45:45), SCREENUPDATE)
EQUIVALENCE (VTSUCB (46:46), WARBYTEINFO)
EQUIVALENCE (VTSUCB (47:47), WARBYTEVALUE)
EQUIVALENCE (VTSUCB (48:48), AUTOTAB)
```

6 PL/I interface



The following sections are applicable to TIAM applications only.

The definitions of the PL/I data structure and the PL/I data fields correspond to those of the COBOL data structure and the COBOL data fields. This chapter, therefore, deals only with details of the PL/I language and syntax as they apply to these data structures. You will find detailed information on the data structures and data fields in the COBOL and ASSEMBLER descriptions.

Support for virtual terminals in PL/I is provided by the data structures PL1CTRC for the logical control characters and VTSUCBP for the VTSU control block.

These data structures are available as include elements in the \$TSOS.SYSLIB.VTSU-B.110 library and can be copied into your application program.

The VTSU interface for PL/I is available as of VTSU V10.1A

6.1 Logical control characters: PL1CTRC

The PL1CTRC data structure is copied into the application program by means of the following %INCLUDE statement. It contains the logical control characters for input and output in line mode, along with the printer control characters.

```
%INCLUDE $TSOS.SYSLIB.VTSU-B.110(PL1CTRC)
```

The structure of PL1CTRC is shown below. For detailed information on the individual parameters, see the COBOL description as of page 125 or the ASSEMBLER description as of page 55.

```

/*****
/*      PL1CTRC          050          921231          VTSU-B          U */
/*****
/*
/*      COPYRIGHT (C) SIEMENS NIXDORF INFORMATIONSSYSTEME AG 1991      */
/*
/*      ALL RIGHTS RESERVED                                          */
/*
/*
/*****

DCL  DIMENSION_24_80          CHARACTER (6) STATIC INIT ('D24080');
DCL  DIMENSION_32_80          CHARACTER (6) STATIC INIT ('D32080');
DCL  DIMENSION_43_80          CHARACTER (6) STATIC INIT ('D43080');
DCL  DIMENSION_27_132        CHARACTER (6) STATIC INIT ('D27132');
DCL  TRANSPARENT_OUT          CHARACTER STATIC INIT ('T');
DCL  REPEAT_SYMBOL            CHARACTER STATIC INIT ('R');
DCL  DISPLAY_ATTRIBUTES       CHARACTER STATIC INIT ('I');
DCL  FIELD_CHARACTERISTICS    CHARACTER STATIC INIT ('F');

DCL  CHARACTER_SET_HEX (55)   BIT (8) STATIC INITIAL
      ( 21, 12, 13, 41, 42, 33, 32,
        29, 31, 19, 20, 30, 18, 44, 43,
         14, 15,
         54, 8, 17, 6, 9, 10,
         36, 35, 56, 26, 27, 51, 53, 57, 59,
         62, 7, 22, 63,
         39, 60, 5, 11,
         23,
         1, 2, 4, 8, 16, 0,
         1, 32, 2, 4, 8, 16, 0, 64);
DCL 1 CHARACTER_SET_CHARS DEFINED CHARACTER_SET_HEX (1),

      /* Logical record delimiters          */
2 NEW_LINE          CHARACTER (1),
2 NEW_PAGE          CHARACTER (1),
2 CONT_SAME_LINE    CHARACTER (1),
2 CONT_LINE_N       CHARACTER (1),
2 CONT_COL_N        CHARACTER (1),
2 SHEET_FEED_N      CHARACTER (1),
2 CONT_ACT_POS      CHARACTER (1),

      /* Logical unit delimiters          */
2 EMPH_LAYOUT1      CHARACTER (1),
2 EMPH_LAYOUT2      CHARACTER (1),
2 EMPH_LAYOUT3      CHARACTER (1),

```



```

2 EMPH_LAYOUT4           CHARACTER (1),
2 NORMAL_LAYOUT         CHARACTER (1),
2 DARK_LAYOUT           CHARACTER (1),
2 PART_LINE_UP          CHARACTER (1),
2 PART_LINE_DOWN        CHARACTER (1),

2 SECOND_CHAR_SET       CHARACTER (1),
2 NORMAL_CHAR_SET       CHARACTER (1),

2 START_PROT_AREA       CHARACTER (1),
2 END_PROT_AREA         CHARACTER (1),
2 START_NUM_DATA        CHARACTER (1),
2 START_CHARSET_N       CHARACTER (1),
2 START_LOCAL_ATTR      CHARACTER (1),
2 END_LOCAL_ATTR        CHARACTER (1),

2 VERT_MOVE_IND         CHARACTER (1),
2 HORIZ_MOVE_IND        CHARACTER (1),
2 LEFT_MARGIN           CHARACTER (1),
2 START_PROP_TYPE       CHARACTER (1),
2 END_PROP_TYPE         CHARACTER (1),
2 MAX_LINE_LEN          CHARACTER (1),
2 MAX_LINE_NUM          CHARACTER (1),
2 START_NEAR_LETT       CHARACTER (1),
2 END_NEAR_LETT        CHARACTER (1),

/* Special functions */
2 EXTEND_LOG_CHAR       CHARACTER (1),
2 DELETE_CHAR           CHARACTER (1),
2 BACKSPACE             CHARACTER (1),
2 SUBSTITUTE            CHARACTER (1),

/* Physical unit delimiters */
2 PHYS_ESC              CHARACTER (1),
2 PHYS_DC4              CHARACTER (1),
2 PHYS_HT               CHARACTER (1),
2 PHYS_VT               CHARACTER (1),

2 SELECT_COLOUR         CHARACTER (1),

/* Display attributes */
2 FLASHING              CHARACTER (1),
2 UNDERSCORED          CHARACTER (1),
2 BLANKED               CHARACTER (1),
2 REDUCED_INTENSITY     CHARACTER (1),
2 INVERSE_MODE          CHARACTER (1),
2 RESET                 CHARACTER (1),

/* Field characteristics */
2 PROTECTED_NOT_SENDABLE CHARACTER (1),
2 PROTECTED_SENDABLE    CHARACTER (1),
2 NUMERIC               CHARACTER (1),
2 PREMODIFIED           CHARACTER (1),
2 MARKABLE              CHARACTER (1),
2 PRINTABLE             CHARACTER (1),
2 INPUT                 CHARACTER (1),
2 AUTOSKIP              CHARACTER (1);

```

6.2 VTSU control block: VTSUCBP

The **TIAM** access method also offers you a VTSUCB for PL/I applications. In **PL/I applications** you can include the VTSUCB in the RDATA, WROUT and WRTRD macros, for example:

```
DCL WROUT ENTRY EXTERNAL OPTIONS (ASSEMBLER);
CALL WROUT (TIAM_CONTROL_INFO, user area, VTSUCB);
```

Parameters having a value other than the default must be set in the VTSUCB by means of a statement.

The VTSU-CONTROL-BLOCK data structure is copied into the application program by means of the following %INCLUDE statement:

```
%INCLUDE $TSOS.SYSLIB.VTSU-B.110(VTSUCBP)
```

The structure of the VTSUCB is shown below. For detailed information on the individual parameters, see the COBOL description as of page 153 or the ASSEMBLER description as of page 89.

Data structure of the VTSUCB

```

/*****
/*      VTSUCBP          050          921209          VTSU-B          U */
/*****
/*
/*      COPYRIGHT (C) SIEMENS NIXDORF INFORMATIONSSYSTEME AG 1992      */
/*      ALL RIGHTS RESERVED                                          */
/*
/*****
DCL  1  VTSUCB,
      2  STANDARD_HEADER,
          3  FUNCTION_UNIT      BIN FIXED(15) INIT(49),
          3  FUNCTION_VERS     BIN FIXED(15) INIT(260),
          3  RETURN_CODE,
              4  SUBCODES      BIN FIXED(15) INIT(-1),
              4  MAINCODE     BIN FIXED(15) INIT(-1),
      2  BLOCKLEN              BIN FIXED(15) INIT(48),
                                  /*      LENGTH OF VTSUCB          */
      2  INPUT_MODE            CHAR(1),
                                  /*      INPUT MODE FOR MIXED MODE  */
      2  OUTPUT_MODE           CHAR(1),
                                  /*      OUTPUT MODE FOR MIXED MODE */
      2  FILLER1                CHAR(4),
                                  /*      RESERVED FOR RETURN_INFO  */
      2  MODE                    CHAR(1) INIT('L'),
                                  /*      MODE OF MESSAGE          */
      2  HARDCOPY                CHAR(1) INIT('N'),
                                  /*      HARDCOPY FUNCTION        */
      2  BELL                    CHAR(1) INIT('N'),
                                  /*      BELL FUNCTION            */

```

```

2 NO_LOG_CHARS CHAR(1) INIT('N'),
/* NO LOG CHARS TO INTERPRET */
2 RETURN_BYTES CHAR(2),
/* RETURN INFO BYTES */
2 RET_INFO CHAR(1) INIT('N'),
/* RETURN INFORMATION FCT */
2 LOCAL_CHARS CHAR(1) INIT('N'),
/* INPUT OF LOCAL CHARACTERS */
2 OUTPUT_HEADER CHAR(1) INIT('N'),
/* OUTPUT HEADER FUNCTION */
2 CODE_TRANSLATION CHAR(1) INIT(' '),
/* CODE TRANSLATION FUNCTION */
2 INPUT_HEADER CHAR(1) INIT('Y'),
/* INPUT HEADER FUNCTION */
2 LOWER_CHARS CHAR(1) INIT('N'),
/* LOWER CHARACTERS FUNCTION */
2 SPECIAL_INPUT CHAR(1) INIT('N'),
/* SPECIAL INPUT FUNCTION */
2 FUNCTION_CODE CHAR(1) INIT('N'),
/* FUNCTION CODE */
2 HOM_OUTPUT CHAR(1) INIT('N'),
/* HOMOGENEOUS OUTPUT */
2 NO_POSITION CHAR(1) INIT('N'),
/* OUTPUT ON CURRENT LINE */
2 CCSNAME CHAR(8) INIT(' '),
2 CURSOR_POSITION CHAR(1) INIT('N'),
/* CURSOR POSITION REQUESTED */
2 CURSOR_POS_LINE BIT(8),
/* CURSOR LINE POS */
2 CURSOR_POS_COLUMN BIT(8),
/* CURSOR COLUMN POS */
2 READ_MODE CHAR(1) INIT('U'),
/* READ MODIFIED/UNPROTECTED */
2 SCREEN_UPDATE CHAR(1) INIT('N'),
/* SCREEN UPDATE EXT.LINE MSG */
2 WARBYTE_INFO CHAR(1) INIT('N'),
/* WAR BYTE VALUE REQUESTED */
2 WARBYTE_VALUE CHAR(1),
/* RETURNED WAR BYTE VALUE */
2 AUTOMATIC_TABULATION CHAR(1);
/* AUTOMATIC TABUL. REQUEST */
/*
*/
DCL LINE CHAR(1) STATIC INIT('L'); /* LINE MODE */
DCL EXTEND CHAR(1) STATIC INIT('E'); /* EXTENDED LINE MODE */
DCL INFO CHAR(1) STATIC INIT('I'); /* INFO LINE MESSAGE */
DCL PHYS CHAR(1) STATIC INIT('P'); /* PHYSICAL MODE */
DCL TRANS CHAR(1) STATIC INIT('T'); /* TRANSPARENT MODE */
DCL FORM CHAR(1) STATIC INIT('F'); /* FORM MODE */
DCL CHIP CHAR(1) STATIC INIT('C'); /* CHIP CARD MODE */
DCL MIXED CHAR(1) STATIC INIT('M'); /* MIXED MODE */
/* */
DCL NO_HC CHAR(1) STATIC INIT('N'); /* NO HARDCOPY */
DCL YES_HC CHAR(1) STATIC INIT('Y'); /* LOCAL / CENTRAL HC */
DCL NO_BELL CHAR(1) STATIC INIT('N'); /* NO BELL */
DCL YES_BELL CHAR(1) STATIC INIT('Y'); /* BELL AFTER OUTPUT */
DCL NO_NOLOG CHAR(1) STATIC INIT('N'); /* LOG. CHAR. TO INTERPRET */
DCL YES_NOLOG CHAR(1) STATIC INIT('Y'); /* NO LOG CHAR TO INTERPRET */

```

```

DCL NO_RIN      CHAR(1) STATIC INIT('N'); /* NO RETURN INFORMATION */
DCL YES_RIN     CHAR(1) STATIC INIT('Y'); /* RETURN INFORM. REQUIRED */
DCL NO_LOC      CHAR(1) STATIC INIT('N'); /* NO LOCAL CHAR REQUIRED */
DCL YES_LOC     CHAR(1) STATIC INIT('Y'); /* LOCAL CHARACT. REQUIRED */
DCL NO_OHDR     CHAR(1) STATIC INIT('N'); /* NO OUT. HEADER IN USER */
DCL YES_OHDR    CHAR(1) STATIC INIT('Y'); /* OUT. HEADER IN USER MSG */
DCL NO_TRANSLATION CHAR(1) STATIC INIT('N');
/*                               NO CODE TRANSL. TO/FROM CCS */
DCL YES_TRANSLATION CHAR(1) STATIC INIT('Y');
/*                               CODE TRANSLATION REQUIRED */
DCL NO_IHDR     CHAR(1) STATIC INIT('N'); /* NO INPUT HEADER REQUIRED */
DCL YES_IHDR    CHAR(1) STATIC INIT('Y'); /* INPUT HEADER REQUIRED */
DCL NO_LOW      CHAR(1) STATIC INIT('N'); /* TRANSLATE LOWER-CASES */
DCL YES_LOW     CHAR(1) STATIC INIT('Y'); /* RETAIN LOWER CHARACTERS */
DCL NO_SPEC_INPUT CHAR(1) STATIC INIT('N'); /* NO SPECIAL INPUT */
DCL ID_CARD_READER CHAR(1) STATIC INIT('I');
/*                               INPUT FROM ID_CARD READER */
DCL CONFIDENTIAL CHAR(1) STATIC INIT('C'); /* CONFIDENTIAL INPUT */
DCL NO_FC        CHAR(1) STATIC INIT('N'); /* NO FUNCT. CODE REQUIRED */
DCL YES_FC       CHAR(1) STATIC INIT('Y'); /* FUNCTION CODE REQUIRED */
DCL NO_HOM       CHAR(1) STATIC INIT('N'); /* NO HOM. OUTPUT REQUIRED */
DCL YES_HOM      CHAR(1) STATIC INIT('Y'); /* HOM. OUTPUT REQUIRED */
DCL NO_NOPOS     CHAR(1) STATIC INIT('N'); /* OUTPUT ON NEXT LINE */
DCL YES_NOPOS    CHAR(1) STATIC INIT('Y'); /* OUTPUT ON CURRENT LINE */
DCL NO_CURSOR    CHAR(1) STATIC INIT('N'); /* CURSOR POS NOT REQUESTED */
DCL YES_CURSOR   CHAR(1) STATIC INIT('Y'); /* CURSOR POS REQUESTED */
DCL READ_MODIFIED CHAR(1) STATIC INIT('M'); /* READ MODE FOR .. */
DCL READ_UNPROTECTED CHAR(1) STATIC INIT('U'); /*... EXTENDED LINE */
DCL NO_SCREEN_UPDATE CHAR(1) STATIC INIT('N'); /* SCREEN OVERWRITE */
DCL YES_SCREEN_UPDATE CHAR(1) STATIC INIT('Y'); /* SCREEN UPDATE */
DCL NO_WARBYTE_INFO CHAR(1) STATIC INIT('N'); /* NO WAR BYTE */
DCL YES_WARBYTE_INFO CHAR(1) STATIC INIT('Y'); /* RETURN WAR BYTE */
DCL STD_AUTOTAB   CHAR(1) STATIC INIT('S'); /* STANDARD AUTOTAB */
DCL NO_AUTOTAB    CHAR(1) STATIC INIT('N'); /* NO AUTOMATIC TAB */
DCL YES_AUTOTAB   CHAR(1) STATIC INIT('Y'); /* AUTOMATIC TABUL. */

```

7 C interface



The following sections are applicable to TIAM applications only.

The definitions of the C data structure and the C data fields correspond to those of the COBOL data structure and the COBOL data fields. This chapter, therefore, deals only with details of the C language and syntax as they apply to these data structures. You will find detailed information on the data structures and data fields in the COBOL and ASSEMBLER descriptions.

Support for virtual terminals in C is provided by the data structures `cctrc.h` for the logical control characters and `vtsubc.h` for the VTSU control block.

These data structures are available as include elements in the `$TSOS.SYSLIB.VTSU-B.110` library and can be copied into your application program.

The VTSU interface for C is available as of VTSU V10.1A.

7.1 Logical control characters: cctrc.h

The cctrc.h data structure is copied into the application program by means of the following #INCLUDE statement. It contains the logical control characters for input and output in line mode, along with the printer control characters.

```
#include <cctrc.h>
```

This include member is in the library \$TSOS.SYSLIB.VTSU-B.110.

The structure of cctrc.h is shown below. For detailed information on the individual parameters, see the COBOL description as of page 125 or the ASSEMBLER description as of page 55.

```

/*****
/*      CCTRC.H          053          920908          VTSU-B          U */
/*****
/*
/*      COPYRIGHT (C) SIEMENS NIXDORF INFORMATIONSSYSTEME AG 1991      */
/*      ALL RIGHTS RESERVED                                          */
/*                                                                    */
/*****

#ifdef WAS_CCTRC
#else
#define WAS_CCTRC
/* Logical record delimiters                                     */
#define NEW_LINE          21
#define NEW_PAGE         12
#define CONT_SAME_LINE   13
#define CONT_LINE_N      41
#define CONT_COL_N       42
#define SHEET_FEED_N     33
#define CONT_ACT_POS     32

/* Logical unit delimiters                                     */
#define EMPH_LAYOUT1     29
#define EMPH_LAYOUT2     31
#define EMPH_LAYOUT3     19
#define EMPH_LAYOUT4     20
#define NORMAL_LAYOUT    30
#define DARK_LAYOUT      18
#define PART_LINE_UP     44
#define PART_LINE_DOWN   43

#define SECOND_CHAR_SET  14
#define NORMAL_CHAR_SET  15
#define SELECT_COLOUR    23

#define START_PROT_AREA  54
#define END_PROT_AREA    8
#define START_NUM_DATA   17
#define START_CHARSET_N  6

```

```

#define DIMENSION_24_80          "D24080"
#define DIMENSION_32_80          "D32080"
#define DIMENSION_43_80          "D43080"
#define DIMENSION_27_132         "D27132"

#define START_LOCAL_ATTR         9
#define END_LOCAL_ATTR           10

#define VERT_MOVE_IND            36
#define HORIZ_MOVE_IND           35
#define LEFT_MARGIN              56
#define START_PROP_TYPE          26
#define END_PROP_TYPE            27
#define MAX_LINE_LEN             51
#define MAX_LINE_NUM             53
#define START_NEAR_LETT         57
#define END_NEAR_LETT           59

/* Special functions */
#define EXTEND_LOG_CHAR          62
#define DELETE_CHAR              7
#define BACKSPACE                22
#define SUBSTITUTE               63

/* Physical unit delimiters */
#define PHYS_ESC                 39
#define PHYS_DC4                 60
#define PHYS_HT                   5
#define PHYS_VT                  11

#define TRANSPARENT_OUTPUT       'T'

#define DISPLAY_ATTRIBUTES       'I'
#define FLASHING                  1
#define UNDERSCORED              2
#define BLANKED                   4
#define REDUCED_INTENSITY         8
#define INVERSE_MODE             16
#define RESET                      0

#define FIELD_CHARACTERISTICS     'F'
#define PROTECTED_NOT_SENDABLE    1
#define PROTECTED_SENDABLE       32
#define NUMERIC                   2
#define PRE_MODIFIED              4
#define MARKABLE                  8
#define PRINTABLE                 16
#define INPUT                      0
#define AUTOSKIP                  64

#define REPEAT_SYMBOL            'R'

#endif

```

7.2 VTSU control block: vtsucb.h

The **TIAM** access method offers you a VTSUCB for C applications. In **C applications** you can include the VTSUCB in the RDATA, WROUT and WRTRD macros, for example:

```
extern cwROUT ();
cwROUT (&tiam_control_info, &output_area, &vtsu_cb);
```

This data structure has no default values, which means that all the VTSUCB fields must be initialized, with the exception of the reserved fields and the dummy fields.

The VTSU-CONTROL-BLOCK data structure is copied into the application program by means of the following macro:

```
#include <vtsucb.h>
```

This include member is in the library \$TSOS.SYSLIB.VTSU-B.110.

The structure of the VTSUCB is shown below. For detailed information on the individual parameters, see the COBOL description as of page 153 or the ASSEMBLER description as of page 89.

Data structure of the VTSUCB

```

/*****
/*      VTSUCB.H          050      921209      VTSU-B      U */
/*****
/*
/*      COPYRIGHT (C) SIEMENS NIXDORF INFORMATIONSSYSTEME AG 1992
/*      ALL RIGHTS RESERVED
/*
/*
/*****
/*
#ifdef WAS_VTSUCB
#else
#define WAS_VTSUCB

#define LINE          'L'      /* line mode          */
#define EXTEND       'E'      /* extended line mode */
#define INFO         'I'      /* info line message  */
#define PHYS         'P'      /* physical mode      */
#define TRANS        'T'      /* transparent mode    */
#define FORM         'F'      /* form mode          */
#define CHIP         'C'      /* chip card mode     */
#define MIXED        'M'      /* mixed mode         */
/* */
#define NO_HC        'N'      /* no hardcopy        */
#define YES_HC       'Y'      /* local / central hc */
#define NO_BELL      'N'      /* no bell            */
#define YES_BELL     'Y'      /* bell after output  */
#define NO_NOLOG     'N'      /* log. char. to interpret */
#define YES_NOLOG    'Y'      /* no log char to interpret */
#define NO_RIN       'N'      /* no return information */
#define YES_RIN      'Y'      /* return inform. required */

```



```

#define NO_LOC          'N'    /* no local char required      */
#define YES_LOC         'Y'    /* local charact. required     */
#define NO_OHDR        'N'    /* no output header in user    */
#define YES_OHDR       'Y'    /* output header in user msg   */
#define NO_TRANSLATION 'N'    /* no code transl. to/from ccs */
#define YES_TRANSLATION 'Y'   /* code translation required   */
#define IGN_TRANSLATION 'Y'   /* code translation default    */
#define NO_IHDR        'N'    /* no input header required    */
#define YES_IHDR       'Y'    /* input header required       */
#define NO_LOW         'N'    /* translate lower-cases       */
#define YES_LOW        'Y'    /* retain lower characters     */
#define NO_SPEC_INPUT  'N'    /* no special input            */
#define ID_CARD_READER 'I'    /* input from id_card reader   */
#define CONFIDENTIAL   'C'    /* confidential input          */
#define NO_FC          'N'    /* no function code required   */
#define YES_FC         'Y'    /* function code required      */
#define NO_HOM         'N'    /* no hom. output required     */
#define YES_HOM        'Y'    /* hom. output required        */
#define NO_NOPOS       'N'    /* output starts on next line  */
#define YES_NOPOS      'Y'    /* output starts on current line */
#define NO_CURSOR      'N'    /* cursor pos. not requested   */
#define YES_CURSOR     'Y'    /* cursor position requested   */
#define READ_MODIFIED  'M'    /* Read modified mode         */
#define READ_UNPROTECTED 'U'  /* Read unprotected mode      */
#define NO_UPDATE      'N'    /* Screen update in ext.L.M.   */
#define YES_UPDATE     'Y'    /* Screen overwrite in ext.L.M. */
#define NO_WARBYTE_INFO 'N'   /* No WAR byte to return      */
#define YES_WARBYTE_INFO 'Y'  /* Return WAR byte value      */
#define STD_AUTOTAB    'S'    /* Standard automatic tabulat. */
#define NO_AUTOTAB     'N'    /* No automatic tabulation     */
#define YES_AUTOTAB    'Y'    /* Automatic tabulation       */

typedef struct {
    short          function_unit :16 ;
    unsigned char  function_nber  : 8 ;
    unsigned char  function_vers  : 8 ;
} function_header;

typedef struct {
    unsigned char  subcode2 ;
    unsigned char  subcode1 ;
    short         maincode ;
} std_return_codes;
typedef struct {
    function_header  std_vtsuch_fct ;
    std_return_codes std_vtsuch_rc ;
} standard_header;

typedef struct {
    standard_header vtsuch_std_hdr; /* standard header          */
    short  blocklen ; /* length of VTSCUB        */
    char   input_mode ; /* input mode for mixed mode */
    char   output_mode ; /* output mode for mixed mode */
    char   filler1[4] ; /* reserved for return_info */
    char   mode ; /* mode of message          */
    char   hardcopy ; /* hardcopy function        */
    char   bell ; /* bell function            */
}

```


8 Setting operating parameters



These sections are of interest to system administrators only.

The following terminal characteristics for the TIAM access method can be defined at BS2000 system generation time. These will subsequently apply as default settings after connection setup:

- the type of screen overflow control
- the operating mode of 816x, 9749, 975x and 9763 Data Display Terminals
- the acknowledgment system to be used for outputs to the terminal.

These standard settings (defaults) can be changed dynamically for any individual connection by means of the TIAM command `MODIFY-TERMINAL-OPTIONS`.

VTSU operating parameters are also available that are of significance for CDAM, TIAM and UTM applications (see page 191).

8.1 Screen overflow control

The type of screen overflow control can be defined at BS2000 system generation time using the (class 2) system operand **TCHOFLO** (see "System Installation" manual). The value specified here can be changed using the TIAM command **MODIFY-TERMINAL-OPTIONS** (operand **OVERFLOW=CONTROL**, see TIAM User Guide).

The following values can be defined at system generation time:

- TCHOFLO=A[CK]** Overflow control by acknowledgment (default).
In the event of an overflow the system requests an acknowledgment from the terminal user with the prompt: "%PLEASE ACKNOWLEDGE". This allows the user to control the speed of computer output sequences. Any input other than ESCAPE or BREAK may be used as acknowledgment. After the receipt of an acknowledgment, computer output is resumed.
- TCHOFLO=T[IMER]** Overflow control by time delay.
When there is an overflow, the system waits for 6 seconds to give the user time to read the output data and initiate the ESCAPE/BREAK function.
- TCHOFLO=N[O]** No overflow control.
The system makes no provision for the use of the ESCAPE function in the case of long computer output nor does it prevent the premature overwriting of data on the screen.

This system parameter is interpreted when a connection is set up to \$DIALOG and causes a change call to be issued to VTSU.

8.2 Operating modes of terminals

The operating mode of 816x, 9749 and 975x Data Display Terminals can be defined at BS2000 system generation time, using the (class 2) system parameter **TCHREAD** (see "System Installation" manual). This value can be altered with the TIAM command MODIFY-TERMINAL-OPTIONS (WRITE-READ-MODE see TIAM User Guide).

The following values can be defined at system generation time:

TCHREAD=M[ODIF] Read modified fields (default).
Data is displayed on the screen in fields and input to the DP system consists of the transfer of modified fields (operating mode 1, see the "Network Access for Terminals" manual).

TCHREAD=U[NPRO] Read unprotected fields.
Data is displayed on the screen without fields and input to the DP system consists of the transfer of the screen contents from the start of screen or the cursor position (operating mode 2, see the "Network Access for Terminals" manual).

The system parameter **TCHREAD** is evaluated when a connection is set up to **\$DIALOG** and causes a change call to be issued to **VTSU**.

Note

On 815x, 8162, and 9750-62 Data Display Terminals, and on all printer terminals, the operating mode cannot be defined with the system parameter **TCHREAD**.

8.3 Acknowledgment of terminal output

The acknowledgment mechanism to be used for outputs to the terminal can be defined at BS2000 system generation time by means of the (class 2) system parameter **TCHTACK**. (See "System Installation" manual). This value can be changed with the TIAM command MODIFY-TERMINAL-OPTIONS (ACKNOWLEDGE-OUTPUT see TIAM User Guide).

The following values can be defined at system generation time:

TCHTACK=N[O] The system does not employ the internal acknowledgment mechanism when an output is performed to a terminal (default).

Note

With this setting, messages which immediately precede LOGOFF may be lost when the connection is cleared down.

TCHTACK=Y[ES] The system requests internal acknowledgments whenever there is an output to a terminal.

This system parameter is evaluated when a connection is set up to \$DIALOG.

8.4 Setting operating parameters during dynamic VTSU loading

These operating parameters are located in an SAM file called SYSPAR.VTSU-B.xxx (xxx=version number) which must be installed under \$TSOS by the system administrator. There is a default setting for all operating parameters. This can be changed by the system administrator to meet the user's requirements. Changes become effective only after VTSU is reloaded. All settings then apply system-wide until VTSU is unloaded or until SHUTDOWN.

The parameters UTM-PERM8, DCAM-PERM8 and TIAM-PERM8 are new; they have been introduced by way of providing 8-bit support. These parameters are accepted as of BS2000 V10.0A, but are effective only in BS2000/OSD-BC V1.0 or higher.

This chapter also discusses the PDN free-text parameter that VTSU uses to define the type of connection and the printer type.

Structure of the operating parameter file SYSPAR.VTSU-B.110

Beginning of file	/ BEGIN VTSU-PARAMS
Parameter line	param=value
or	
Comment line	* Comment
End of file	/ END-PARAMS

The text must begin in column 1 on all lines.

Any contents violating this format will be rejected and an error message will be sent to the console.

Individual parameters may be specified in any sequence. Likewise, any number of comment lines may be inserted between beginning of file and end of file. Comment lines are identified by '*' in column 1.

Parameter description

TERMROLL

- =Y Each output with MODE=LINE to a 975x, 976x or 816x Terminal starts in the final screen line. Prior to this, the old screen contents are rolled up. It is not possible to work with screen formats (control character DIM in VTCSET).
- =N Each new output causes the old screen contents to be cleared from the current cursor position. The output always begins in the line following the cursor; if the cursor is in the final line, the screen contents are rolled up first.

T8151ETB	
=Y	Messages on 8151 Terminals are terminated with ETB.
=N	Messages on 8151 Terminals are terminated with ETX.
CODETR	
=Y	All input message texts are converted into a special EBCDI code. All output message texts are converted from a special EBCDI code into the standardized EBCDI code. The user decides which special EBCDI code table is to be used. This table must be entered separately in the VTSU code.
=N	No special code conversion is carried out.
LINEND	
=char	The character entered for "char" is used as the logical end-of-line character for 8110 and 3270 terminals. (Default='/' or 'ö')
INFOHOLD	
=Y	The system line remains intact when there is a mode change (e.g. from MODE=PHYS to MODE=LINE).
=N	The system line is reset when a mode change takes place.
TSILENT	
=Y	Connection of SILENT terminals possible.
=N	Connection of SILENT terminals not possible.
EXPROPOS	
=Y	Positioning to protected fields is possible in extended line mode. (A complete hardcopy screen dump can then be obtained by positioning at beginning of screen and pressing the LA1 key.)
=N	Positioning to protected fields is not possible in extended line mode.
T8152ETX	
=Y	All messages sent with MODE=COMP to an 8152 Terminal terminated with ETX.
=N	All messages to an 8152 Terminal are terminated with ETB.

STRCTDIA
=Y Further keyboard inputs are possible following input from a data display terminal (no dialog lock).
=N The dialog lock is set following each input in strict dialog. After this only inputs from K keys are accepted. The dialog lock remains in force until after the next output.

SICAD
=Y An 8152 Terminal is operated like a 9731 Graphics Workstation.
=N Normal support for 8152 Terminal.

As regards defining the connection type (xxxxxxxLIN8) and the printer type (xxxxxxxDEV8), see the notes on PDN free-text parameters at the end of this chapter, page 197.

9001-3DEV8
=Y The 9001-32 Printer is an 8-bit printer.
=N The 9001-32 Printer is a 7-bit printer.

9001-3LIN8
=Y The 9001-32 Printer is connected to the host via an 8-bit line.
=N The 9001-32 Printer is connected to the host via a 7-bit line.

9001-893DEV8
=Y The 9001-832 Printer is an 8-bit printer.
=N The 9001-832 Printer is a 7-bit printer.

9001-893LIN8
=Y The 9001-832 Printer is connected to the host via an 8-bit line.
=N The 9001-832 Printer is connected to the host via a 7-bit line.

9011-18DEV8
=Y The 9011-28 Printer is an 8-bit printer.
=N The 9011-28 Printer is a 7-bit printer.

9011-18LIN8
=Y The 9011-28 Printer is connected to the host via an 8-bit line.
=N The 9011-28 Printer is connected to the host via a 7-bit line.

9011-19DEV8	
=Y	The 9011-29 Printer is an 8-bit printer.
=N	The 9011-29 Printer is a 7-bit printer.
9011-19LIN8	
=Y	The 9011-29 Printer is connected to the host via an 8-bit line.
=N	The 9011-29 Printer is connected to the host via a 7-bit line.
9012DEV8	
=Y	The 9012 Printer is an 8-bit printer.
=N	The 9012 Printer is a 7-bit printer.
9012LIN8	
=Y	The 9012 Printer is connected to the host via an 8-bit line.
=N	The 9012 Printer is connected to the host via a 7-bit line.
9013DEV8	
=Y	The 9013 Printer is an 8-bit printer.
=N	The 9013 Printer is a 7-bit printer.
9013LIN8	
=Y	The 9013 Printer is connected to the host via an 8-bit line.
=N	The 9013 Printer is connected to the host via a 7-bit line.
9014DEV8	
=Y	The 9014 Printer is an 8-bit printer.
=N	The 9014 Printer is a 7-bit printer.
9014LIN8	
=Y	The 9014 Printer is connected to the host via an 8-bit line.
=N	The 9014 Printer is connected to the host via a 7-bit line.
9021DEV8	
=Y	The 9021 Printer is an 8-bit printer.
=N	The 9021 Printer is a 7-bit printer.
9021LIN8	
=Y	The 9021 Printer is connected to the host via an 8-bit line.
=N	The 9021 Printer is connected to the host via a 7-bit line.

COMPKEYS

=Y Assign the PF keys (6-24) of a 3270 terminal to the corresponding F keys (6-24) of a 9750 DDT.

=N Do not assign the PF keys of a 3270 DDT to the corresponding F keys of a 9750 DDT.

Compkey=N			Compkey=Y		
CODE	3270	9750	CODE	3270	9750
03	PF6	K3	26	PF6	F6
04	PF7	K4	27	PF7	F7
05	PF8	K5	28	PF8	F8
06	PF9	K6	29	PF9	F9
07	PF10	K7	2A	PF10	F10
08	PF11	K8	2B	PF11	F11
09	PF12	K9	2C	PF12	F12
0A	PF13	K10	2D	PF13	F13
0B	PF14	K11	2E	PF14	F14
0C	PF15	K12	2F	PF15	F15
0D	PF16	K13	46	PF16	F16
0E	PF17	K14	47	PF17	F17
26	PF18	F6	48	PF18	F18
27	PF19	F7	49	PF19	F19
28	PF20	F8	4A	PF20	F20
29	PF21	F9	4B	PF21	F21
2A	PF22	F10	4C	PF22	F22
2B	PF23	F11	4D	PF23	F23
2C	PF24	F12	4E	PF24	F24

DARPRINTABLE

=Y The text characters subsequent to the display control character DAR can be printed.

=N The text characters subsequent to the display control character DAR cannot be printed (as of VTSU V9.0B).

If DARPRINTABLE=N, all other logical control characters, including SPA and NUM, are reset by DAR. It is not possible to combine DAR with other logical control characters.

UTM-PERM8

=Y The application is set to an 8-bit mode. This parameter is effective only on 8-bit terminals, if XHCS has been activated and if a valid extended user standard code has been assigned to the user ID of this application (in the JOIN file). The application, moreover, must be executable in an 8-bit environment.

Note that 8-bit characters entered in 7-bit formats are either ignored or rejected by FHS (validity check for input characters). The reaction depends on the FHS version in use.

=N The application is set to a 7-bit mode.

DCAM-PERM8

=Y

The application is set to an 8-bit mode.

This parameter is effective only on 8-bit terminals, if XHCS has been activated and if a valid extended user standard code has been assigned to the user ID of this application (in the JOIN file). The application, moreover, must be executable in an 8-bit environment.

=N

The application is set to a 7-bit mode.

TIAM-PERM8

=Y

The application is set to an 8-bit mode.

This parameter is effective only on 8-bit terminals, if XHCS has been activated and if a valid extended user standard code has been assigned to the user ID of this application (in the JOIN file). The application, moreover, must be executable in an 8-bit environment.

This operating parameter has the same effect as the command
MODIFY-TERMINAL-OPTION CODED-CHARACTER-SET=8-BIT-DEFAULT.

=N

The application is set to a 7-bit mode.

Printer definition by means of the PDN free-text parameter



This section is of interest to network administrators only

As of V11, VTSU uses the new PDN free-text parameter (PDN V11) to define type of connection and printer type and to define the 8-bit variants to be supported for the printers concerned.

This new free-text parameter has higher priority than the corresponding VTSU operating parameters. This means, therefore, that if both the free-text parameter and the corresponding VTSU operating parameters are specified the free-text parameter is used to define the connection type and the type of printer. In the absence of the free-text parameter, the VTSU operating parameters are used to define connection type and type of printer.

Call:

XOPCH,FREITXT=DxxVzzzz

xx = 77	7-bit printer
87	8-bit printer with 7-bit connection
88	8-bit printer with 8-bit connection

zzzz = List of 8-bit ISO variants supported
e.g. 1200 for printers that support ISO8859-1 and ISO8859-2

Note the following:

- The parameter must always be 8 bytes in length. Consequently, you must pad the list of supported variants with zeros as otherwise VTSU would ignore the parameter.
- Unknown entries for xx are interpreted as 77 and no warning is issued.
- If a string of zeros is entered for zzzz, the printer supports all ISO variants. In this event, VTSU does not check whether printer status has been set correctly. You must, therefore, set the printer status correctly before each 8-bit output.

The parameters that have to be set for the various combinations of printers and connection modes are listed in the table below.

8-bit printer	PDN values	Type of connection	Operating parameters of the printers	Operating parameters of the line
9001-32	X'65'	BAM	9001-3DEV8=Y	9001-3LIN8=N
		HDLC/AFP	9001-3DEV8=Y	9001-3LIN8=N
		HC 7-bit data display terminal with BAM connection	9001-3DEV8=Y	9001-3LIN8=N
		HC 8-bit data display terminal with BAM connection	9001-3DEV8=Y	9001-3LIN8=N
		HC 8-bit data display terminal with HDLC connection	9001-3DEV8=Y	9001-3LIN8=N
9001-832	X'66'	BAM	9001-893DEV8=Y	9001-893LIN8=N
		HDLC/AFP	9001-893DEV8=Y	9001-893LIN8=N
		HC 7-bit data display terminal with BAM connection	9001-893DEV8=Y	9001-893LIN8=N
		HC 8-bit data display terminal with BAM connection	9001-893DEV8=Y	9001-893LIN8=N
		HC 8-bit data display terminal with HDLC connection	9001-893DEV8=Y	9001-893LIN8=N
9011-28	X'6B'	BAM	9011-18DEV8=Y	9011-18LIN8=N
		HDLC/AFP	9011-18DEV8=Y	9011-18LIN8=N
		HC 7-bit data display terminal with BAM connection	9011-18DEV8=Y	9011-18LIN8=N
		HC 8-bit data display terminal with BAM connection	9011-18DEV8=Y	9011-18LIN8=N
		HC 8-bit data display terminal with HDLC connection	9011-18DEV8=Y	9011-18LIN8=N

8-bit printer	PDN values	Type of connection	Operating parameters of the printers	Operating parameters of the line
9011-29	X'6C'	BAM	9011-19DEV8=Y	9011-19LIN8=N
		HDLC/AFP	9011-19DEV8=Y	9011-19LIN8=N
		HC 7-bit data display terminal with BAM connection	9011-19DEV8=Y	9011-19LIN8=N
		HC 8-bit data display terminal with BAM connection	9011-19DEV8=Y	9011-19LIN8=N
		HC 8-bit data display terminal with HDLC connection	9011-19DEV8=Y	9011-19LIN8=N
9012	X'5B'	BAM	9012DEV8=Y	9012LIN8=N
		HDLC/AFP	9012DEV8=Y	9012LIN8=N
		HC 7-bit data display terminal with BAM connection	9012DEV8=Y	9012LIN8=N
		HC 8-bit data display terminal with BAM connection	9012DEV8=Y	9012LIN8=N
		HC 8-bit data display terminal with HDLC connection	9012DEV8=Y	9012LIN8=N
9013	X'5C'	BAM	9013DEV8=Y	9013LIN8=N
		HDLC/AFP	9013DEV8=Y	9013LIN8=N
		HC 7-bit data display terminal with BAM connection	9013DEV8=Y	9013LIN8=N
		HC 8-bit data display terminal with BAM connection	9013DEV8=Y	9013LIN8=N
		HC 8-bit data display terminal with HDLC connection	9013DEV8=Y	9013LIN8=N

8-bit printer	PDN values	Type of connection	Operating parameters of the printers	Operating parameters of the line
9014	X'72'	BAM	9014DEV8=Y	9014LIN8=N
		HDLC/AFP	9014DEV8=Y	9014LIN8=N
		HC 7-bit data display terminal with BAM connection	9014DEV8=Y	9014LIN8=N
		HC 8-bit data display terminal with BAM connection	9014DEV8=Y	9014LIN8=N
		HC 8-bit data display terminal with HDLC connection	9014DEV8=Y	9014LIN8=N
9021 (PCL)	X'70'	BAM	9021DEV8=Y	9021LIN8=N
		HDLC/AFP	9021DEV8=Y	9021LIN8=Y
		HC 8-bit data display terminal with BAM connection	9021DEV8=Y	9021LIN8=N
		HC 8-bit data display terminal with HDLC connection	9021DEV8=Y	9021LIN8=Y

The 9001-32, 9001-832, 9011-x, 9012, 9013 and 9014 Printers are 8-bit printers. Nevertheless, they require the control character SO in order to access the high-value part of the character set. Consequently, the value of operating parameter xxxxxxxxLIN8 (xxxxxxx=PDN device name) must be 'N', even if the printer is connected via an 8-bit line.

9 VTSU trace



This chapter is of interest to system administrators only.

VTSU trace creates a log of messages sent to data terminals. Input and output messages can be logged in their logical and physical forms. VTSU trace logs **all** input/output messages and thus represents a record of security-relevant information. Consequently, only the system administrator can use VTSU trace.

VTSU trace is loaded as a DSSM subsystem. The DSSM statements supplied for the VTSUTRAC subsystem must be incorporated in the catalog. The files supplied are as follows:

```
SYSLNK.VTSUTRAC.110 link library for DSSM
SYSPRC.VTSUTRAC.110 procedure for starting a trace
                    (includes /CREATE-SUBSYSTEM)
```

When a message is logged, the following information is written into the SERSLOG file:

- current error log sequence number
- calling address
- ITN and TSN of the sender
- time of the entry
- RECTYPE of the message
- message

It would be impractical to log all messages, so a number of selection criteria can be applied.

These criteria are:

- input/output messages
- a certain device type
- one or more TSNs
- all data terminals for which the MODIFY-TERMINAL-OPTIONS command was used to set the characteristics APL and GRAPH
- system or user messages
- station level parameter block (SPB). The SPB contains the parameters of the VTSU interface ISLP and the areas that VTSU uses to store information on the corresponding connection.

The procedure must be started under \$TSOS, and the EDT must also be available under \$TSOS. The trace is evaluated with the aid of the ELFE tool.

The procedure is called under \$TSOS as follows:

```
/CALL-PROC SYSPRC.VTSUTRAC.xxx      (xxx=VTSU version)
```

The parameters of the procedure

Parameter	Input	Meaning
HELP	Y/N	Help requested, view procedure outline (see page 203)
INPUTMSG	Y/N	Trace - input messages
OUTPUTMSG	Y/N	- output messages
USER MSG	Y/N	- user messages
SYMSG	Y/N	- system messages
SPB	Y/N	- SPB
SPCTSN	Y/N	Select group of TSNs (1-10)
T1-T10	tsn	Specific TSN
TNUMBER	1..10	Number of TSNs
SPCDVCTYPE	Y/N	Select specific device type
DVCTYPE	00..FF	Device type as per device table (hexadecimal)
SONDTRACE	Y/N	Additional information for internationalized spell111 trace routine

If you specify SPCTSN=N and SPCDVCTYPE=N, the trace includes only messages for terminals that have issued the following command:

```
/MODIFY-TERMINAL-OPTIONS APL-CHARACTER-SET=YES,GRAPHICS=YES
```

Make your selections and start the VTSUTRAC subsystem, thus transferring your selection criteria to the system. If you want to change your selection you must stop the subsystem and repeat the call.

The command for stopping the subsystem is:

```
/STOP-SUBSYSTEM VTSUTRAC
```

Once the subsystem has been stopped, you must close the current SERSLOG file. The command for closing the SERSLOG file is:

```
/CHANGE-SERSLOG
```

Once it has been closed, the SERSLOG can be evaluated with ELFE. See the BS2000 Ready Reference for a description of ELFE.

DCA is the record type of the messages that VTSUTRAC enters in SERSLOG.

Example of the procedure outline:

```
*****
*   PROCEDURE PURPOSE : CREATE SUBSYSTEM VTSUTRAC WITH SOME PARAMETERS
*
*   THESE PARAMETERS DETERMINE THE DIFFERENT TASKS TO BE MADE BY THESE
*   SUBSYSTEMS :
*       WRITING - INPUT/OUTPUT MESSAGES
*               - PHYSICAL (SYSTEM)/USER MESSAGES
*               - STATION LEVEL PROCESSING BLOCK (SPB)
*               - FOR UP TO 10 DIFFERENT TASK SEQUENCE NUMBERS (TSN)
*               - FOR EACH TSN
*               - FROM ONE SPECIAL VIRTUAL DEVICETYPE (1 BYTE)
*               - FROM ALL DEVICETYPES
*               - FROM DEVICES WITH TCHAR=(APL,GRAPH)
*
*       IN A SERSLOG FILE.
*       AND ACTIVATING THE TRACE FROM SPECIAL ROUTINES
*
*   VTSUTRAC WRITE MESSAGES THAT ENTER/LEAVE VTSU
*   NOTE : IF YOU WANT NEITHER A SPECIAL DEVICETYPE NOR A SPECIAL TSN
*           ONLY MESSAGES FROM DEVICES WITH TCHAR=(APL,GRAPH) ARE WRITTEN
*           STOP VTSUTRACE BY "STOP SUBSYSTEM VTSUTRAC"
*   THE MESSAGES WILL BE WRITTEN IN THE SERSLOG-FILE UNTIL THE SUBSYSTEM
*   IS DELETED.
*   WITH 'ELFE' ONE CAN WATCH THE CONTENT OF THE SERSLOG-FILE. (AFTER IT
*   IS CLOSED).
*   HOW TO USE E L F E IS DESCRIBED IN 'BS2000 Diagnosis Manual'
*****
```


10 Special characteristics of 3270 terminals

VTSU supports a mode in which IBM 3270 terminals can be operated like 9750 Data Display Terminals. The differences between the 3270 and the 9750 DDT are summarized below.

1. There is no roll-up mode. Once the last line on the screen has been reached, output continues in the first line.
2. When an output has terminated, the remaining information on the screen is not deleted; only the remainder of the current line and the entire following line are deleted (if output terminates in the last line on the screen, the following line is then the first on the screen). The display control character NOR/NORMAL-LAYOUT is inserted at the start of the next line but one.
3. Use of logical control characters:
 - Each logical control character requires one character position on the screen. The control characters NL/NEW-LINE and VPA/CONT-LINE-N require two positions (one at the end of the line in which they are specified and one in the line in which output is to continue). Two or more successive control characters are merged into one character on the screen.

Each logical control character causes a new field to be created. Thus, when logical control characters EM1/EMPH-LAYOUT1 through EM4/EMPH-LAYOUT4, DAR/DARK-LAYOUT and NOR/NORMAL-LAYOUT (these do not cause a new field to be created with TRANSDATA terminals) are used with EXTEND=NO, an output reentered after it has been modified may be shorter than when TRANSDATA terminals are used.

- With the logical control character NUM/START-NUM-DATA, other specifications than those for TRANSDATA terminals are possible.
- When the logical control character HPA/CONT-COL-N is used, the cursor is positioned at the start of the next unprotected field.

4. Effect of the EDIT options and the VTSUCB parameters:

HCOPY/HARDCOPY

The entire screen is always printed out, which means that some inputs and outputs may be printed out more than once. Outputs for which no printout is required but which are on the screen at the time the hardcopy output is requested are also printed out.

EXTEND/MODUS="E"

Only limited use can be made of null characters in input as they are not transferred to the computer. VTSU, however, adds null characters to all fields that are returned shortened, so that the user receives the length expected. In the event of individual fields not returning at all (e.g. because the "ERASE INPUT" key was pressed), VTSU supplies a return code.

GETFC/FUNCTION-CODE

For the mapping of 3270 function key codes, see the table on page 251.

IHDR/INPUT-HEADER

If the entire message header is sent to the application program, the message prefix contains the application identifier (AID) and the cursor position, which is 2 bytes long.

SPECIN/SPECIAL-INPUT

At 3270 display terminals, data can be input from a defined ID card reader at any time. If input from an ID card reader is requested, any other input is converted to K14.

Unlike TRANSDATA terminals, at 3270 display terminals, data can be input from a defined ID card reader at any time. If input from an ID card reader is requested, any other input is converted to K14.

AUTOTAB/AUTOMATIC-TABULATION

If automatic tabulation is defined, it is always possible to move the cursor to protected fields by means of the arrow keys. If, however, AUTOTAB=YES, the cursor automatically moves from one input field to the next as soon as a character is entered at the end of the field.

The EXPROPOS operating parameter is ignored by 3270 terminals.

5. In the READ=MODIFIED mode, the virtual positions of the modified fields are returned in incremental order. The fields starting at line 1 column 1 are an exception. The logical position of each of these fields is returned as the last logical position of the screen, because the ST (start of field) control character for each of these fields is located on the last line and in the last column of the preceding screen.
6. If the positioning function (HPA/VPA) is in use, data output commences at the subsequent position, because the ST (start of field) control character is displayed as a blank. If UPDATE=YES is specified in extended line mode, the user must increment the basic position by one in order to update a field, because otherwise the SF control character would be overwritten and the field suppressed.

11 Support for special data terminals



This chapter is of interest to system administrators only.

Arabic/Persian 8-bit data display terminals, Euro 7-bit data display terminals and Escape printers are special data display terminals and printers that must be supported by special routines. VTSU calls these routines internally. You can use Arabic/Persian 8-bit display terminals and Euro 7-bit display terminals only if you have specified support for these terminals in the installation procedure (see page 210).

You must specify the Euro 7-bit terminals and Escape printers for which support is required in the configuration file (see page 211).

Note that specifications and modifications incorporated in the installation procedure or the configuration file are not effective until the VTSU subsystem is restarted.

Arabic/Persian 8-bit data display terminals are terminals that support the Arabic languages Arabic/Latin (ARA), Persian/Latin (Farsi, FAR) and Arabic/French (North African, NAF) in 8-bit code. You can use Arabic/Persian 8-bit display terminals only if you have specified support for these terminals in the installation procedure (see page 210).

As for other 8-bit data display terminals, you must define the appropriate tables in XHCS and the user must note the associated characteristics, for example of the user standard code or the VTSUCB.

Euro 7-bit data display terminals are terminals that support the languages Swedish, Hungarian, Cyrillic and Greek in 7-bit code. You can use Euro 7-bit display terminals only if you have specified support for these terminals in the installation procedure (see page 210).

You must also enter the station name, processor name and variant for each data display terminal in the configuration file (see page 211).

Escape printers differ from standard printers in that they can use special Escape sequences to switch between the right-hand and left-hand sides of an 8-bit code table. Standard printers employ the control characters SI and SO for this purpose. You must specify the station name, processor name and variant for each Escape printer in the configuration file (see page 211).

11.1 Installation procedure

The installation procedure allows you to enable support for Arabic/Persian 8-bit data display terminals and/or Euro 7-bit data display terminals. You cannot use these terminals until the installation procedure has been run. It is not necessary to call the installation procedure in order to implement support for Escape printers.

After running the installation procedure, you must restart the VTSU subsystem.

The name of the installation procedure is `SYSPRC.VTSU-B.110.INSTALL`.

The installation procedure is called as follows:

```
/CALL-PROCEDURE SYSPRC.VTSU-B.110.INSTALL
```

When you call the VTSU installation procedure, you will be asked to specify the output language for the procedure's message texts.

The language you select is then used for the prompts asking you to define the terminals to be supported. In English, for example:

```
NATIONAL 7-BIT SUPPORT (Y/N) ? :  
ARA/FAR/NAF SUPPORT (Y/N) ? :
```

The only valid inputs are Y and N. Any other input is interpreted as N, the default.

11.2 Configuration file

You must define all Euro 7-bit data display terminals and Escape printers for which support is required in the configuration file.

You must create the configuration file yourself, as a SAM file having the following name:

```
SYSPAR.VTSU-B.110.CONFIG
```

The format shown below is mandatory for the configuration file. One terminal is defined per line. Each definition must begin in column 1.

Format of the configuration file:

```
1st      10th      19th column
+-----+-----+-----
```

```
xxxxxxxxx yyyyyyyy zzz
```

```
xxxxxxxxx is a station name, must always be eight characters long
yyyyyyyyy is a processor name, must always be eight characters long
zzz       is a variant, must always be three characters long
          The options for zz are as follows:
```

- CYR for Cyrillic 7-bit
- SWE for Swedish 7-bit
- HUN for Hungarian 7-bit
- GRE for Greek 7-bit
- ESC for the ESC printer

The separator between station name, processor name and variant must always be one blank.

You can use '?' as a wild card in the station name and processor name. Bear in mind, however, that both station name and processor name must be exactly eight characters long. Names that are shorter than eight characters must be padded to the right length with blanks.

If you make any entries in or changes to the configuration file, you must restart the VTSU subsystem for the changes to take effect.

Examples of definition lines

DSN31101 D241KR31 SWE

The data display terminal having the station name 'DSN31101' and the processor name 'D241KR31' supports Swedish in the 7-bit code (SWE).

PRN31102 D241KR31 ESC

The printer having the station name PRN31102 and the processor name D241KR31 is an Escape printer.

DSN31??? D241KR31 CYR

All display terminals having station names beginning with 'DSN31' and the processor name 'D241KR31' support Cyrillic in 7-bit code (CYR).

12 Type 9750 Data Display Terminals

Type 9750 DDTs are data display terminals having an active screen area of 1920 characters in 24 lines by 80 columns. The terminals of this type are the 974x, 975x and 9763 DDTs. The properties common to all these terminals are described below. Special functions peculiar to particular terminals are described in the manuals supplied with the terminals.

Operating mode

There are two operating modes for the exchange of messages between the BS2000 host and the terminals:

1. Field-oriented display from top to bottom with automatic roll-up, input of modified fields
(possible in timesharing and inquiry-and-transaction modes)
2. Display without fields in top-down mode with two input options:
 - input from start of screen
 - input from cursor positionThis mode is possible only in the timesharing mode and is defined in the MODIFY-TERMINAL-OPTIONS command (see TIAM User Guide).

Mode 1 (field-oriented)

Field-oriented display

A data field is created on the screen for each incoming or outgoing message. The appearance of a particular field depends on the type of message:

Type of message	Ident-ifier	Font	Bright-ness	Modifi-able
Incoming message from DCAM applications (inq. and trans.) or user programs		normal	low inten.	yes
Incoming messages from operating system	%			no
Input request from user	*			
Input request from operating system	/			
Field for outgoing messages	zeros		high int.	

It is easy to find the invisible boundary between two fields by pressing "Right tab (→)" or "Left tab (←)". The cursor moves to the subsequent or preceding start of field, as applicable.

Top-down mode with automatic roll-up

Messages output by the BS2000 host to the terminal are written into the line after the current cursor position, and the contents of the screen deleted as of the cursor position. If the output is longer than one screen line it is continued into the next line. If the insertion point reaches the end of the input field on the screen, the roll-up mode is automatically activated. In this mode, the contents of the screen roll up one line and the output from the BS2000 host is written into the bottom line. The contents of the first line are lost and, if this line contained the start of a multi-line data field, the field attributes, too, are lost. The text in the remainder of the field is displayed at high intensity and can no longer be modified and thus transmitted.

Modifying fields

Modifying means writing, inserting, deleting or removing characters in modifiable fields. The most common for of modification is writing a new text in an input field. If the text reaches the end of the field, the contents of the screen automatically roll up one line and the input can be continued in the bottom line.

It is also possible to modify incoming texts and texts issued beforehand and send these modified texts to the BS2000 host. The symbols for the input prompt (*, /) are protected and cannot be modified. If you attempt to write in a protected field the keyboard is locked and 'ERR' appears on the screen. You may also hear an acoustic warning.

The keyboard is unlocked when the cursor is moved to an unprotected field. If you inadvertently modify a field and want to prevent its transfer, you can reset the modifications for all fields by pressing ESC and then the 'm' key. You must then remodify the fields you want to transfer to the host.

Transfer to the BS2000 host

Press the SEND key (DUE1) to send to the BS2000 host the contents of all fields that are between the start of the screen and an end marker and which were modified since the last transfer. If there is no end marker all modified fields are transferred. The position of the cursor has no effect on the transfer. Once transfer is completed, the cursor is behind the end marker or at the start of the screen. This permits you to influence the way in which incoming messages are displayed. If you have set an end marker, output starts in the line below the end marker and automatic roll-up is activated if necessary. If no end marker has been set the contents of the screen are deleted and the incoming messages are written from the start of the screen down.

Mode 2 (without fields)

Display without fields

All incoming characters and the characters you input are displayed at high intensity and you can modify them. The arrow keys move the cursor forward or backward to positions 1, 17, 33, 49, and 65 in the current screen line.

Top-down output

Messages from the BS2000 host are written into the screen line after the current cursor position. If the message is longer than one screen line it is continued into the next line. When the last screen line is filled output continues in the first line, overwriting the original contents. An input with input version 1 clears the screen and the incoming message is displayed as of the start of the screen.

Transfer to the BS2000 host

There are two input variants:

1. Input from start of screen (default) If the BS2000 host expects an entry from the data display terminal, the cursor is positioned at the start of the screen. You can now enter your text or modify the texts sent by the BS2000 host. The entire contents of the screen are sent when you press DUE1. If you want to send only a part of the screen, you must set an end marker at the appropriate point. The position of the cursor has no effect on transfer and changes as transfer progresses. When transfer is completed, the cursor is positioned after the end marker or at the start of the screen, as applicable.
2. Input from cursor position If the BS2000 host expects an entry, the cursor is positioned at the next start of line. You can now write text entries or modify existing texts anywhere on the screen. The input text is completed by an end marker and the cursor positioned at the first character of the text to be transferred. When you press DUE1, the contents of the screen from the cursor position to the end marker are transferred to the host. When transfer is complete, the cursor is positioned after the end marker or at the start of screen, as applicable. The next incoming message clears the line from the cursor position to the end of line and the message is displayed from the start of the next line.

Changing from one variant to the other

You can toggle from one input variant to the other by pressing F2 (data transfer marked) instead of DUE1 to transfer the data.

The position of the cursor indicates which variant is selected: if input variant 1 is active, the cursor is at the start of screen. If input variant 2 is selected, the cursor is at the start of line.

Additional data-transfer option

In both modes, you have the option of transferring texts from a start marker down to the cursor. This option transfers all characters after the start marker down to and including the character on which the cursor is positioned. Press the AM key to create the start marker. If there are more than one start markers on the screen, all characters from the marker closest to the cursor are transferred. If the screen does not contain a start marker, transfer is from the start of screen. Press DUE2 to initiate data transfer.

Inputs from ID card readers

The BS2000 host can also request inputs from ID card readers. A high-intensity field is created for the purpose on the screen. The keyboard is locked, the only keys still active being SHIFT, LOCK and the short telegram function K14 (ESC and ':' keys). You can now insert your ID card in the reader. If your ID card is accepted, the input field is blanked, the authentication information is transferred to the BS2000 host and the input field is deleted. The next output starts on a new screen. If the ID card reader discovers an error an acoustic alarm may sound and either ID CARD ERROR or CARD BLANK appears in the input field. You are being prompted in this way to insert a valid ID card. You can cancel the prompt to insert an ID card with short telegram K14 (press ESC and ':'). When you remove a valid ID card from the reader the data display terminal automatically sends short telegram K14 to the BS2000 host. If you insert an ID card in the reader without being prompted to do so, 'LOCKED' appears in the message line and data transfer is not initiated. Press the RS key to reset and cancel the message.

Insert, remove, delete

Insert character

Press the EFG key to insert characters in existing text (EFG light up). In the insert mode the character beneath the cursor and the text to the right is shifted one place to the right as far as a null character or a field boundary. Null characters are overwritten, characters shifted past a field boundary are lost. To reset the insert mode press RS. The mode is also reset when data is transferred by the BS2000 host (EFG goes out).

Insert line

The contents of the line in which the cursor is located and the lines beneath it are moved down by one line. The contents of the last screen line are copied into the cursor line.

Remove character

Press AFG to remove the character beneath the cursor from the text. Text to the right of the cursor is shifted one place to the left and the text string shifted in this way is padded with a null.

Remove line

The contents of the line below the line in which the cursor is located are copied into the next line up. The contents of the cursor line are copied into the last line.

Roll-up

The contents of the screen are rolled up one line. The contents of the first screen line are lost. The cursor is positioned at the start of the blank line at the bottom of the screen.

Clear screen buffer

The contents of the screen buffer are deleted. The cursor is positioned at the start of screen.

Delete line or field

From the cursor position onward, the rest of the line (operating mode 2) or the rest of the field (operating mode 1) is overwritten with nulls. If the text to be deleted contains an end marker, only the text up to the end marker is deleted, along with the marker itself.

Data transfer functions

DÜ1	Initiate data transfer defined by the BS2000 host
DÜ2	Initiate data transfer from start marker to cursor
F2	Initiate data transfer defined by the BS2000 host and change input variant in operating mode 2.
K2	Change from program mode to system mode (timesharing)

13 PLUS

With the P key load-and-save program PLUS the terminal user can:

- define the assignment of the P keys interactively,
- save P key formats in libraries,
- load P key formats into the P keys of the display terminal (for valid terminals see page 221).

Stored P key formats can also be loaded into P keys by means of an application program (see page 237), generally with the aid of FHS (Format Handling System).

PLUS offers the following functions:

- definition of P statements in symbolic form with automatic chaining of free P keys (for restrictions see page 226, P key memory);
- definition of an output message;
- documentation of the P/MSG statements in symbolic form on the terminal and/or printer;
- saving the P key formats in a BS2000 library.
- loading defined or saved P key formats at the terminal.

PLUS can run in both interactive and batch mode. The P keys of a display terminal, however, can only be loaded in **interactive mode**. All the other functions are also possible in batch mode.

13.1 Control statements

Definition of statements

Operation	Operands	Meaning
TY55	[w]	Definition of the 512-byte P key memory
Pi_	$\left\{ \begin{array}{l} \left\{ \begin{array}{l} \text{'text' } \\ \text{cmd[*n]} \\ \text{poscmd[:l/c]} \end{array} \right\} [, \dots] [, Pj] \\ * \\ - \\ ? \end{array} \right\}$	Definition of P key i } Key contents } Keep key free } Clear key contents } Query key contents
MSG_	$\left\{ \begin{array}{l} \left\{ \begin{array}{l} \text{'text' } \\ \text{cmd[*n]} \\ \text{poscmd[:l/c]} \end{array} \right\} [, \dots] \\ - \\ ? \end{array} \right\}$	Definition of an output message } Output message } Delete output message } Query output message
DEL		Clear all P key contents and/or delete output message
PUT		Load contents of P keys and/or output message at the terminal
SAV_	mapname[,libname] { [, *PLAM] } { [, *LMR] }	Save P key formats and/or output message in a BS2000 library
GET_	mapname[,libname]	Load P key formats and/or output message from a BS2000 library
DOC_	$\left\{ \begin{array}{l} \text{N[ONE]} \\ \text{O[UT]} \\ \text{L[ST]} \\ \text{B[OTH]} \end{array} \right\}$	Definition of the documentation medium } No documentation } Documentation on SYSOUT } Documentation on SYSLST } Documentation on SYSOUT and SYSLST
SYS		Switch to system mode
END		End command; ends processing of all P key formats and/or output messages not yet fully processed.
STOP		Quit the PLUS program

Input format

Each control statement consists of an operation code and possibly operands (upper case and/or lowercase). The operation code and operands must be separated by at least one blank, and operands by commas. The end of the operands is identified by the end of the input record or a blank. Subsequent characters are interpreted as comments and are not processed.

The maximum length for the input record must not be exceeded, otherwise the record will be truncated at this point and the error message "*** INPUT TOO LONG: TRUNCATED" is issued.

The length of the control statement depends on the size of the P key memory in the terminal. The following table shows the maximum possible record length for the various display terminals.

Record length	Terminal												
	8160	8162	9748	9749	9750	9752	9753	9754	9755	9758	9759	9762C	9763
256 bytes	x	x	x	x	x	x	x	x	-	-	-	-	-
512 bytes	-	-	-	-	-	-	-	-	x	x	x	x	x

Processing of control statements

The operation code and the operands are validated from left to right. Further processing of the control statements depends on the statements concerned and is therefore described under the individual statements.

The following table gives an overview of which functions the various statements affect.

Control statement	Processed by			Overwrite (optional)	BS2000 library	Error message on SYSOUT (error location flashes)	Error-free device code (internal only)	
	GET	SAV	PUT END				Buffer	Process
P			x	x		x	x	
MSG			x	x		x	x	
DEL			x	x				
DOC	x		x					
SAV		x			x	x		
GET	x				x	x		
PUT			x					x
SYS								
END			x					x
STOP								

Description of the control statements

Explanation of the following examples:

Input text text which the user (programmer) has to enter in a P key Pi;

Output text output/result displayed on the screen after the P key Pi is pressed.

TY55 Definition of the 512-byte P key memory

This statement has to be issued after the PLUS call. It enables the 512-byte P key memory of the data display terminal to be used independently of the generation.

Operation	Operands	Meaning
TY55	[w]	Definition of the 512-byte P-key memory

w 8160, 8162, 9748, 9749, 9750, 9752, 9753 and 9754 DDTs issue a warning for every P key having an assignment longer than 16 bytes (PLUS V9.1B and higher).

P Definition of the P statement

The P statement is used to define the contents of a P key which are loaded at the terminal with the PUT or END statement.

Operation	Operands	Meaning
Pi_	$\left\{ \left\{ \begin{array}{l} \text{'text' } \\ \text{cmd[*n]} \\ \text{poscmd[:l/c]} \end{array} \right\} [, \dots] [, Pj] \right\}$	$\left. \begin{array}{l} \\ \\ \\ \end{array} \right\} \begin{array}{l} \text{Key contents} \\ \text{Keep key free} \\ \text{Clear key contents} \\ \text{Query key contents} \end{array}$

Pi_ P key to be programmed,
i = decimal number ($1 \leq i \leq 20$).

The specification of key Pi must be separated from the operand by one blank.

Example

Statement for P key 10: P10_

'text' Characters enclosed in quotes are loaded 1:1 into P key Pi. Any quotes to be interpreted as text must be doubled.

Example

Input text: 'WRPASS C'XYZ''
Output text: WRPASS C'XYZ'

cmd	<p>Mnemonic name of a control character string to be stored in P key Pi.</p> <p><i>Example</i></p> <p>Transmit all data from the beginning of the line to the cursor.</p> <p>Input text: EM,SZA,DUE</p>
*n	<p>Repetition factor; n = decimal number ($1 \leq n \leq 255$); store the control character sequence n times in P key Pi.</p> <p><i>Note</i></p> <p>Control character sequences specified using the repetition factor may not exceed a total length of 255 bytes.</p> <p><i>Example</i></p> <p>Move cursor 8 positions to the right.</p> <p>Input text: SMR*8</p>
poscmd	<p>Mnemonic name of a control character sequence containing the specification of an absolute position (POS, LVA, WDH) to be loaded into P key Pi (see page 242, "Positioning device functions").</p> <p><i>Example</i></p> <p>Position the cursor at the start of the screen and then initiate hardcopy output.</p> <p>Input text: POS,LA1</p>
:l/c	<p>Line and column specification for a control character sequence; l/c = decimal number ($1 \leq l \leq 24$ and $1 \leq c \leq 80$).</p> <p>If the line and column specification is missing, position 1/1 (start of screen) is assumed.</p> <p><i>Example</i></p> <p>Write blanks from cursor position to end of screen.</p> <p>Input text: WDH:24/80,' '</p>

- Pj Chain key Pj to contents of key Pi (explicit chaining),
j = decimal number ($i < j \leq 20$).
- When key Pi is pressed, the contents of Pi are executed first,
followed by the contents of Pj.
- Example*
- Chain key P6 to keys P4 and P5.
- ```
Input text:P4 'ASSIGN-SYSDTA',P6
Input text:P5 'ASSIGN-SYSLST',P6
Input text:P6 ' TO=*PRIMARY'
```
- Output text after pressing key
- P4: ASSIGN-SYSDTA TO=\*PRIMARY
  - P5: ASSIGN-SYSLST TO=\*PRIMARY
- \* Keep key Pi free (automatic chaining has no effect).
- Example*
- Keep key P17 free.
- ```
Input text:P17 *
```
- Delete contents of key Pi (automatic chaining has no effect).
- Example*
- Delete assignment of key P1.
- ```
Input text for key P1: -
```
- ? Display on screen in symbolic form the edited contents of key Pi.
- Example*
- Display the contents of key P7 symbolically.
- ```
Input text: P7 '123'
Input text: P7 ?
Output text: P07 '123'
```

Overwriting

In interactive mode, if there is already an entry for the specified P key, the statement is displayed together with the existing entry and a prompt asking whether the entry should be overwritten. Depending on the reply the old entry is then either left as it is or overwritten.

In batch mode there is no query: the old entry is overwritten.

Buffering

If the statement is valid, the corresponding device code is stored in the internal buffer.

If the statement is incorrect, it is logged with an appropriate error message on SYSOUT (in interactive mode on the terminal).

The position of the error is made to flash on the screen. Incorrect statements are not entered in the internal PLUS buffer.

P key memory

Terminals with 16 bytes per key

On 8160, 8162, 9748, 9749, 9750, 9752, 9753 and 9754 Data Display Terminals, the capacity of the P key memory is as follows:

– without automatic chaining: P1 through P20, 16 bytes per key

– with automatic chaining:

For these terminals PLUS has 20 P keys with 16 bytes each. If more than 16 bytes are assigned to a key, automatic chaining occurs (the appropriate control characters are inserted).

By using all the keys a maximum assignment length of 268 bytes is thus possible.

The following table shows the length of the individual assignment sections per key (with automatic chaining):

P1: 268 bytes *)	P6: 198 bytes	P11: 133 bytes	P16: 68 bytes
P2: 254 " *)	P7: 185 "	P12: 120 "	P17: 55 "
P3: 240 "	P8: 172 "	P13: 107 "	P18: 42 "
P4: 226 "	P9: 159 "	P14: 94 "	P19: 29 "
P5: 212 "	P10: 146 "	P15: 81 "	P20: 16 "

*) Since the input record length for control statements is limited to 256 bytes, only a maximum of 251 bytes can be utilized with keys P1 and P2.

Terminals with 512 bytes per key

For the 9755, 9758, 9759, 9762 and 9763 Data Display Terminals, PLUS provides an area of 512 bytes, where the P key assignments can be stored in the order in which they are defined by the user. The 512 bytes of the P key memory can be freely distributed over the 20 P keys.

MSG Definition of an output message

The MSG statement is used to generate a message to be displayed on the screen when the P keys are loaded (PUT or END statement). This message remains on the screen until the accompanying text "%PLEASE ACKNOWLEDGE" is acknowledged with DUE.

Operation	Operands	Meaning
MSG_	$\left\{ \left\{ \begin{array}{l} \text{'text' } \\ \text{cmd[*n]} \\ \text{poscmd[:l/c]} \end{array} \right\} [, \dots] \right\}$	$\left. \begin{array}{l} \\ \\ \end{array} \right\} \begin{array}{l} \text{Output message} \\ \text{Delete output message} \\ \text{Query output message} \end{array}$

'text' Characters enclosed in quotes are displayed 1:1 on the screen. Any quotes to be interpreted as text must be doubled.

Example

Request acknowledgment.

```
Input text: 'P keys loaded. Please enter "+".'
Output text: P keys loaded. Please enter '+'.
             %PLEASE ACKNOWLEDGE
```

cmd Mnemonic name of a control character sequence to be executed.

Example

Cursor at start of screen

```
Input text: SBA
```

- *n Repetition factor; $n = \text{decimal number } (1 \leq n \leq 255)$.
Store the control character sequence n times in key Pi.
- Note*
Control character sequences specified using the repetition factor may not exceed a total length of 255 bytes.
- Example*
Move cursor to start of next line; execute 4 times.
- Input text: SNZ*4
- poscmd Mnemonic name of a control character sequence containing the specification of an absolute position (POS, LVA, WDH) to be displayed on screen (see page 242 "Positioning device functions").
- Example*
Set cursor to start of screen.
- Input text: POS
- :/c Line and column specification for a control character sequence; $l/c = \text{decimal number } (1 \leq l \leq 24 \text{ and } 1 \leq c \leq 80)$.
If the line and column specification is omitted, position 1/1 (start of screen) is assumed.
- Example*
Position the cursor at the beginning of line 5.
- Input text: POS:5/1
- Delete a previously edited message.
- Example*
Delete message.
- Input text: MSG –
- ? Display previously edited message on the screen in symbolic form.
- Example*
Display screen message symbolically.
- Input text: MSG 'abc'
Input text: MSG ?
Output text: MSG 'abc'

Overwriting

In interactive mode, if there is already an entry for the specified output message, the statement is displayed together with the existing entry and a prompt asking whether the entry should be overwritten. Depending on the reply the old entry is then either left as it is or overwritten.

In batch mode there is no query: the old entry is overwritten.

Buffering

If the statement is valid, the corresponding device code is stored in the internal buffer. If the statement is incorrect, it is logged with an appropriate error message on SYSOUT (in interactive mode on the terminal). The position of the error is caused to flash on the screen. Incorrect statements are not entered in the internal PLUS buffer.

DEL Delete P key contents

The DEL statement is used to ready the contents of all the P keys and/or an output message for deletion with the PUT or END statement.

Operation	Operands	Meaning
DEL		Ready all P key contents loaded in the data display terminal and/or an output message for deletion

Overwriting P key contents/output messages

In interactive mode, if there is already an entry for a P key and/or an output message, the statement is displayed together with the last entry. The original entry is overwritten. If a warning was issued beforehand for a P key and if DOC OUT is activated, a prompt asking whether the entry should be overwritten appears. Depending on the reply the old entry is then either left as it is or overwritten. In batch mode there is no query: the old entry is overwritten.

If there is no entry, subsequent P/MSG statements are processed without a query. When the P key assignments are output, only the unused keys are cleared.

DOC Definition of the documentation medium

The DOC statement is used to assign the medium on which P or MSG statements are to be documented at PUT or END. The length of MSG documentation depends on the P key memory of the data display terminal; the maximum length is 250 bytes or 506 bytes (see page 226 "P key memory").

Operation	Operands	Meaning
DOC_	$\left\{ \begin{array}{l} \text{N[ONE]} \\ \text{O[UT]} \\ \text{L[ST]} \\ \text{B[OTH]} \end{array} \right\}$	No documentation Documentation on SYSOUT Documentation on SYSLST Documentation on SYSOUT and SYSLST

NONE No documentation (default).

OUT Documentation on SYSOUT (in interactive mode the display terminal). The statements displayed at the terminal can be modified and reentered as control statements.

LST Documentation on SYSLST. If SYSLST is a cataloged file, it can be reused as an input file.

BOTH Documentation on SYSOUT **and** SYSLST.

SAV Saving in a BS2000 library

The SAV statement can be used to store P key formats and/or an output message in a BS2000 library (PLAM or LMR).

Operation	Operands	Meaning
SAV_	mapname[,libname]{[,*PLAM]} {[,*LMR]}	Save P key formats and/or output message in a PLAM or LMR library as a member of type R

mapname Member name under which the P key format and/or the output message is to be included in the library (1 - 8 alphanumeric characters; the 1st character must be a letter).

libname Name of a BS2000 library in which the members (mapname) are to be stored (1 - 54 characters). If no library name is specified, the member is stored in the library "PLUS.MAPLIB" of the user. If this library does not exist, it is generated by PLUS.

{ *PLAM }
{ *LMR }

Defines the type of library in which the P key formats and/or output messages are to be stored. If no library type is specified, the following default values are assumed:
for BS2000 versions ≥ 10 *PLAM and
for BS2000 versions < 10 *LMR

Saving in a BS2000 library

If the statement is valid, it is processed in the same way as "PUT" (automatic chaining of P key contents that are too long, for restrictions see page 226 "P key memory"). The physical output message generated is not, however, issued to the terminal. With the PLAM utility, the output message is saved interactively as a library member of type R. With the LMR utility, the output message is stored in a batch task. In both cases, no listing is generated.

If the statement is incorrect, it is logged on SYSOUT with an appropriate error message. The position of the error is made to flash on the screen. No further processing takes place.

GET Loading from a BS2000 library

The GET statement is used to load P key formats and/or an output message which were previously stored with SAV (in interactive mode only).

Operation	Operands	Meaning
GET_	mapname[,libname]	Load P key formats and/or output message from an OML

mapname Member name under which the P key formats and/or the output message from the library is to be output.

libname Name of a BS2000 library from which the members (mapname) are to be loaded. If no library name is specified, the member is loaded from the library "PLUS.MAPLIB" of the user.

Loading from a BS2000 library

If the statement is valid, the P key assignment or the output message is loaded from the BS2000 library into the program area.

If the loading is completed successfully, the P keys at the terminal are loaded. Using the DOC statement the P/MSG statements can also be logged in symbolic form. The loaded area is then released.

If the loading operation is not completed successfully, an appropriate (system) error message is issued and processing aborted.

Note

The control statement GET has no effect on the internal PLUS buffer. If the stored P key formats are to be changed, the statements documented on SYSOUT can be reused as control statements.

PUT Loading the P keys

The effect of the PUT statement depends on the control statement used:

- P/MSG: the defined P key contents and/or an output message are output to the terminal (in interactive mode only);
- DOC: the P and/or MSG statement are documented;
- DEL: the P key contents and/or an output message are deleted.

Note

On terminals with a 512-byte P key memory (see page 226 "P key memory"), only the remaining free memory locations of the terminal can be used. Longer assignments are truncated without a warning message.

Operation	Operands	Meaning
PUT		Load P key contents and/or output message

Loading at the terminal

The entries in the internal PLUS buffer for filling the P areas at the terminal are grouped together to form one physical output message.

If an entry is longer than 16 bytes, free P keys are chained (for restrictions see page 226 "P key memory").

If an entry cannot be automatically chained, the statement is logged on SYSOUT with an appropriate error message. The position of the error is caused to flash on the screen. The entry is not output.

Once all the entries have been processed, the P areas at the terminal are loaded. Using the DOC statement the P/MSG statements can also be logged in symbolic form.

SYS System mode

The SYS statement is used to change over to system mode. In this mode general BS2000 commands can be issued (i.e. reassignment of SYSDTA etc.). The command /RESU-PROG is used to return to the program and continue reading in from SYSDTA.

Operation	Operands	Meaning
SYS		Switch to system mode

END End input

The END statement is used to end the program.

Operation	Operands	Meaning
END		End command

If P key contents or an output message have been defined and not yet processed with PUT, the appropriate processing is initiated. The program is then terminated.

STOP Exit PLUS

The STOP statement is used to exit the program.

Operation	Operands	Meaning
STOP		Exit the PLUS program

If P key contents or an output message have been defined and not yet processed with PUT, no further processing is initiated. The program is then exited.

13.2 Calling the PLUS program

The program PLUS is called with the START-PROGRAM command:

```
/START-PROG FROM-FILE=[:catid:][$userid.]PLUS
```

PLUS then expects the entry of control statements.

Input medium

PLUS reads the control statements from the logical system file SYSDTA. SYSDTA may be:

- the display terminal,
- a member of a PLAM library,
- a cataloged SAM or ISAM file,
- an SDF-P variable,
- the command file SYSCMD.

The assignment can be changed with

```
/ASSIGN-SYSDTA TO-FILE=
```

- before the PLUS program is started,
- after the control statement SYS,
- after EOF has been detected in a cataloged file.

If the end of the input medium is detected before the END statement, it has the same effect as the SYS control statement, i.e. a changeover to system mode takes place.

Calling the program PLUS from a CALL procedure

The CALL procedure, stored as a file, is started with the CALL (system) command.

Example

A CALL procedure for loading P key contents. The following procedure is stored e.g. under the file name CALL.PKEY.

```

/BEGIN-PROC
/MOD-JOB-SW ON=1
/ASS-SYSDTA *SYSCMD
/START-PROG PLUS
P1 'ASS-SYSLST ',P3
P2 'ASS-SYSDTA ',P3
P3 '*PRIMARY',EM,DUE
P4 'START-PROG $COBOL85',EM,DUE
P5 'CALL-PROC DO.COBOLE',EM,DUE
P6 'START-PROG $ARCHIVE',EM,DUE
P7 'SET-LOGON-PAR MAS,1234',EM,DUE
P8 'O $DIALOG,1/12',EM,DUE
P9 'START-PROG $IFG',EM,DUE
P10 'START-PROG $EDT',EM,DUE
P11 'EXIT-JOB KEEP=YES',EM,DUE
P12 'SH-USER-STATUS',EM,DUE
P13 'SH-FILE-ATTR',EM,DUE
P14 'START-PROG $LMS',EM,DUE
MSG SNZ,'**** P keys loaded ****'
PUT
END
/ASS-SYSDTA *PRIMARY
/MOD-JOB-SW OFF=1
/END-PROC

```

Starting the procedure file with the CALL command

The file stored under the file name CALL.PKEY is called with the CALL command.

```

/CALL-PROC CALL.PKEY
% P500 LOADING
**** P keys loaded ****
%PLEASE ACKNOWLEDGE

```

Storing P key formats

Initiated by the control statement SAV mapname[,libname][,*PLAM] or SAV mapname[,libname][,*LMR], PLUS stores the defined P key contents in a BS2000 library. This is done in a form which allows subsequent implementation by the formatting system FHS.

Loading P key formats with an application program

ASSEMBLER application programs can process stored P key formats, even without FHS. For BS2000 versions < V10, the system macro call

```
LINK ENTRY=mapname,LIBNAME=libname
```

causes the area illustrated below to be loaded into the program area. If the loading is completed successfully, register 1 points to the beginning of the area.

For BS2000 versions ≥ V10, the system macro call BIND must be used. The start address of the area is returned when the SYMBLAD parameter is used.

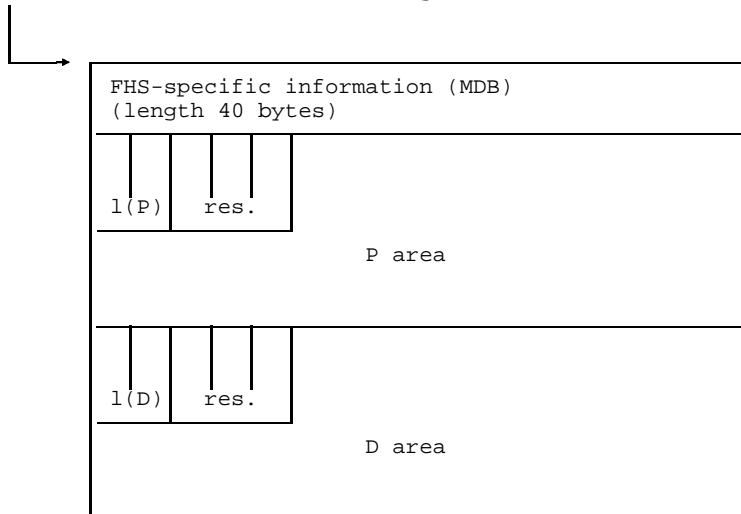
The start addresses, incremented by the appropriate displacement, must then be entered in the parameter area of the output macros WROUT or WRTRD.

The P area can then be output in physical mode (in accordance with WROUT/WRTRD ... ,MODE=PHYS,OHDR=N), and the D area in line mode (in accordance with WROUT/WRTRD ... ,MODE=LINE).

For BS2000 versions < V10, the loaded area can subsequently be released using the system macro UNLOAD. For BS2000 versions ≥ V10, the loaded area can subsequently be released using the system macro UNBIND.

Structure of the P key format

Register 1 after successful loading



Area length l(P) or l(D): 2 bytes
reserved: 3 bytes

The **P area** contains text and device control characters for loading the P programs. In the **D area** the P programs are documented.

Loading P key formats with FHS

FHS (Format Handling System) is a chargeable product available for TIAM, DCAM and UTM access methods and the ASSEMBLER and COBOL programming languages (see 'FHS User Guide').

P key formats stored by PLUS are output the same way as "regular" FHS formats. The format name is the name used when the format was stored with SAV, and the format library the relevant respective library.

The format can be used only as an output format or as an I/O format. Input (e.g. requested by a prompt text in the control statement MSG) cannot be processed by FHS, although it can be used as an acknowledgment (with any contents) for the loading operation. It is not advisable to use a PLUS format as input/output format unless it contains an MSG statement.

Note, too, that a PLUS format with an MSG statement should not be sent to a formatted screen, because part of the screen output will be overwritten by the message (see 'FHS User Guide').

Loading P key formats with UTM

P key formats defined with the program PLUS can be loaded into the appropriate UTM format library as FHS formats using the SAV statement

```
SAV mapname,UTM format library
```

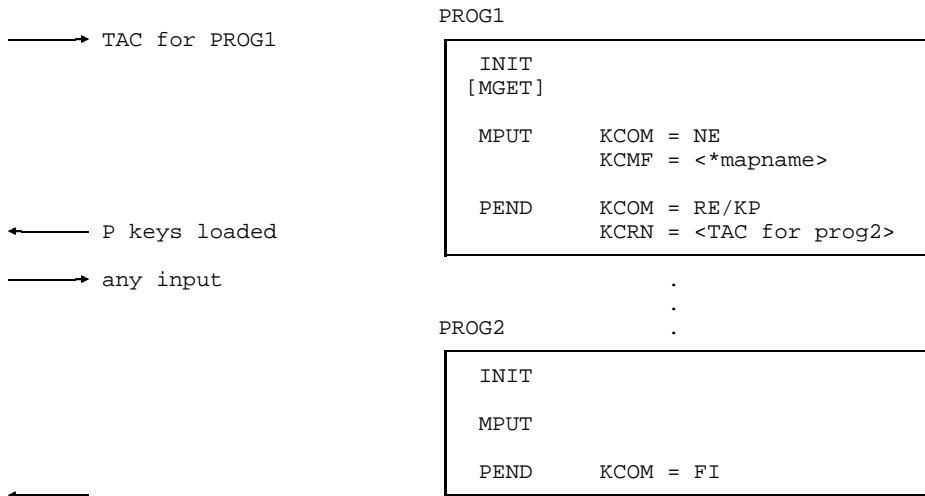
Integrated loading of P keys

Integrated P key loading is possible with # formats. The name of the P key format must be specified as a global attribute when a format is output. The P key format is then output with the output of the actual format (see 'FHS User Guide'). Using the MSG control statement is illegal for integrated P key output. A separate dialog step for output of the P key format is not necessary in this case. For UTM applications, integrated P key loading should only be carried out in the sign-on conversation.

Loading of P keys in a separate dialog step

The P keys are loaded by means of an MPUT call (in a UTM program unit) in which KDCMF=*mapname. The next time something is entered at the terminal, input formatting is performed. A P key format constitutes a special case, so the program following the loading must not contain an MGET call otherwise the program is aborted with 70Z/FR04 and a dump.

Example



13.3 Device function mnemonics

Cursorbewegungsfunktionen

Cursor movement functions

Symbol	Length	Meaning
SBA	**	Cursor to start of screen
SDZ	**	Cursor to start of line above
SML	**	Cursor one position to the left
SMO	**	Cursor up one line
SMR	**	Cursor one position to the right
SMU	**	Cursor down one line
SNZ	**	Cursor to next line
SZA	**	Cursor to start of line
TAL	**	Tabulator jump to the left
TAR	**	Tabulator jump to the right

Text shifting functions

Symbol	Length	Meaning
AFG	**	Delete character
AFZ	**	Delete line
EFG	**	Insert character
EFZ	**	Insert line
RU	**	Roll-up

Delete functions

Symbol	Length	Meaning
LSP	**	Clear memory
LVD	**	Delete variable data
LZF (LVA)	**	Delete up to end of line/field (see "Positioning device functions")

Data transmission functions

Symbol	Length	Meaning
DUE	**	Data transmission 1
DUE1	**	Data transmission 1
DUE2	***	Data transmission 2
F1	**	Short code 1 with text
F2	**	Short code 2 with text
F3	**	Short code 3 with text
F4	**	Short code 4 with text
F5	**	Short code 5 with text
F6	**	Short code 6 with text
F7	**	Short code 7 with text
F8	**	Short code 8 with text
F9	**	Short code 9 with text
F10	**	Short code 10 with text
F11	**	Short code 11 with text
F12	**	Short code 12 with text
F13	**	Short code 13 with text
F14	**	Short code 14 with text
F15	**	Short code 15 with text
F16	**	Short code 16 with text
F17	**	Short code 17 with text
F18	**	Short code 18 with text
F19	**	Short code 19 with text
F20	**	Short code 20 with text
F21	**	Short code 21 with text
F22	**	Short code 22 with text
F23	**	Short code 23 with text
F24	**	Short code 24 with text
K1	**	Short code 1
K2	**	Short code 2
K3	**	Short code 3
K4	**	Short code 4
K5	**	Short code 5
K6	**	Short code 6
K7	**	Short code 7
K8	**	Short code 8
K9	**	Short code 9
K10	**	Short code 10
K11	**	Short code 11
K12	**	Short code 12
K13	**	Short code 13
K14	**	Short code 14

Functions for intra-system data exchange

Symbol	Length	Meaning
LA1	**	Local initiation 1
LA2	**	Local initiation 2
LA3	***	Local initiation 3
LA4	***	Local initiation 4
LA5	***	Local initiation 5
LA6	***	Local initiation 6
LA7	***	Local initiation 7

Special functions

Symbol	Length	Meaning
AKA	**	Acoustic alarm
AM	***	Start marker
BRS	**	Reset BEL indicator
DEL	*	Smudge character
EM	*	End marker
FAZ	**	Reset field modifications
LZE	**	Logical end of line
MAR	**	Mark field
PAK	***	Parameter announcer
RS	**	Reset
SP	*	Space (blank)
SUB	*	Substitute character
VA	**	Clear connection

Positioning device functions

Symbol	Length	Meaning
LVA:l/c	*****	Delete variable data from cursor up to before position line "l", column "c"
POS:l/c	****	Cursor to position line "l", column "c"
WDH:l/c	*****	Repeat next character from cursor up to before position line "l", column "c"

P key chaining function

Symbol	Length	Meaning
P2	**	P key 2
P3	**	P key 3
P4	**	P key 4
P5	**	P key 5
P6	**	P key 6
P7	***	P key 7
P8	***	P key 8
P9	***	P key 9
P10	***	P key 10
P11	***	P key 11
P12	***	P key 12
P13	***	P key 13
P14	***	P key 14
P15	***	P key 15
P16	***	P key 16
P17	***	P key 17
P18	***	P key 18
P19	***	P key 19
P20	***	P key 20

Display control function

Symbol	Length	Meaning		
		intensity	italics/underscored	flashing
DU	**	dark	no	no
HHKTBL	**	low	yes	yes
HHKTNB	**	low	yes	no
HHNTBL	**	low	no	yes
HHNTNB	**	low	no	no
NHKTBL	**	high	yes	yes
NHKTNB	**	high	yes	no
NHNTBL	**	high	no	yes
NHNTNB	**	high	no	no

Field handling control function

Symbol	Length	Meaning				
		protected	transferable	markable	printable	numeric
VNMNDA	**	no	yes	no	no	no
VNMNDN	**	no	yes	no	no	yes
VNMNDRN	**	no	yes	no	yes	no
VNMNDRN	**	no	yes	no	yes	yes
VMANDA	**	no	yes	yes	no	no
VMANDN	**	no	yes	yes	no	yes
VMADRA	**	no	yes	yes	yes	no
VMADRN	**	no	yes	yes	yes	yes
GNMNDA	**	yes	no	no	no	no
GNMNDN	**	yes	no	no	no	yes
GNMNDRA	**	yes	no	no	yes	no
GNMNDRN	**	yes	no	no	yes	yes
GMANDA	**	yes	no	yes	no	no
GMANDN	**	yes	no	yes	no	yes
GMADRA	**	yes	no	yes	yes	no
GMADRN	**	yes	no	yes	yes	yes
UNMNDA	**	yes	yes	no	no	no
UNMNDN	**	yes	yes	no	no	yes
UNMNDRA	**	yes	yes	no	yes	no
UNMNDRN	**	yes	yes	no	yes	yes
UMANDA	**	yes	yes	yes	no	no
UMANDN	**	yes	yes	yes	no	yes
UMADRA	**	yes	yes	yes	yes	no
UMADRN	**	yes	yes	yes	yes	yes
ANMNDA	**	yes	address	no	no	no
ANMNDN	**	yes	address	no	no	yes
ANMNDRA	**	yes	address	no	yes	no
ANMNDRN	**	yes	address	no	yes	yes
AMANDA	**	yes	address	yes	no	no
AMANDN	**	yes	address	yes	no	yes
AMADRA	**	yes	address	yes	yes	no
AMADRN	**	yes	address	yes	yes	yes

13.4 General messages

*** MAP SAVING INITIALIZED ***

Meaning: The ENTER file for saving the P key formats and/or the output message has been generated and initialized.

Action: No response required.

*** EXISTING <old P key contents or output message> *** OVERWRITE ? (Y/N)

Meaning: A P statement or output message has already been defined. Query as to whether it should be overwritten.

Action: Answer 'yes' or 'no' required.

*** ERRORS DETECTED ! CONTINUE ? (Y/N)

Meaning: An error occurred during processing which is indicated by an appropriate error message. Query as to whether the function should still be executed.

Action: Answer 'yes' or 'no' required.

13.5 Error messages

*** P-KEYS CANNOT BE LOADED ON THIS DEVICE

Meaning: The P keys cannot be loaded at the specified terminal because either this display terminal has no P keys or an attempt was made to load P key formats on a terminal with max. 16 bytes per key. Some P key contents longer than 16 bytes were stored in the process.

Action: Connect appropriate terminal.

*** WARNING 9755 FORMAT. P-KEYS LOADED POSSIBLY WITH ERRORS

Meaning: An attempt was made to load P key formats on the data display terminal that have P key contents > 16 bytes.
Depending on the generation parameters, the terminal only has P key areas with a maximum length of 16 bytes each. Despite this, an attempt is made to load the P key format.
P key remainders > 16 bytes are truncated by data display terminals which can only use 16 bytes per key.

Action: Verify whether generation parameters and device match.
If necessary change P key format.

*** UNKNOWN OPERATION

Meaning: Invalid operation code.

Action: Repeat input with correct operation code.

*** UNKNOWN OPERAND

Meaning: Invalid mnemonic name for an operand.

Action: Repeat input with correct operand.

*** UNKNOWN DOC PARAMETER

Meaning: Invalid mnemonic name for an operand in the DOC statement.

Action: Repeat input with correct operand.

*** DELIMITER MISSING

Meaning: A mandatory delimiter ("*,/") was omitted in the control statement.

Action: Correct the input.

***** TOO MANY OPERANDS**

Meaning: Too many operands were entered for the specified P key (buffer overflow), or a quote is missing.

Action: Repeat input

- with fewer operands,
- for a lower P key (only on terminals with max. 16 bytes per key due to automatic chaining),
- terminate operands with a quote.

***** INPUT TOO LONG: TRUNCATED**

Meaning: Only max. 256 or 512 characters of the input (depending on the terminal) are allowed.

Action: Repeat input using control statement of correct length.

Note

The error message "*** TOO MANY OPERANDS" is also issued.

***** P KEY NO OUTSIDE RANGE 1-20**

Meaning: Invalid P statement; key Pi is outside permitted range.

Action: Repeat input with correct P key number.

***** KEY TO CHAIN NOT GREATER THAN CURRENT KEY**

Meaning: P key number too small for explicit chaining.

Action: Repeat input with different chaining key.

***** ILLEGAL OPERANDS AFTER CHAINED KEY**

Meaning: A P operand (explicitly chained P key) is not the last operand in a P statement; subsequent operands are ignored.

Action: Enter operand for chained key in a separate P statement.

*** NO KEY AVAILABLE TO CHAIN <Pi>

Meaning: The P key contents are too long for the specified P key; no P key free for automatic chaining (for restrictions see page 226 "P key memory").

Action: Correct the input.

Note

The error message "*** ERRORS DETECTED ! CONTINUE ? (Y/N)" is also issued.

*** REPETITION FACTOR OUTSIDE RANGE 1-255

Meaning: Repetition factor is invalid.

Action: Repeat input with a correct repetition factor.

*** LINE NUMBER OUTSIDE RANGE 1-24

Meaning: The line number in a positioning statement is invalid.

Action: Repeat input with a correct line number.

*** COLUMN NUMBER OUTSIDE RANGE 1-80

Meaning: The column number in a positioning statement is invalid.

Action: Repeat input with correct column specification.

*** MAP TO BE LOADED WAS NOT A PLUS MAP

Meaning: An attempt was made to load link modules that are not P key formats from an OML.

Action: Repeat input with the correct mapname.

*** MAP NAME MISSING

Meaning: The save name has not been specified.

Action: Repeat input with a correct save name.

*** ILLEGAL MAP NAME

Meaning: The save name is too long or contains an illegal character.

Action: Repeat input with a correct save name.

*** ILLEGAL LIBRARY NAME

Meaning: The name of the save library is too long or contains an illegal character.

Action: Repeat input with the correct save library name.

*** PLAM NOT AVAILABLE

Meaning: The PLAM library program is not available in this system. The P key contents cannot be stored in a PLAM library.

Action: System administration action is required.

*** PLAM ERROR dddd

Meaning: Error in the PLAM library program during processing of SAV. Further error information can be obtained with the command: /HELP-MSG-INFO PLAdddd.

Action: All actions depend on the type of the current error.

14 Appendix

14.1 Table of standardized function key codes

Code normal	with ¹⁾ delete	Meaning	8110	8151	8152	816x	9749 975x	9763	3270 ³⁾
00	10	Data transmission (normal)	DÜ	DÜZ, DÜM	DÜZ, DÜM, DÜB	DÜ, DÜ1, DÜ2	DÜ, DÜ1, DÜ2	SEND	ENTER
01	11	Short code 1		FT1	F1	K1	K1	K1	PA 1
02	12	" 2		FT2 ²⁾	F2 ²⁾	K2 ²⁾	K2 ²⁾	K2 ²⁾	PA 2 ²⁾
03	13	" 3		DVA	F3	K3	K3	K3	PA 3 PF6
04	14	" 4			F1+FZ	K4	K4	K4	PF7
05	15	" 5			F2+FZ ²⁾	K5	K5	K5	PF8
06	16	" 6			F3+FZ	K6	K6	K6	PF9
07	17	" 7				K7	K7	K7	PF10
08	18	" 8				K8	K8	K8	PF11
09	19	" 9				K9	K9	K9	PF12
0A	1A	" 10				K10	K10	K10	PF13
0B	1B	" 11				K11	K11	K11	PF14
0C	1C	" 12				K12	K12	K12	PF15
0D	1D	" 13				K13	K13	K13	PF16
0E	1E	" 14				K14	K14	K14	PF17
20	30	Data transmission (marked) 0			DÜZ, DÜM, DÜB +FZ		DÜZ, DÜM, DÜB, +FZ		
21	31	" 1				F1	F1	F1	PF 1
22	32	" 2				F2	F2	F2	PF 2
23	33	" 3				F3	F3	F3	PF 3
24	34	" 4				F4	F4	F4	PF 4
25	35	" 5				F5	F5	F5	PF 5
26	36	" 6					F6	F6	PF 18
27	37	" 7					F7	F7	PF 19
28	38	" 8					F8	F8	PF 20
29	39	" 9					F9	F9	PF 21

Code normal	with 1) delete	Meaning	8110	8151	8152	816x	9749 975x	9763	3270 3)
2A	3A	" 10						F10	PF22
2B	3B	" 11						F11	PF23
2C	3C	" 12						F12	PF24
2D	3D	" 13						F13	
2E	3E	" 14						F14	
2F	3F	" 15						F15	
30		Data transmission (positional data)			PŮ				
40	50	Data transmission (special) 0					BYPASS input		
41	51	" 1					positive ack- knowledgegment		
42	52	" 2					negative ack- knowledgegment		
43	53	" 3							
45	55	Mag. ID card reader					x	x	x
46	56	Data transmission (marked) 16						F16	
47	57	" 17						F17	
48	58	" 18						F18	
49	59	" 19						F19	
4A	5A	" 20						F20	
4B	5B	" 21						F21	
4C	5C	" 22						F22	
4D	5D	" 23						F23	
4E	5E	" 24						F24	
10		Delete memory							CLEAR
60		Chip card terminal						x	
80		ID card inserted					x	x	x

- 1) Screen cleared prior to input
- 2) When using the TIAM access method, reserved for ESCAPE/BREAK function
- 3) For default values, see VTSU operating parameter COMPKEYS, page 195.

14.2 Support for terminals

The terminals supported by VTSU are listed below.

Data display terminals supported:

3270	8150	8160	9748	9750	9763
	8151	8162	9749	9751	9770
	8152	8167		9752	
	x29			9753	
				9754	
				9755	
				9756	
				9758	
				9759	

Printers supported:

3287	8121	9001-893	9011-18/28	9021-2
4810-P10	8122	9001-3x	9011-19/29	9022-200
	8124	9001-xxx	9012	9022
		9002	9013	9025
		9003	9014-11	90037
		9004	9014-15	9026
			9014-16	

14.3 Messages

DCM0200 VTSU-B NOT LOADED

Meaning

VTSU-B is not loaded. As a result it cannot be called.

DCM0202 VTSU-B INIT: DMS ERROR '(&01)' WHEN OPENING FILE '(&00)'.
IN SYSTEM MODE: /HELP-MSG DMS(&01)

Meaning

File (&00) exists but cannot be opened.

Possible reasons:

- The file is defective.
- The file is protected by a password.

For more detailed information about the DMS error code enter /HELP-MSG in system mode or see the BS2000 manual 'System Messages'.

Response

Take measures in correspondence to the DMS error code.

If necessary specify the password.

If the file is defective, reconstruct the file.

Contact the system administrator, if necessary.

DCM0203 VTSU-B INIT: DMS ERROR '(&01)' WHEN ACCESSING FILE '(&00)'.
IN SYSTEM MODE: HELP-MSG DMS(&01)

Meaning

The user is not authorized to read the file (&00).

For more detailed information about the DMS error code enter /HELP-MSG in system mode or see the BS2000 manual 'System Messages'.

DCM0204 VTSU-B INIT: DMS ERROR '(&01)' WHEN CLOSING FILE '(&00)'.
IN SYSTEM MODE: /HELP-MSG DMS(&01)

Meaning

For more detailed information about the DMS error code enter /HELP-MSG in system mode or see the BS2000 manual 'System Messages'.

Response

Check the file, if necessary contact the system administrator.

DCM0205 VTSU-B INIT: VTSU CONTROL RECORD MISSING IN FILE '(&00)'

Meaning

The first line must begin /BEGIN VTSU PARAMS or /BEGIN VTSU-PARAMS.

Response

Correct the file.

DCM0206 VTSU-B INIT: INVALID OR UNKNOWN PARAMETER '(&00)' IN FILE '(&01)'

Response

Correct the file.

DCM0207 VTSU-B INIT: END RECORD MISSING IN FILE '(&00)'

Meaning

The very last line of the file (&00) must be /END PARAMS or /END-PARAMS.

Response

Correct the file.

DCM0208 'CSTRING' OPERAND: '(&00)' UNKNOWN OR ILLEGAL BY SUBSYSTEM VTSUTRAC
INITIALIZATION

Meaning

The value of the CSTRING operand in the /CREATE-SUBSYSTEM command contains unknown or illegal parameters.

Response

Correct the operand value and try again.

DCM0209 WARNING: INTERNAL ERROR X'(&00)' IN '(&01)' INTERFACE.
7 BIT CODED CHARACTER SET ASSUMED

Meaning

Internal error in system interfaced.

Response

Analyze the received error code and contact the system administrator.

DCM0210 WARNING: INVALID USER DEFAULT CODED CHARACTER SET.
STANDARD 7 BIT CODED CHARACTER SET ASSUMED

Meaning

The user default coded character set is invalid.

Possible causes are:

- (1) the default coded character set is not known by XHCS-SYS.
- (2) the default coded character set (not EDF031RV) has a restricted compatibility with its reference coded character set.
- (3) the default coded character set is not a EBCDIC one.
- (4) the special routines for Arabic coded character set support are not present.

Response

- Ask the system administrator to define the desired 8 bit coded character set in the XHCS-SYS tables or
- ask the system administrator for the list of 8 bit character sets supported by XHCS-SYS and supported by this terminal and have such a code defined as user default character set or set in XHCS tables
- ask the system administrator to install the special routines for Arabic coded character set support

DCM0211 **WARNING: USER DEFAULT CODED CHARACTER SET NOT SUPPORTED BY TERMINAL. 7 BIT CODED CHARACTER SET ASSUMED**

Meaning

The user default coded character set is a 8 bit coded character set that the terminal does not support.

Response

- Employ a terminal supporting this 8 bit coded character set or
- modify the set up of the terminal for this 8 bit coded character set (with SIDATA) or
- ask the system administrator to modify the user default coded character set to one supported by the terminal.

DCM0212 **WARNING: XHCS NOT AVAILABLE. 7 BIT CODED CHARACTER SET ASSUMED**

Meaning

The subsystem XHCS-SYS is not available. As a result the use of the 8 bit functionality is not possible. Only the 7 bit functionality is available.

Response

Ask the system administrator to remove user default coded character set to install XHCS.

DCM0213 **WARNING : INVALID USER DEFAULT CODED CHARACTER SET WITH 7 BIT TERMINAL. 7 BIT CODED CHARACTER SET ASSUMED**

Meaning

On a 7-bit terminal, only blanks or EDF031RV is allowed as user default coded character set name.

Response

- Check user default character set name and employ an adequate terminal or
- ask the system administrator to change the user default coded character set to EDF031RV.

DCM0214 WARNING: XHCS NOT PRESENT. 7 BIT CODED CHARACTER SET ASSUMED

Meaning

The subsystem XHCS-SYS is not present. As a result the use of the 8 bit functionality is not possible. Only the 7 bit functionality is available.

Response

Contact the system administrator about XHCS-SYS availability.

DCM0215 VTSU-B V11.0A VTSUCB V04

DCM0216 REQUESTED VARIANT IS NOT SUPPORTED

DCM0217 NO MEMORY AVAILABLE

DCM0218 ESC-SEQUENCE NOT ALLOWED

DCM0219 OVERFLOW OF BUFFER

DCM0220 ERROR IN TABLE

14.4 Glossary

Application

An application processes requests exchanged between communication partners. A distinction is drawn between system applications and user-defined applications. There are, for example DCAM, TIAM, UTM and PDN applications.

BAM

Bit-serial asynchronous transfer procedure for connecting data terminals to cluster controllers.

BCAM

Basic Communication Access Method in BS2000

Broadcast

A message sent to all stations in a network

CCITT

Comite Consultatif International Telegraphique et Telephonique

Character set

Set of letters, digits and special characters from which words and other elementary components of a language (including computer languages) are constructed.

CMX

Communications Manager SINIX

Transport access system of INCA. System program discharging functions situated above the Transport Layer (Layer 4) of the OSI Reference Model.

Communication Access Method

Software providing the application with interfaces for communication.

Communication computer

Computer in TRANSDATA with the PDN operating system.

Connection

Association between two communication partners allowing them to exchange data.

Control character

A character that influences the transfer or processing of characters. Any symbol can be declared a control character.

Data

Characters or continuous functions representing information by known or agreed conventions and primarily for the purposes of processing.

Data communication system

A complex combination of hardware and software products that permits communication partners to exchange data in accordance with certain rules.

Data display terminal (DDT)

Device with monitor and keyboard for the input and output of data.

Datagram

Self-contained message.

Data medium

Medium used for long-term storage of data, e.g. floppy disks, magnetic disks, magnetic tapes.

Data station, terminal

Data stations may be data display terminals, PCs, teleprinters or printer stations or data-station systems.

DCAM

Data Communication Access Method (BS2000 software product)

Format

Logical data structure describing a form.

Format terminal

Operating mode of a virtual terminal, in which the message consists of a format (form, mask).

Front-end processor (FEP)

Communication processor connected directly to the I/O channel of a general-purpose computer.

Gateway

Computer forming the interface between two or more networks with differing protocol environments.

Hardcopy mode

Operating mode of a printer connected locally to a data display terminal. A message displayed on the screen is also printed on the printer.

Host

Designation for BS2000 systems or MVS systems in SNA, etc.

IDCAM

Interface DCAM

IFG

The Interactive Format Generator (IFG) is a tool for generating, modifying and administrating formats for the dialog between terminal users and application programs.

Installation

To prepare a computer (e.g. INCA) for use by loading the system software supplied with the computer from the data mediums and initializing the software.

Intelligent Network and Communication Attachment (INCA)

The INCA is a small, powerful processor that combines the functionality of a cluster controller and an administration center. As a cluster controller, the INCA provides a simple means of connecting data display terminals and printers to a local area network or public data network, controlling communication between the data terminals and applications in remote BS2000 hosts. In its capacity as administration center the INCA institutes autonomous administration for all the data display terminals and printers in its domain.

Interface

Posited or actual transition at the boundary between two functional units with agreed rules for the transfer of data or signals.

Internet protocol (IP)

IP is a protocol of the TCP/IP architecture. It is responsible for routing in a computer network.

ISO

International Organization for Standardization

Line terminal

Operating mode of a virtual terminal in which the message is structured in lines.

Message

A logically associated set of data to be sent by a communication partner or received by a communication partner.

NEA

TRANSDATA-specific communication protocol. In terms of function, the NEA network architecture largely complies with the layers of the ISO/OSI Reference Model.

NEABT

Protocol for controlling data display terminals and printers in interactive mode or transaction mode by means of formatted message headers.

NEABV

Protocol for ascertaining a user service via a transport connection and its parameters, for example partner characteristics or character sets.

OSI Reference Model

The OSI Reference Model (Open System Interconnection) lays down a framework for the standardization of communications between open systems. ISO has codified this model in the international standard ISO IS7498. The OSI Reference Model divides the functions needed for communication between systems across seven logical layers. Each layer has clearly defined interfaces to the next highest and next lowest layers.

Packet

A group of characters of a number defined by the data network, treated as a unit and containing commands and data for transfer.

Packet Assembly/Disassembly Facility (PAD)

Adaptation function permitting X.28 terminals access to packet switching networks. PAD packages character strings and converts codes and procedures.

Packet switching

The process of receiving, buffering and passing on data packets in data networks.

PDN

Program system for data communication and network control (operating system of communication computers)

Process

A facility for executing a program within a task.

Protocol

Rules for the coordination and cooperation of entities on the same functional layer in the data communication system, thus permitting global usage of the services provided by these entities.

SNA

Systems Network Architecture, IBM communication structure

SOCKET interface

A group of programming interfaces exclusively for communication between processes. The Socket interface is a generic communications interface. In addition to TCP and UDP, it can be used to implement local interprocess communication in the UNIX domain.

Special character

A graphically displayable character that is not a letter, a digit, or a blank.

Standard

Document containing definitions and stipulations for a precisely described function.

Status inquiry

A status inquiry returns basic information about the data display terminal (e.g. type of station, type and number of logically addressable character sets, number of colors) and a description of the connected peripherals (e.g. ID card reader, chip card terminal).

Subsystem

A self-contained system that can be installed and activated as an entity.

System message

A generic term for information of various types that the system can send to the user.

Task

The carrier for processes. IN BS2000 tasks are used, among other things, for execution of user jobs (e.g. batch job, interactive task) or the operation of applications (DCAM, UTM, TTX) (execution of all procedures specified between the BS2000 commands LOGON and LOGOFF).

TCP

Transmission Control Protocol

TCP is " protocol in the TCP/IP architecture. It handles the transport of data between two computers. TCP belongs approximately on Layer 4 of the OSI Reference Model, and thus corresponds to UDP. TCP is an extremely stable data-transfer protocol.

Terminal

A data display terminal, printer, or other terminal device

Terminal Support (TSU)

System programs which manage communication with data display terminals and printers above the transport layer (OSI Reference Model, Layers 1-4).

TIAM

Terminal Interactive Access Method (BS2000 software product)

UDP

User Datagram Protocol

UDP is a protocol in the TCP/IP architecture. It handles the transport of data between two computers. UDP belongs approximately on Layer 4 of the OSI Reference Model, and thus corresponds to TCP. UDP is a datagram protocol which supports broadcasting. Unlike TCP (protected end-to-end protocol) UDP merely provides assurance that the message is successfully sent. The user receives no indication of who received the data.

UTM

Universal Transaction Monitor (BS2000 software product)

Virtual terminal

Modeled terminal, the functions of which can be mapped to the physical characteristics of different physical terminals.

VTSU

Virtual Terminal Support

XHCS

Extended Host Code Support

Software product for the BS2000 operating system for support of extended character sets on 8-bit data terminals and for central support of code tables.

X.25

Interface for the transmission of data packets in public networks.

X.28 Terminal

Terminal featuring the X.28 interface. It uses start/stop transfer in order to exchange data with the PAD to which it is connected.

X.29

Process for exchanging control information and user data between a packet-data terminal and a PAD.

14.5 Abbreviations

BCAM	Basic Communication Access Method
CMX	Communications Manager SINIX
DCAM	Data Communication Access Method (BS2000 inquiry-and-transaction processing)
DSSM	Dynamic Subsystem Management
DDT	Data Display Terminal
ELFE	Error Logging File Evaluator
FHS	Format Handling System
FTOS	File Transfer OSI Support
IFG	Interactive Format Generator
ISLP	Interface Station Level Processor
ISO	International Organization for Standardization
NEA	TRANSDATA network architecture from Siemens
NEABV	NEA user service connection protocol
OSI	Open Systems Interconnection
OSS	OSI Services
PDN	Program system for data communication and network control
SNI	Siemens Nixdorf Informationssysteme
SPB	Station Level Parameter Block
TCP/IP	Transmission Control Protocol/Internet Protocol
TIAM	Terminal Interactive Access Method (BS2000 interactive or timesharing mode)
TSU	Terminal Support Unit
UDP	User Datagram Protocol
UTM	Universal Transaction Monitor
VTSU	Virtual Terminal Support
XHCS	Extended Host Code Support

Related publications

BS2000/OSD

Diagnostics Handbook

User Guide

BS2000/OSD

User Commands (SDF Format)

Volume 1, A-Q

User Guide

BS2000/OSD

User Commands (SDF Format)

Volume 2, R-Z

User Guide

Data Display Terminals (TRANSDATA)

Functional Description

Data Display Terminals (TRANSDATA)

Code Tables

9001 Printer (TRANSDATA)

User's Guide

9011-18/19 Printer (TRANSDATA)

User's Guide

9012 Printer (TRANSDATA)

User's Guide

9013 Printer (TRANSDATA)

User's Guide

9022 Printer (TRANSDATA)

User's Guide

DCAM (BS2000/OSD,TRANSDATA)

Program Interfaces

Reference Manual

DCAM (BS2000/OSD,TRANSDATA)

Macros

User Guide

DCAM (BS2000/OSD,TRANSDATA)

COBOL Calls

User Guide

SOCKETS-DE (BS2000)

Communication Method in BS2000

User Guide

CMX (BS2000)

Communication Method in BS2000

User Guide

BS2000/OSD

Utility Routines

User Guide

FHS (BS2000/OSD,TRANSDATA)

User Guide

IFG for FHS (BS2000/OSD,TRANSDATA)

User Guide

TRANSDATA, PDN-GA

Generating a Data Communication System (TRANSDATA)

User Guide

TRANSVIEW-NMA/-NMAE

TRANSVIEW-NTAC2/-NTAC2E

Network Management in BS2000

User Guide

TRANSVIEW-NMA (PDN)

TRANSVIEW-NMAE (PDN)

Network Management and Measurement Data Compilation in PDN

(TRANSDATA, PDN)

Functions and Facilities

User Guide

TRANSVIEW-NMA (PDN)

TRANSVIEW-NMAE (PDN)

Network Management and Measurement Data Compilation in PDN

(TRANSDATA, PDN)

Commands

User Guide

Network Access for Terminals (TRANSDATA)

User's Guide

BS2000/OSD

Executive Macros

User Guide

SDF (BS2000/OSD)

SDF Management

User Guide

BS2000/OSD

System User

Ready Reference

BS2000/OSD

System Exits

User Guide

BS2000/OSD

System Installation

User Guide

BS2000/OSD

System Administrator's Guide

User Guide

BS2000

System Administrator Commands (SDF Format)

Reference Manual

TIAM (TRANSDATA, BS2000/OSD)

User Guide

UTM, UTM-D (TRANSDATA, BS2000/OSD)**Programming Applications**

User Guide

UTM, UTM-D (TRANSDATA, BS2000/OSD)**Generating and Administering Applications**

User Guide

UTM (SINIX, BS2000/OSD)**Concepts and Functions**

User Guide

UTM, UTM-D (TRANSDATA, BS2000/OSD)**Messages, Debugging and Diagnostics**

User Guide

XHCS (BS2000/OSD, TRANSDATA)
Extended Host Code Support for BS2000/OSD
User Guide

X.25PORT, VTSU-X.29
(TRANSDATA, BS2000, PDN)
Packet Switching, User Guide

EHCS (BS2000)
Extended Host Code Support
Generation Manual

IHB
Internationalization Handbook
Guidelines for the Manufacture of International Products
Reference Manual

IHB Tables/IHB-Tabellen
Ergänzungen zum Internationalisierungshandbuch
Ready Reference - Tabellenbuch

IHB Checklist
Checklist for the Internationalization Handbook
Catalog of Questions

Ordering manuals

The manuals listed above and the corresponding order numbers can be found in the Siemens Nixdorf *List of Publications*. New publications are described in the *Druckschriften-Neuerscheinungen (New Publications)*.

You can arrange to have both of these sent to you regularly by having your name placed on the appropriate mailing list. Please apply to your local office, where you can also order the manuals.

Index

3270 terminal 13
3270 terminals, special characteristics 205
8-bit support 15
810 protocol 13

A

acknowledgment of terminal output 190
acoustic alarm, BELL 96, 159
activated extended character set 42, 116
advance half line down, PART-LINE-DOWN [COB] 138
advance half line up, PART-LINE-UP [COB] 137
alarm, acoustic 96, 159
area
 end protected EPA [ASS] 61
 start protected SPA [ASS] 61
ASF automatic sheet feeding [ASS] 71
automatic sheet feeding, ASF [ASS] 71
automatic tabulator
 AUTOMATIC-TABULATION [COB] 163
 AUTOTAB [ASS] 101
AUTOMATIC-TABULATION, automatic tabulator [COB] 163
AUTOTAB, automatic tabulator [ASS] 101

B

backspace
 BACKSPACE [COB] 132
 BS [ASS] 72
BASIC, basic information on terminal [ASS] 38
basic information on terminal, querying, BASIC [ASS] 38
basic terminal information, querying, VTSUBAS [COB] 111
BELL, acoustic alarm 96, 159
bit pattern, transparent mode 95, 158
blanked display, DARK-LAYOUT [COB] 131
BS, backspace 72

C

calling VTSU 17

CAP, continue actual position [ASS] 71

card reader 118

CCSNAME

 character set name [ASS] 95

 character set name [COB] 162

cctrc.h, logical control characters 181, 182

character set 25, 35, 109, 124

 shift out SO [ASS] 59

 switching SECOND-CHAR-SET [COB] 131

character set description, VTSUMONC [COB] 119

character set name

 CCSNAME [ASS] 95

 CCSNAME [COB] 162

characters

 emphasized EM [ASS] 58

 highlighting EMPH-LAYOUT [COB] 130

 maximum number of 70

chip card terminal 38, 95, 118, 158

choice of colors, COL [ASS] 62

CHS, loadable character set [ASS] 61

CL, logical end of record [ASS] 56

code conversion, CODETR [ASS] 96

code translation, CODE-TRANSLATION [COB] 160

CODE-TRANSLATION, code translation [COB] 160

CODETR, code conversion [ASS] 96

COL, choice of colors [ASS] 62

color monitor 34

color screen 123

colors, SELECT-COLOR [COB] 133

combined input/output modes 94, 158

configuration file, for special terminals 211

connected peripherals, querying 117

connection type 197

CONT-ACT-POS, continue at current carriage position [COB] 137

CONT-COL-N, position on column [COB] 137

CONT-LINE-N, continue at start of line [COB] 130

CONT-SAME-LINE, logical end-of-record [COB] 130

continue at start of line CONT-LINE-N [COB] 130

create local attributes, START-LOCAL-ATTR [COB] 132

CURPOS, cursor position [ASS] 100

cursor position
CURPOS [ASS] 100
CURSOR-POSITION [COB] 162
CURSOR-POSITION, cursor position [COB] 162

D

DAR, dark layout [ASS] 59
dark layout, DAR [ASS] 59
DARK-LAYOUT, blanked display [COB] 131
data display terminal description, VTSUMONC [COB] 119
data display terminals, support 253
data structure
cctrc.h [C] 181, 182
FORCTRC [FORTRAN] 165, 166
PL1CTRC [PL/I] 175, 176
TIAMCTRC [COB] 109, 125
VTSUBAS [COB] 109, 110, 111
vtsucb.h [C] 181, 184
VTSUCBC [COB] 109, 153
VTSUCBF [FORTRAN] 165, 170
VTSUCBP [PL/I] 175, 178
VTSUMONC [COB] 109, 110, 119
VTSUPERP [COB] 109, 110, 117
DC4, Escape [ASS] 73
DCSTA
query status information [ASS] 26
status inquiry [ASS] 25
define character spacing, HORIZ-MOVE-IND [COB] 138
define line spacing, VERT-MOVE-IND [COB] 138
DEL, delete [ASS] 72
delete, DEL [ASS] 72
delete character, DELETE-CHAR [COB] 132
DELETE-CHAR, delete character [COB] 132
device control characters 109
device-specific control characters 25
DIMENSION, screen format [COB] 134
DISPLAY-ATTR, set output attributes [COB] 134

E

EDOPT, static edit options [ASS] 31
ELFE 202
EM, emphasizing characters [ASS] 58
EMPH-LAYOUT, highlight characters [COB] 130
END-LOCAL-ATTR, reset local attributes [COB] 132
END-NEAR-LETT, stop letter quality [COB] 139

- end-of-line character, NEW-LINE [COB] 130
- end-of-page character
 - NEW-PAGE [COB] 130
 - NP [ASS] 56
- END-PROP-TYPE, proportional spacing off [COB] 138
- END-PROT-AREA, text characters, protected [COB] 132
- EPA, end protected area [ASS] 61
- ESC, Escape [ASS] 72
- Escape
 - DC4 [ASS] 73
 - ESC [ASS] 72
 - PHYS-DC4 [COB] 133
 - PHYS-ESC [COB] 133
- ESCAPE sequences 160
- evaluate logical control characters, NO-LOG-CHARS [COB] 159
- EXT DIM, physical screen dimension [ASS] 63
- EXT DIS, setting display attributes [ASS] 60
- EXT FLD, set field characteristics [ASS] 63
- EXT RPT, repeat character [ASS] 74
- EXT TRA, transparent output [ASS] 73
- EXTEND, extended standard code 95, 162
- extended character set 41, 115
- extended line mode 10, 55, 93, 157
- extended standard code, EXTEND 95, 162

- F**
- field, protected 93, 157
- field attributes 123
 - effect 84, 150
- field control characters 61
- FIELD-CHARACTERISTICS, set field attributes [COB] 134
- FORCTRC, logical control characters 165, 166
- format mode 94, 158
- format terminal 9
- free-text parameter 197
 - XOPCH 197
- function key code
 - FUNCTION-CODE [COB] 161
 - GETFC [ASS] 96
 - table 251
- FUNCTION-CODE, function key code [COB] 161

G

generation 15
GETFC, function key code [ASS] 96

H

handling
 input messages 14
 output messages 14
Hardcopy, HCOPY [ASS] 96
hardcopy, HARDCOPY [COB] 158
hardcopy device 118
hardcopy unit, symbolic name 38, 41
heterogeneous output 96, 161
HMI, character pitch [ASS] 68
HOM, homogeneous output [ASS] 96
HOM-OUTPUT, homogeneous output [COB] 161
homogeneous output, HOM [ASS] 96
homogenous output, HOM-OUTPUT [COB] 161
HORIZ-MOVE-IND, define character spacing [COB] 138
horizontal motion index, HMI [ASS] 68
horizontal position absolute HPA [ASS] 70
horizontal tabulation, HT [ASS] 73
horizontal tabulator, PHYS-HT [COB] 133
HPA
 horizontal position absolute [ASS] 70
 positioning on column [ASS] 57
HT, horizontal tabulation [ASS] 73

I

I/O parameters 109
ID card reader 38, 41, 99, 114
IHDR, message prefix [ASS] 97
information line
 "|" [COB] 157
 INFO [ASS] 94
INPUT-HEADER, message header [COB] 160
input/output modes, combined 158
input/output modes combined 94
input/output parameters 25
installation procedure for special terminals 210
internationalization 18
interpret logical control characters, NOLOG [ASS] 98

L

- left margin, LM [ASS] 69
- LEFT-MARGIN, set left margin [COB] 138
- LIDIM, logical terminal dimensions, querying 30
- line length, maximum 139
- line mode 10, 30, 55, 92, 157
- line terminal 10, 11
- lines, maximum number 70, 139
- LM, left margin [ASS] 69
- loadable character set
 - CHS [ASS] 61
 - START-CHARSET-N [COB] 132
- LOC, set local attributes [ASS] 65
- local attribute support, LOCAL-CHARS [COB] 159
- local attributes, LOCIN [ASS] 97
- local control characters 65
- LOCAL-CHARS, local attribute support [COB] 159
- LOCIN, local attribute input [ASS] 97
- logging a message 201
- logical control characters 25, 75ff, 109
 - cctrc.h [C] 182
 - FORCTRC [FORTRAN] 166
 - PL1CTRC [PL/I] 176
 - TIAMCTRC [COB] 125
 - VTCSET [ASS] 55ff
- logical display control characters 58, 141ff
- logical end of line, NL [ASS] 55
- logical end of record, CL [ASS] 56
- logical end-of-line character 43
- logical end-of-record, CONT-SAME-LINE [COB] 130
- logical line end character 116
- logical printer control characters 68
- logical record control characters 55
- logical terminal dimensions, query, LIDIM 30
- LOW, lowercase [ASS] 98
- LOWER-CHARS, lowercase letters [COB] 160
- lowercase, LOW [ASS] 98
- lowercase letters, LOWER-CHARS [COB] 160
- LOX, reset local attributes [ASS] 66

M

Macro, VTCSET 55

macro

DCSTA 25, 26

VTCSET 25

VTSUCB 25, 89

MAX-LINE-LEN, maximum line length [COB] 139

MAX-LINE-NUM, maximum number of lines [COB] 139

maximum line length

MAX-LINE-LEN [COB] 139

MLL [ASS] 70

maximum line number, MLN [ASS] 70

maximum number of characters 70

maximum number of lines 70

MAX-LINE-NUM [COB] 139

message header, INPUT-HEADER [COB] 160

message logging 201

message output, NOPOS [ASS] 99

message prefix, IHDR [ASS] 97

MIXED mode 94, 158

MLL, maximum line length [ASS] 70

MLN, maximum line number [ASS] 70

mode

"C" [COB] 158

"E" [COB] 157

"F" [COB] 158

"I" [COB] 157

"L" [COB] 157

"M" [COB] 158

"P" [COB] 158

"T" [COB] 158

CHIP [ASS] 95

EXTEND [ASS] 93

FORM [ASS] 94

format 94, 158

INFO [ASS] 94

LINE [ASS] 92

MIXED [ASS] 94

PHYS [ASS] 94

physical 94, 158

TRANS [ASS] 95

MONCS

character set description [ASS] 34

monitor description [ASS] 34

monitor format 123
monitoring functions 14
monochrome monitor 34
monochrome screen 123

N

NEABT protocol 20
NEABV protocol 20
near letter quality exit, NLX [ASS] 69
near letter quality start, NLQ [ASS] 69
NEW-LINE, end-of-line character [COB] 130
NEW-PAGE, end-of-page character [COB] 130
NL, end of line character [ASS] 55
NLQ, near letter quality start [ASS] 69
NLX, near letter quality exit [ASS] 69
NO-LOG-CHARS, evaluate logical control characters [COB] 159
NO-POSITION, position of output message [COB] 161
NOLOG, interpret logical control characters [ASS] 98
NOPOS, position of output message [ASS] 99
NOR, normal layout [ASS] 59
normal display, NORMAL-LAYOUT [COB] 131
normal layout, NOR [ASS] 59
NORMAL-CHAR-SET, character set, switching [COB] 131
NORMAL-LAYOUT, normal display [COB] 131
notational conventions 6
NP, end-of-page character [ASS] 56
NUM, numeric area [ASS] 61
number of characters 41, 114
number of colors 41, 114
number of lines 41, 114
numeric area, NUM [ASS] 61
numeric field, START-NUM-DATA [COB] 132

O

OFLOW, screen overflow [ASS] 33
OHDR, user-specific message prefix [ASS] 98
operating modes of terminals 189
OSI Reference Model 16, 19
output
 heterogeneous 96, 161
 homogeneous 96, 161
output attributes, effect 83, 149
OUTPUT-HEADER, user-specific message prefix [COB] 159
overflow control 188

P

PART-LINE-DOWN, advance half line down [COB] 138
PART-LINE-UP, advance half line up [COB] 137
partial line down, PLD [ASS] 68
partial line up, PLU [ASS] 68
partner characteristics 25
PDN free-text parameter 197
peripherals 25, 109
 querying 38
 querying, VTSUPERP [COB] 117
permanent hardcopy 43, 116
PERPH, query peripherals [ASS] 38
PHDIM, physical dimensions [ASS] 30
PHYS-DC4, Escape [COB] 133
PHYS-ESC, Escape [COB] 133
PHYS-HT, horizontal tabulator [COB] 133
PHYS-VT, vertical tabulator [COB] 133
physical dimensions, query, PHDIM [ASS] 30
physical dimensions of terminal, querying 30
physical mode 13, 94, 158
physical read mode 43, 116
 READ [ASS] 99
 READ-MODE [COB] 163
physical screen dimension, EXT DIM [ASS] 63
PL1CTRC, logical control characters [PL/I] 175, 176
plausibility check 15
PLD, partial line down [ASS] 68
PLU, partial line up [ASS] 68
PLUS, load and save program keys 219
position of output message, NO-POSITION [COB] 161
positioning on column CONT-COL-N [COB] 137
positioning on column HPA [ASS] 57
positioning on line field VPA [ASS] 57
printer
 definition 197
 return information 98, 159
 type 197
printers, supported 253
PRNAM, processor name 33
processor name, PRNAM 33
proportional spacing off, END-PROP-TYPE [COB] 138
proportional spacing on, START-PROP-TYPE [COB] 138
proportional type exit, PTX [ASS] 69
proportional type start, PTS [ASS] 69

protected field 93, 157
PTS, proportional type start [ASS] 69
PTX, proportional type exit [ASS] 69

Q

query basic information on terminal 38
query character set description 119
query character sets 34
 MONCS [ASS] 34
query data terminal description 119
query field attributes 34
query logical terminal dimensions 30
query monitor 34
query monitor description, MONCS [ASS] 34
query peripherals 117
 PERPH [ASS] 38
query physical dimensions 30
query physical type, TCHAR [ASS] 28
query processor name 33
query screen overflow control 33
query static edit options 31
query station name 33
query status information 26
query terminal characteristics 28
query terminal physical type 28
query virtual device type 31
querying basic terminal information 111
querying terminal status 38

R

READ, physical read mode [ASS] 99
read mode
 READ [ASS] 99
 READ-MODE [COB] 163
READ-MODE, physical read mode [COB] 163
repeat character
 EXT RPT [ASS] 74
 REPEAT-SYMBOL [COB] 134
REPEAT-SYMBOL, repeat character [COB] 134
reset local attributes
 END-LOCAL-ATTR [COB] 132
 LOX [ASS] 66
RETINF, printer return information [ASS] 98

S

- screen format 25, 109
 - DIMENSION [COB] 134
- screen overflow 187
 - OFLOW [ASS] 33
- screen overflow control, querying 33
- screen size 35
- SCREEN-UPDATE, update screen [COB] 163
- SECOND-CHAR-SET, switch character set [COB] 131
- SELECT-COLOR, colors [COB] 133
- set display attributes, EXT DIS [ASS] 60
- set field attributes FIELD-CHARACTERISTICS [COB] 134
- set field characteristics EXT FLD [ASS] 63
- set left margin, LEFT-MARGIN [COB] 138
- set local attributes, LOC [ASS] 65
- set output attributes, DISPLAY-ATTR [COB] 134
- setting operating parameters 191
- sheet ejection, SHEET-FEED-N [COB] 137
- sheet feed, SHEET-FEED-N [COB] 137
- SHEET-FEED-N
 - sheet ejection [COB] 137
 - sheet feed [COB] 137
- shift in to character set SI [ASS] 59
- SI, character set, shift in [ASS] 59
- SO, shift out to character set [ASS] 59
- software product VTSU-X.29 10
- SPA, start protected area [ASS] 61
- special data display terminals
 - Arabic/Persian 8-bit terminals 209
 - Euro 7-bit terminals 209
- special ESCAPE sequences 96
- special input
 - SPECIAL-INPUT [COB] 161
 - SPECIN [ASS] 99
- special terminals, Escape printers 209
- SPECIAL-INPUT, special input [COB] 161
- SPECIN, special input [ASS] 99
- start letter quality, START-NEAR-LETT [COB] 139
- START-CHARSET-N, loadable character set [COB] 132
- START-LOCAL-ATTR, create local attributes [COB] 132
- START-NEAR-LETT, start letter quality [COB] 139
- START-NUM-DATA, numeric field [COB] 132
- START-PROP-TYPE, proportional spacing on [COB] 138
- START-PROT-AREA, text characters, protected [COB] 131

- static edit options, querying, EDOPT [ASS] 31
- station name
 - querying 33
 - STNAM 33
- status information, querying 110
- status message 34, 123
- STNAM, station name 33
- stop letter quality, END-NEAR-LETT [COB] 139
- SUB, substitute [ASS] 72
- SUBSTITUTE, substitute character [COB] 133
- substitute character 43, 116
 - SUBSTITUTE [COB] 133
- support for data display terminals 253
- support for printers 253
- switching character set NORMAL-CHAR-SET [COB] 131

T

- tabulation
 - horizontal 73
 - vertical 73
- tabulator
 - automatic 101, 163
 - horizontal 133
 - vertical 133
- TCHAR, physical type [ASS] 28
- terminal, line 11
- terminal monitor 34
- terminal type 40, 113, 114
- text characters
 - protected START-PROT-AREA [COB] 131
 - unprotected END-PROT-AREA [COB] 132
- TIAMCTRC, logical control characters [COB] 109, 125
- transparent mode, bit pattern 95, 158
- transparent output
 - EXT TRA [ASS] 73
 - TRANSPARENT-OUTPUT [COB] 134
- TRANSPARENT-OUTPUT, transparent output [COB] 134
- type 9750 Data Display Terminals 213
- type of data terminal 109
- type of monitor 123
- type of terminal 25, 41

U

UPDATE, screen update [ASS] 100
update screen
 SCREEN-UPDATE [COB] 163
 UPDATE [ASS] 100
user-specific message prefix
 OHDR [ASS] 98
 OUTPUT-HEADER [COB] 159

V

VDT, virtual device type [ASS] 31
VERT-MOVE-IND, define line spacing [COB] 138
vertical motion index, VMI [ASS] 68
vertical position absolute VPA [ASS] 70
vertical tabulation, VT [ASS] 73
vertical tabulator, PHYS-VT [COB] 133
virtual device type, querying 31
virtual line 11
virtual line terminal 9
virtual terminal 9
VMI, vertical motion index [ASS] 68
VPA
 positioning on line field [ASS] 57
 vertical position absolute [ASS] 70
VT, vertical tabulation [ASS] 73
VTCSET, logical control characters [ASS] 25, 55
VTSU
 as subsystem 17
 calling 17
 in UTM applications 23
VTSU control block
 VTSUCB [ASS] 89
 vtsucb.h [C] 184
 VTSUCBC [COB] 153
 VTSUCBF [FORTRAN] 170
 VTSUCBP [PL/I] 178
VTSU in system environment 16
VTSU operating parameters 191
VTSU trace, VTSUTRAC 201
VTSUBAS 109, 110
 querying basic terminal information [COB] 111
VTSUCB, VTSU control block [ASS] 25, 89
vtsucb.h, VTSU control block [C] 181, 184
VTSUCBC, VTSU control block [COB] 109, 154

VTSUCBF, VTSU control block [FORTRAN] 165, 170

VTSUCBP, VTSU control block [PL/I] 175, 178

VTSUMONC 109, 110

 query character sets [COB] 119

 query data terminal description [COB] 119

VTSUPERP 109, 110

 query peripherals [COB] 117

VTSUTRAC, VTSU trace 201

X

XOPCH, free-text parameter 197

Contents

1	Preface	1
1.1	Target group	1
1.2	Summary of contents	2
1.3	Changes since the last version of the manual	3
1.4	Notational conventions	6
2	Introduction to VTSU	9
2.1	Virtual terminals	9
2.2	Functions of VTSU	14
2.3	VTSU in the system environment	16
2.3.1	Calling VTSU	17
2.3.2	VTSU as a subsystem	17
2.3.3	Internationalization	18
2.4	How VTSU interworks with other terminal-support implementations	19
2.4.1	The OSI Reference Model	19
2.4.2	Terminal support in BS2000	20
2.5	BS2000 and version independence	23
2.6	VTSU in UTM applications	23
3	ASSEMBLER interface	25
3.1	Status information: DCSTA	26
3.2	Logical control characters: VTCSET	55
3.3	VTSU control block: VTSUCB	89
4	COBOL interface	109
4.1	Status information	110
4.1.1	Basic information: VTSUBAS	111
4.1.2	Peripherals: VTSUPERP	117
4.1.3	Data display terminal and character sets: VTSUMONC	119
4.2	Logical control characters: TIAMCTRC	125
4.3	VTSU control block: VTSUCBC	153
5	FORTRAN interface	165
5.1	Logical control characters: FORCTRC	166
5.2	VTSU control block: VTSUCBF	170

6	PL/I interface	175
6.1	Logical control characters: PL1CTRC	176
6.2	VTSU control block: VTSUCBP	178
7	C interface	181
7.1	Logical control characters: cctrc.h	182
7.2	VTSU control block: vtsucb.h	184
8	Setting operating parameters	187
8.1	Screen overflow control	188
8.2	Operating modes of terminals	189
8.3	Acknowledgment of terminal output	190
8.4	Setting operating parameters during dynamic VTSU loading	191
9	VTSU trace	201
10	Special characteristics of 3270 terminals	205
11	Support for special data terminals	209
11.1	Installation procedure	210
11.2	Configuration file	211
12	Type 9750 Data Display Terminals	213
13	PLUS	219
13.1	Control statements	220
13.2	Calling the PLUS program	235
13.3	Device function mnemonics	240
13.4	General messages	245
13.5	Error messages	246
14	Appendix	251
14.1	Table of standardized function key codes	251
14.2	Support for terminals	253
14.3	Messages	254
14.4	Glossary	258
14.5	Abbreviations	264
	Related publications	265
	Index	269

VTSU V11.0A

User Guide

Target group

Users of the DCAM and TIAM access methods and of UTM, and also the system and network administrator.

Contents

VTSU (Virtual Terminal Support) is a software product of the BS2000 operating system. It implements a virtual line terminal. A virtual terminal permits programming that is independent of the physical characteristics of the terminal in question.

Edition: **April 1994**

File: **VTSU.PDF**

BS2000 is a registered trademark of Siemens Nixdorf Informationssysteme AG.

Copyright © Siemens Nixdorf Informationssysteme AG, 1994. All rights reserved.

The reproduction, transmission, translation or exploitation of this document or its contents is not permitted without express written authority. Offenders will be liable for damages.

Delivery subject to availability; right of technical modifications reserved.



Information on this document

On April 1, 2009, Fujitsu became the sole owner of Fujitsu Siemens Computers. This new subsidiary of Fujitsu has been renamed Fujitsu Technology Solutions.

This document from the document archive refers to a product version which was released a considerable time ago or which is no longer marketed.

Please note that all company references and copyrights in this document have been legally transferred to Fujitsu Technology Solutions.

Contact and support addresses will now be offered by Fujitsu Technology Solutions and have the format ...@ts.fujitsu.com.

The Internet pages of Fujitsu Technology Solutions are available at

[http://ts.fujitsu.com/...](http://ts.fujitsu.com/)

and the user documentation at <http://manuals.ts.fujitsu.com>.

Copyright Fujitsu Technology Solutions, 2009

Hinweise zum vorliegenden Dokument

Zum 1. April 2009 ist Fujitsu Siemens Computers in den alleinigen Besitz von Fujitsu übergegangen. Diese neue Tochtergesellschaft von Fujitsu trägt seitdem den Namen Fujitsu Technology Solutions.

Das vorliegende Dokument aus dem Dokumentenarchiv bezieht sich auf eine bereits vor längerer Zeit freigegebene oder nicht mehr im Vertrieb befindliche Produktversion.

Bitte beachten Sie, dass alle Firmenbezüge und Copyrights im vorliegenden Dokument rechtlich auf Fujitsu Technology Solutions übergegangen sind.

Kontakt- und Supportadressen werden nun von Fujitsu Technology Solutions angeboten und haben die Form ...@ts.fujitsu.com.

Die Internetseiten von Fujitsu Technology Solutions finden Sie unter

[http://de.ts.fujitsu.com/...](http://de.ts.fujitsu.com/), und unter <http://manuals.ts.fujitsu.com> finden Sie die Benutzerdokumentation.

Copyright Fujitsu Technology Solutions, 2009