INTEGRATED INFORMATION SUPPORT SYSTEM (IISS)
Volume VIII - User Interface Subsystem
Part 12 - Virtual Terminal Product Specification

S. Barker
Control Data Corporation
Integration Technology Services
2970 Presidential Drive
Fairborn, OH 45324-6209

September 1990

Final Report for Period 1 April 1987 - 31 December 1989

Approved for Public Release; Distribution is Unlimited

MANUFACTURING TECHNOLOGY DIRECTORATE
WRIGHT RESEARCH AND DEVELOPMENT CENTER
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433-6532
NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever, regardless whether or not the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data. It should not, therefore, be construed or implied by any person, persons, or organization that the Government is licensing or conveying any rights or permission to manufacture, use, or market any patented invention that may in any way be related thereto.

This technical report has been reviewed and is approved for publication.

DAVID L. JUDSON, Project Manager
WRDC/MTI
Wright-Patterson AFB, OH 45433-6533

FOR THE COMMANDER:

BRUCE A. RASMUSSEN, Chief
WRDC/MTI
Wright-Patterson AFB, OH 45433-6533

If your address has changed, if you wish to be removed from our mailing list, or if the addressee is no longer employed by your organization please notify WRDC/MTI, Wright-Patterson Air Force Base, OH 45433-6533 to help us maintain a current mailing list.

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.
**Title:** (See block 19) Virtual Ter

**Type of Report:** Final Report

**Date Covered:** 4/1/87 - 12/31/90

**Date of Report:** 1990 September 30

**Page Count:** 162

**Abstract:**

This specification establishes the detailed design of the Virtual Terminal computer program.

**Block 11:**

**INTEGRATED INFORMATION SUPPORT SYSTEM**

**Vol VIII - User Interface Subsystem**

**Part 12 - Virtual Terminal Product Specification**
FOREWORD

This technical report covers work performed under Air Force Contract F33600-87-C-0464, DAPro Project. This contract is sponsored by the Manufacturing Technology Directorate, Air Force Systems Command, Wright-Patterson Air Force Base, Ohio. It was administered under the technical direction of Mr. Bruce A. Rasmussen, Branch Chief, Integration Technology Division, Manufacturing Technology Directorate, through Mr. David L. Judson, Project Manager. The Prime Contractor was Integration Technology Services, Software Programs Division, of the Control Data Corporation, Dayton, Ohio, under the direction of Mr. W. A. Osborne. The DAPro Project Manager for Control Data Corporation was Mr. Jimmy P. Maxwell.

The DAPro project was created to continue the development, test, and demonstration of the Integrated Information Support System (IISS). The IISS technology work comprises enhancements to IISS software and the establishment and operation of IISS test bed hardware and communications for developers and users.

The following list names the Control Data Corporation subcontractors and their contributing activities:

<table>
<thead>
<tr>
<th>SUBCONTRACTOR</th>
<th>ROLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control Data Corporation</td>
<td>Responsible for the overall Data Model design development and implementation, IISS integration and test, and technology transfer of IISS.</td>
</tr>
<tr>
<td>D. Appleton Company</td>
<td>Responsible for providing software information services for the Common Data Model and IDEF1X integration methodology.</td>
</tr>
<tr>
<td>ONIEK</td>
<td>Responsible for defining and testing a representative integrated system base in Artificial Intelligence techniques to establish fitness for use.</td>
</tr>
<tr>
<td>Simpact Corporation</td>
<td>Responsible for Communication development.</td>
</tr>
<tr>
<td>Structural Dynamics</td>
<td>Responsible for User Interfaces, Virtual Terminal Interface, and Network Transaction Manager design, development, implementation, and support.</td>
</tr>
<tr>
<td>Research Corporation</td>
<td></td>
</tr>
<tr>
<td>Arizona State University</td>
<td>Responsible for test bed operations and support.</td>
</tr>
</tbody>
</table>
TABLE OF CONTENTS

SECTION 1.0 SCOPE ..................................... 1-1
1.1 Identification ..................................... 1-1
1.2 Functional Summary ............................... 1-1

SECTION 2.0 DOCUMENTS .................................. 2-1
2.1 Reference Documents ............................... 2-1
2.2 Terms and Abbreviations ......................... 2-2

SECTION 3.0 REQUIREMENTS ............................... 3-1
3.1 Structural Description ............................ 3-1
3.2 Functional Flow .................................. 3-1
3.3 Interfaces ........................................ 3-2
3.3.1 Physical Terminal .............................. 3-2
3.3.2 Application .................................... 3-2
3.3.2.1 VT Process as Master ...................... 3-3
3.3.2.2 VT Process as Slave ....................... 3-3
3.3.3 Terminal User .................................. 3-4
3.4 Program Interrupts ............................... 3-4
3.5 Timing and Sequencing Description .............. 3-4
3.6 Special Control Features ....................... 3-5
3.7 Storage Allocation ............................... 3-5
3.8 Object Code Creation ............................ 3-5
3.9 Adaptation Data .................................. 3-5
3.10 Detailed Design Description .................... 3-5
3.10.1 Main Program List ............................ 3-5
3.10.2 Module List .................................. 3-7
3.10.3 External Routines List ...................... 3-11
3.10.4 Include File List ............................ 3-14
3.10.5 Where Include File Used List ............... 3-16
3.10.6 Where External Routine Used List .......... 3-26
3.10.7 Main Program Parts List .................... 3-38
3.10.8 Module Documentation ....................... 3-42
3.10.9 Include File Descriptions ................... 3-122
3.10.10 Hierarchy Chart ............................ 3-134

SECTION 4.0 QUALITY ASSURANCE PROVISIONS ............. 4-1
4.1 Introduction and Definitions .................... 4-1
4.2 Computer Programming Test and Evaluation ...... 4-1
# LIST OF ILLUSTRATIONS

<table>
<thead>
<tr>
<th>Figure</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3-1</td>
<td>Virtual Terminal Structure</td>
<td>3-1</td>
</tr>
<tr>
<td>3-2</td>
<td>Virtual Terminal Data Flow</td>
<td>3-2</td>
</tr>
</tbody>
</table>
SECTION 1
SCOPE

1.1 Identification

This specification establishes the detailed design of a computer program identified as the Virtual Terminal, hereinafter referred to as the VT. The VT is one configuration item of the Integrated Information Support System (IISS) User Interface (UI).

1.2 Functional Summary

One of the objectives of the IISS testbed is to allow applications to be run from a wide variety of terminals. Instead of the application programmer having to worry about what commands to send to which type of terminal to perform what functions, he just uses commands for the Virtual Terminal. The Virtual Terminal is defined just like a real terminal; it has a set of functions which it can perform, a set of attributes that it supports, a set of commands for invoking the functions, and modes of operation.

The VT translates between the Virtual Terminal commands and commands for the particular type of terminal a user has. This process is not as simple as it sounds since no single terminal provides all of the functions and attributes that the Virtual Terminal does. Thus, the Virtual Terminal Interface must simulate missing functions with existing ones.

In addition to supporting real terminals, the VT also performs another function -- interfacing existing applications to the testbed. An existing application sends (and expects to receive) commands for a particular type of terminal. In the testbed these commands are intercepted and sent to the Virtual Terminal which then converts the commands into Virtual Terminal commands, just as if they had been entered from a real terminal. Of course, it also converts Virtual Terminal commands to the specific terminal commands the application expects to receive. The Virtual Terminal allows an application to be run from a terminal other than the one it was designed for.
2.2 Terms and Abbreviations

American Standard Code for Information Interchange: (ASCII), the character set defined by ANSI X3.4 and used by most computer vendors.

Application Interface: (AI), subset of the IISS User Interface that consists of the callable routines that are linked with applications that use the Form Processor or Virtual Terminal. The AI enables applications to be hosted on computers other than the host of the User Interface.

Application Process: (AP), a cohesive unit of software that can be initiated as a unit to perform some function or functions.

Attribute: field characteristic such as blinking, highlighted, black, etc. and various other combinations. Background attributes are defined for forms or windows only. Foreground attributes are defined for items. Attributes may be permanent, i.e., they remain the same unless changed by the application program, or they may be temporary, i.e., they remain in effect until the window is redisplayed.

Communication Services: allows on host interprocess communication and inter-host communication between the various Test Bed subsystems.

Computer Program Configuration Item: (CPCI), an aggregation of computer programs or any of their discrete portions, which satisfies an end-use function.

Device Drivers: (DD), software modules written to handle I/O for a specific kind of terminal. The modules map terminal specific commands and data to a neutral format. Device Drivers are part of the UI Virtual Terminal.

Extended Binary Coded Decimal Interchange Code: (EBCDIC), the character set used by a few computer vendors (notably IBM) instead of ASCII.

Field: two dimensional space on a terminal screen.
2.1 **Reference Documents**


Integrated Information Support System: (IISS), a test computing environment used to investigate, demonstrate and test the concepts of information management and information integration in the context of Aerospace Manufacturing. The IISS addresses the problems of integration of data resident on heterogeneous data bases supported by heterogeneous computers interconnected via a Local Area Network.

Logical Device: a conceptual device which to an application is indistinguishable from a physical device and is then mapped to part or all of a physical device.

Network Transaction Manager: (NTM), IISS subsystem that performs the coordination, communication and housekeeping functions required to integrate the Application Processes and System Services resident on the various hosts into a cohesive system.

Operating System: (OS), software supplied with a computer which allows it to supervise its own operations and manage access to hardware facilities such as memory and peripherals.

Physical Device: a hardware terminal.

User Interface: (UI), IISS subsystem that controls the user's terminal and interfaces with the rest of the system. The UI consists of two major subsystems: the User Interface Development System (UIDS) and the User Interface Management System (UIMS).

User Interface Management System: (UIMS), the runtime UI. It consists of the Form Processor, Virtual Terminal, Application Interface, the User Interface Services and the Text Editor.

User Interface Monitor: (UIM), part of the Form Processor that handles messaging between the NTM and the UI. It also provides authorization checks and initiates applications.

User Interface/Virtual Terminal Interface: (UI/VTI), another name for the User Interface.
Virtual Terminal: (VT), subset of the IISS User Interface that performs the interfacing between different terminals and the UI. This is done by defining a specific set of terminal features and protocols which must be supported by the UI software which constitutes the virtual terminal definition. Specific terminals are then mapped against the virtual terminal software by specific software modules written for each type of real terminal supported.

Virtual Terminal Interface: (VTI), the callable interface to the VT.

Window: dynamic area of a terminal screen on which predefined forms may be placed at run time.

Window Manager: a facility which allows the following to be manipulated: size and location of windows, the device on which an application is running, the position of a form within a window. It is part of the Form Processor.
SECTION 3
REQUIREMENTS

3.1 Structural Description

Figure 3-1 describes the structure of the Virtual Terminal. The Virtual Terminal consists of some routines that are linked with the application that uses it (VT Application Monitor) as well as a process that performs monitoring, window management activities and translation of VT commands into commands for a specific device.

![Figure 3-1 Virtual Terminal Structure]

3.2 Functional Flow

The Virtual Terminal can be used in two different modes: master and slave. Currently, the master mode is used for interactive devices and the slave mode is used for batch devices such as printers.

Figure 3-2 is a data flow for the Virtual Terminal in master and slave mode.
3.3 **Interfaces**

3.3.1 **Physical Terminal**

The interface to a physical terminal is a function of the host Operating System (OS) and is highly system dependent. When somewhat portable functions are recognized, they are isolated into system independent routines thus making as much of the code portable as is possible.

3.3.2 **Application**

Applications can use the VT Application Monitor to communicate with the VT process. The types of messages that are sent to and from the VT process are dependent upon the mode (master or slave) of the VT process and are described in the next two sections. Device data messages (type DD) contain VT commands and data. The VT commands are described in Appendix A of the Virtual Terminal Development Specification (DS 620344300).
### 3.3.2.1 VT Process as Master

**RECEIVE MESSAGES**

<table>
<thead>
<tr>
<th>Message Type</th>
<th>Action Taken/Buffer</th>
</tr>
</thead>
<tbody>
<tr>
<td>SD</td>
<td>Terminates Device Driver.</td>
</tr>
<tr>
<td>DD</td>
<td>Send Device Data to be output to Virtual Terminal.</td>
</tr>
<tr>
<td>DQ</td>
<td>Send Device Data to be output to Virtual Terminal and request for acknowledgement.</td>
</tr>
</tbody>
</table>

**SEND MESSAGES**

<table>
<thead>
<tr>
<th>Message Type</th>
<th>Action Taken/Buffer</th>
</tr>
</thead>
<tbody>
<tr>
<td>DE</td>
<td>Informs UIM that Master Device Driver is entering the UIM system control.</td>
</tr>
<tr>
<td>DD</td>
<td>Send Device Data which was input to Virtual Terminal.</td>
</tr>
</tbody>
</table>

Section 3.2.2.1.2 of the Virtual Terminal Development Specification contains a detailed description of these messages.

### 3.3.2.2 VT Process as Slave

**RECEIVE MESSAGES**

<table>
<thead>
<tr>
<th>Message Type</th>
<th>Action Taken/Buffer</th>
</tr>
</thead>
<tbody>
<tr>
<td>DE</td>
<td>Starts up Slave Device Driver.</td>
</tr>
<tr>
<td>SD</td>
<td>Terminates Device Driver.</td>
</tr>
<tr>
<td>DD</td>
<td>Send Device Data to be output to Virtual Terminal.</td>
</tr>
</tbody>
</table>
SEND MESSAGES

<table>
<thead>
<tr>
<th>Message Type</th>
<th>Action Taken/Buffer</th>
</tr>
</thead>
<tbody>
<tr>
<td>DI</td>
<td>Informs UIM that slave driver is alive and should be initialized with proper size.</td>
</tr>
<tr>
<td>DD</td>
<td>Send Device Data which was input to Virtual Terminal.</td>
</tr>
</tbody>
</table>

Section 3.2.2.2.2 of the Virtual Terminal Development Specification contains a detailed description of these messages.

3.3.3 Terminal User

When the VT process is the master it is started by the terminal user. It has a number of parameters that can be used if scripting is required.

- `w <scripting file name>` - write script file
- `r <scripting file name>` - read script file
- `s <save file name>` - saves output from session

These arguments are optional. The user can either create a script file, read a script file, or do neither. The user can also save or not save the output from a session.

3.4 Program Interrupts

Attention interrupts received from the terminal (CNTL/C, break) cause the VT process to terminate by calling the NTM routine TRMNAT.

3.5 Timing and Sequencing Description

The Monitor processes two types of input: keyboard characters and NTM messages. First, a check is made for available keyboard characters. As long as characters are available, they are processed. When no characters are available, a check is made for NTM messages. If a message is found, it is processed and the Monitor again checks for keyboard characters. If no message is available, the Monitor waits for approximately .1 seconds before again checking for keyboard characters.
3.6 **Special Control Features**

The detailed design of the VT does not include any special control features as defined in the ICAM Documentation Standards manual.

3.7 **Storage Allocation**

The executable sizes for the device driver routines for each supported terminal are:

- ADM3A 172 blocks
- CI600 165 blocks
- IBM3270 not available
- PRINTR 153 blocks
- PW3270 not available
- VIP 174 blocks
- VT100 176 blocks
- VT100W 176 blocks
- VTMIN 175 blocks

3.8 **Object Code Creation**

The VT routines were compiled using a C compiler developed by Interactive Software under VAX/VMS.

3.9 **Adaptation Data**

The C source modules for the VT can be compiled using any UNIX version 7 compatible C compiler. All routines beginning with 'TRM' are device dependent, and the routine TERMIO.C is VAX specific.

3.10 **Detailed Design Description**

3.10.1 **Main Program List**

The following is a list of all "Main Programs" which are modules that are not called by any other module being documented here. These modules are either program entry points or, if they are hooked into another set of programs via subroutine calls, they are the points the external programs can call and therefore enter through. To differentiate between the two types of entry points, look at the individual Module Documentation (section 3.10.8) and look at Module Type for each of the Main Program modules listed. Note whether the routine is a Program, Subroutine, or Function. If it is a Program, it is truly a main program entry point. If not, then it is merely called by other programs not being documented here.
VIRTUAL TERMINAL Main Program List

<table>
<thead>
<tr>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>DRIVER/MAIN</td>
<td>MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER</td>
</tr>
</tbody>
</table>
3.10.2 Module List

The following is a list of all the modules being documented here along with their purpose. Each module has a unique name, no matter what language it was written in.
## VIRTUAL TERMINAL Module List

<table>
<thead>
<tr>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABSPOS</td>
<td>ABSOLUTIZE CURSOR POSITION OF FIELD</td>
</tr>
<tr>
<td>BLDMSG</td>
<td>BUILD MESSAGE</td>
</tr>
<tr>
<td>BLDMSG/BLDBUF</td>
<td>BUILD BUFFER</td>
</tr>
<tr>
<td>BLDMSG/REDOFF</td>
<td>READ FLAG TURNED OFF</td>
</tr>
<tr>
<td>BVTIDS</td>
<td>BUILD VTI DATA STRUCTURE</td>
</tr>
<tr>
<td>BVTIDS/BVTIFM</td>
<td>BUILD VTI FIELD MAP</td>
</tr>
<tr>
<td>BVTIDS/CLRFLG</td>
<td>CLEAR FLAGS</td>
</tr>
<tr>
<td>BVTIDS/CVTIFM</td>
<td>CLEAR VTI FIELD MAP</td>
</tr>
<tr>
<td>BVTIDS/INSFLD</td>
<td>INSERT FIELD</td>
</tr>
<tr>
<td>BVTIDS/RVTIFM</td>
<td>REBUILD VTI FIELD MAP</td>
</tr>
<tr>
<td>CLRMOD</td>
<td>CLEAR MODIFY FLAGS</td>
</tr>
<tr>
<td>DEFFLD</td>
<td>DEFINE FIELD</td>
</tr>
<tr>
<td>DEFWND</td>
<td>DEFINE WINDOW</td>
</tr>
<tr>
<td>DOSCR/ERASE</td>
<td>ERASE PART OF SCREEN</td>
</tr>
<tr>
<td>DOSCR/HSCR</td>
<td>HORIZONTAL SCROLL</td>
</tr>
<tr>
<td>DOSCR/VSCR</td>
<td>VERTICAL SCROLL</td>
</tr>
<tr>
<td>DOSCREEN</td>
<td>DO COMMAND TO INTERNAL SCREEN</td>
</tr>
<tr>
<td>DRIVER/MAIN</td>
<td>MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER</td>
</tr>
<tr>
<td>ERAWND</td>
<td>ERASE WINDOW</td>
</tr>
<tr>
<td>FATAL</td>
<td>REPORT FATAL ERROR</td>
</tr>
</tbody>
</table>
VIRTUAL TERMINAL Module List

<table>
<thead>
<tr>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>FNDWND</td>
<td>FIND WINDOW</td>
</tr>
<tr>
<td>GETVT</td>
<td>GET DATA FROM VIRTUAL TERMINAL</td>
</tr>
<tr>
<td>GVTTICMD</td>
<td>GET VIRTUAL TERMINAL INTERFACE COMMAND</td>
</tr>
<tr>
<td>INTVT</td>
<td>INITIALIZE VIRTUAL TERMINAL</td>
</tr>
<tr>
<td>INVIS</td>
<td>CHECK FOR INVISIBILITY</td>
</tr>
<tr>
<td>PCHVTI</td>
<td>PUT SCREEN CHARACTERS TO VTI DATA STRUCTURE</td>
</tr>
<tr>
<td>PRCMD</td>
<td>PROCESS COMMAND</td>
</tr>
<tr>
<td>PUTVT</td>
<td>PUT DATA TO VIRTUAL TERMINAL</td>
</tr>
<tr>
<td>PVTICMD</td>
<td>PUT VTI COMMAND</td>
</tr>
<tr>
<td>PVTICMD/PUTNUM</td>
<td>PUT NUMBER</td>
</tr>
<tr>
<td>REFRESH</td>
<td>REFRESH TERMINAL</td>
</tr>
<tr>
<td>REFTERM</td>
<td>REFRESH TERMINAL</td>
</tr>
<tr>
<td>RMVWND</td>
<td>REMOVE WINDOW</td>
</tr>
<tr>
<td>SLINEND</td>
<td>FIND SCREEN LINE END</td>
</tr>
<tr>
<td>STFMTF</td>
<td>SET FORMAT FLAG FOR ALL CHILDREN WINDOWS AND FIELDS</td>
</tr>
<tr>
<td>STRDPN</td>
<td>SET READ PENDING FLAGS</td>
</tr>
<tr>
<td>STRDPN/STFDRD</td>
<td>SET FIELD READ PENDING</td>
</tr>
<tr>
<td>SWHPRC</td>
<td>SET WINDOW PRECEDENCE</td>
</tr>
<tr>
<td>TPUTNUM</td>
<td>TERMINAL PUT NUMBER</td>
</tr>
<tr>
<td>TPMTS</td>
<td>TERMINAL PUT STRING</td>
</tr>
</tbody>
</table>
**VIRTUAL TERMINAL Module List**

<table>
<thead>
<tr>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>TRMCHK</td>
<td>TERMINAL CHECK</td>
</tr>
<tr>
<td>TRMEND</td>
<td>TERMINAL END</td>
</tr>
<tr>
<td>TRMFLS</td>
<td>TERMINAL FLUSH</td>
</tr>
<tr>
<td>TRMGET</td>
<td>TERMINAL GET</td>
</tr>
<tr>
<td>TRMINI</td>
<td>TERMINAL INITIALIZE</td>
</tr>
<tr>
<td>TRMPUT</td>
<td>TERMINAL PUT</td>
</tr>
<tr>
<td>TRMVT</td>
<td>TERMINATE VIRTUAL TERMINAL</td>
</tr>
<tr>
<td>TVTPRC</td>
<td>TERMINATE VTI PROCESS</td>
</tr>
<tr>
<td>VT100/MOVCUR</td>
<td>MOVE CURSOR (INTERNAL)</td>
</tr>
<tr>
<td>VT100/SETATR</td>
<td>SET ATTRIBUTES (INTERNAL)</td>
</tr>
</tbody>
</table>
3.10.3 **External Routines List**

The following is a list of all routines or functions not documented here that are called by modules that are documented here. The first caller, in alphabetical order, is listed as well. The specification in which any module is documented may be found in the Module Documentation Index (Document Number CM 620100001). See section 3.10.6 for a list of the modules that call each of these external routines.
VIRTUAL TERMINAL External Routines List

<table>
<thead>
<tr>
<th>Module Name</th>
<th>First User</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLDCMD</td>
<td>BVTIDS</td>
</tr>
<tr>
<td>CABIT</td>
<td>TRMPUT</td>
</tr>
<tr>
<td>CALLOC</td>
<td>DEFWND</td>
</tr>
<tr>
<td>CBIT</td>
<td>DOSCR/ERASE</td>
</tr>
<tr>
<td>COL</td>
<td>TRMPUT</td>
</tr>
<tr>
<td>CSTR</td>
<td>INTVT</td>
</tr>
<tr>
<td>DELAY</td>
<td>DRIVER/MAP</td>
</tr>
<tr>
<td>EXIT</td>
<td>TVTPRC</td>
</tr>
<tr>
<td>FCLOSE</td>
<td>DRIVER/MAP</td>
</tr>
<tr>
<td>FFBDA</td>
<td>TRMPUT</td>
</tr>
<tr>
<td>FFBSA</td>
<td>VT100/SETATR</td>
</tr>
<tr>
<td>FFBSB</td>
<td>DOSSCREEN</td>
</tr>
<tr>
<td>FIX</td>
<td>DOSSCREEN</td>
</tr>
<tr>
<td>FLOOR</td>
<td>DOSSCREEN</td>
</tr>
<tr>
<td>FOPEN</td>
<td>DRIVER/MAP</td>
</tr>
<tr>
<td>FPRINTF</td>
<td>DRIVER/MAP</td>
</tr>
<tr>
<td>FREE</td>
<td>BVTIDS/INSFLD</td>
</tr>
<tr>
<td>FSEARCH</td>
<td>DRIVER/MAP</td>
</tr>
<tr>
<td>FWRITE</td>
<td>DRIVER/MAP</td>
</tr>
<tr>
<td>GETCHAR</td>
<td>TRMPUT</td>
</tr>
<tr>
<td>INITEX</td>
<td>DRIVER/MAP</td>
</tr>
<tr>
<td>ISDIGIT</td>
<td>TRMGET</td>
</tr>
<tr>
<td>IPRINTF</td>
<td>GVTICMD</td>
</tr>
<tr>
<td>LIMIT</td>
<td>DOSSCREEN</td>
</tr>
<tr>
<td>MALLOC</td>
<td>BVTIDS/INSFLD</td>
</tr>
<tr>
<td>MAX</td>
<td>PCHVTI</td>
</tr>
<tr>
<td>MEMCMP</td>
<td>DRIVER/MAP</td>
</tr>
<tr>
<td>MEMCPY</td>
<td>DRIVER/MAP</td>
</tr>
<tr>
<td>MEMSET</td>
<td>DEFFLD</td>
</tr>
<tr>
<td>MIN</td>
<td>DOSCR/VSCR</td>
</tr>
<tr>
<td>NSEND</td>
<td>DRIVER/MAP</td>
</tr>
<tr>
<td>POS</td>
<td>TRMPUT</td>
</tr>
<tr>
<td>PRINTF</td>
<td>FATAL</td>
</tr>
<tr>
<td>PRNEND</td>
<td>TRMEND</td>
</tr>
<tr>
<td>PRNFLS</td>
<td>TRMPUT</td>
</tr>
<tr>
<td>PRINI</td>
<td>TRMINI</td>
</tr>
<tr>
<td>PRNPUT</td>
<td>TRMPUT</td>
</tr>
<tr>
<td>PUTC</td>
<td>DRIVER/MAP</td>
</tr>
<tr>
<td>RCV</td>
<td>DRIVER/MAP</td>
</tr>
<tr>
<td>ROW</td>
<td>REFRESH</td>
</tr>
<tr>
<td>SBIT</td>
<td>DOSCR/VSCR</td>
</tr>
<tr>
<td>Module Name</td>
<td>First User</td>
</tr>
<tr>
<td>--------------</td>
<td>---------------------</td>
</tr>
<tr>
<td>SIGNAL</td>
<td>DRIVER/MAIN</td>
</tr>
<tr>
<td>SPRINTF</td>
<td>DRIVER/MAIN</td>
</tr>
<tr>
<td>STRASN</td>
<td>BVTIDS/CVTIFM</td>
</tr>
<tr>
<td>STRCAT</td>
<td>DRIVER/MAIN</td>
</tr>
<tr>
<td>STRCPY</td>
<td>DRIVER/MAIN</td>
</tr>
<tr>
<td>STRLEN</td>
<td>BLDMSG/BLDBUF</td>
</tr>
<tr>
<td>TBIT</td>
<td>DOSCR/ERASE</td>
</tr>
<tr>
<td>TBOPEN</td>
<td>TRMINI</td>
</tr>
<tr>
<td>TCHECK</td>
<td>TRMCHK</td>
</tr>
<tr>
<td>TCLOSE</td>
<td>TRMEND</td>
</tr>
<tr>
<td>TFLUSH</td>
<td>TRMFLS</td>
</tr>
<tr>
<td>TGETC</td>
<td>TRMGET</td>
</tr>
<tr>
<td>TLOWER</td>
<td>DRIVER/MAIN</td>
</tr>
<tr>
<td>TPURGE</td>
<td>TRMGET</td>
</tr>
<tr>
<td>TPUTC</td>
<td>VT100/SETATR</td>
</tr>
<tr>
<td>TRMNAT</td>
<td>DRIVER/MAIN</td>
</tr>
<tr>
<td>ZERO</td>
<td>DOSCREEN</td>
</tr>
</tbody>
</table>
3.10.4 Include File List

The following is a list of all include files called in by modules being documented here. Each include file has a unique name regardless of the language being used. The purpose of each include file is listed as well. A more complete description of each include file is given in section 3.10.9. The purpose listed is the one that is in the source code of the include file.

A purpose of "**** PURPOSE NOT FOUND BY STRIPPER ****" indicates that a purpose statement was not written into the include file itself. The most common reason for this is that the include file comes from system libraries that were not developed by the project, such as 'C' libraries that are provided with the 'C' compiler.

See section 3.10.6 for a set of lists which show all the modules which call in each of these include files.
### VIRTUAL TERMINAL Include File List

<table>
<thead>
<tr>
<th>File Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>BITS</td>
<td>INCLUDE FILE FOR BIT MANIPULATION ROUTINES</td>
</tr>
<tr>
<td>CI600.C&quot;</td>
<td>**** PURPOSE NOT FOUND BY STRIPPER ****</td>
</tr>
<tr>
<td>CTLCHR</td>
<td>CONTROL CHARACTERS</td>
</tr>
<tr>
<td>CTYPE</td>
<td>**** PURPOSE NOT FOUND BY STRIPPER ****</td>
</tr>
<tr>
<td>DEVICE</td>
<td>PHYSICAL DEVICE DATA STRUCTURE</td>
</tr>
<tr>
<td>DEVINI</td>
<td>DEVICE INITIALIZATIONS</td>
</tr>
<tr>
<td>FUNCTS</td>
<td>FUNCTION DEFINITIONS</td>
</tr>
<tr>
<td>NTM</td>
<td>NTM INTERFACE INCLUDE FILE</td>
</tr>
<tr>
<td>SCREEN</td>
<td>INTERNAL SCREEN DEFINITIONS</td>
</tr>
<tr>
<td>SIGNAL</td>
<td>**** PURPOSE NOT FOUND BY STRIPPER ****</td>
</tr>
<tr>
<td>STDIO</td>
<td>**** PURPOSE NOT FOUND BY STRIPPER ****</td>
</tr>
<tr>
<td>STDTyp</td>
<td>STANDARD TYPE DEFINITIONS</td>
</tr>
<tr>
<td>TERMIO</td>
<td>TRANSPARENT TERMINAL I/O DEFINITIONS</td>
</tr>
<tr>
<td>TRMRTN</td>
<td>TERMINAL (DEVICE DRIVER) ROUTINES</td>
</tr>
</tbody>
</table>
3.10.5 Where Include File Used List

The following lists each include file from 3.10.4 and all the modules documented in this specification which include them. The purpose of each module is listed as well.
VIRTUAL TERMINAL Where-include-file-used List

<table>
<thead>
<tr>
<th>Include File</th>
<th>Module Name</th>
<th>Module Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>BITS</td>
<td>BLDMSG</td>
<td>BUILD MESSAGE</td>
</tr>
<tr>
<td></td>
<td>BLDMSG/BL</td>
<td>BUILD BUFFER</td>
</tr>
<tr>
<td></td>
<td>BLDMSG/RE</td>
<td>READ FLAG TURNED OFF</td>
</tr>
<tr>
<td></td>
<td>BVTIDS</td>
<td>BUILD VTI DATA STRUCTURE</td>
</tr>
<tr>
<td></td>
<td>BVTIDS/BV</td>
<td>BUILD VTI FIELD MAP</td>
</tr>
<tr>
<td></td>
<td>BVTIDS/CL</td>
<td>CLEAR FLAGS</td>
</tr>
<tr>
<td></td>
<td>BVTIDS/CV</td>
<td>CLEAR VTI FIELD MAP</td>
</tr>
<tr>
<td></td>
<td>BVTIDS/IN</td>
<td>INSERT FIELD</td>
</tr>
<tr>
<td></td>
<td>BVTIDS/RV</td>
<td>REBUILD VTI FIELD MAP</td>
</tr>
<tr>
<td></td>
<td>CLRMOD</td>
<td>CLEAR MODIFY FLAGS</td>
</tr>
<tr>
<td></td>
<td>DEFFLD</td>
<td>DEFINE FIELD</td>
</tr>
<tr>
<td></td>
<td>DEFWND</td>
<td>DEFINE WINDOW</td>
</tr>
<tr>
<td></td>
<td>DOSCR/ERA</td>
<td>ERASE PART OF SCREEN</td>
</tr>
<tr>
<td></td>
<td>DOSCR/HSC</td>
<td>HORIZONTAL SCROLL</td>
</tr>
<tr>
<td></td>
<td>DOSCR/VSC</td>
<td>VERTICAL SCROLL</td>
</tr>
<tr>
<td></td>
<td>DOSCREEN</td>
<td>DO COMMAND TO INTERNAL SCREEN</td>
</tr>
<tr>
<td></td>
<td>DRIVER/MA</td>
<td>MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER</td>
</tr>
<tr>
<td></td>
<td>ERAWND</td>
<td>ERASE WINDOW</td>
</tr>
<tr>
<td></td>
<td>GETVT</td>
<td>GET DATA FROM VIRTUAL TERMINAL</td>
</tr>
<tr>
<td></td>
<td>GVTICMD</td>
<td>GET VIRTUAL TERMINAL INTERFACE COMMAND</td>
</tr>
<tr>
<td></td>
<td>INTVT</td>
<td>INITIALIZE VIRTUAL TERMINAL</td>
</tr>
<tr>
<td></td>
<td>INVIS</td>
<td>CHECK FOR INVISIBILITY</td>
</tr>
<tr>
<td></td>
<td>PCHVTI</td>
<td>PUT SCREEN CHARACTERS TO VTI DATA STRUCTURE</td>
</tr>
<tr>
<td></td>
<td>PRCCMDS</td>
<td>PROCESS COMMAND</td>
</tr>
<tr>
<td></td>
<td>PUTVT</td>
<td>PUT DATA TO VIRTUAL TERMINAL</td>
</tr>
<tr>
<td></td>
<td>PVTICMD</td>
<td>PUT VTI COMMAND</td>
</tr>
<tr>
<td></td>
<td>PVTICMD/P</td>
<td>PUT NUMBER</td>
</tr>
<tr>
<td></td>
<td>REFRESH</td>
<td>REFRESH TERMINAL</td>
</tr>
<tr>
<td></td>
<td>REFTERM</td>
<td>REFRESH TERMINAL</td>
</tr>
<tr>
<td></td>
<td>RMVWND</td>
<td>REMOVE WINDOW</td>
</tr>
<tr>
<td></td>
<td>SLINEND</td>
<td>FIND SCREEN LINE END</td>
</tr>
<tr>
<td></td>
<td>STFMTF</td>
<td>SET FORMAT FLAG FOR ALL CHILDREN WINDOWS AND FIELDS</td>
</tr>
<tr>
<td></td>
<td>SWNPRC</td>
<td>SET WINDOW PRECEDENCE</td>
</tr>
<tr>
<td></td>
<td>TRMCHK</td>
<td>TERMINAL CHECK</td>
</tr>
<tr>
<td></td>
<td>TRMEND</td>
<td>TERMINAL END</td>
</tr>
<tr>
<td></td>
<td>TRMFLS</td>
<td>TERMINAL FLUSH</td>
</tr>
</tbody>
</table>
VIRTUAL TERMINAL Where-include-file-used List

<table>
<thead>
<tr>
<th>Include File</th>
<th>Module Name</th>
<th>Module Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>TRMGET</td>
<td>TERMINAL GET</td>
<td></td>
</tr>
<tr>
<td>TRMINI</td>
<td>TERMINAL INITIALIZE</td>
<td></td>
</tr>
<tr>
<td>TRMPUT</td>
<td>TERMINAL PUT</td>
<td></td>
</tr>
<tr>
<td>TRMVT</td>
<td>TERMINATE VIRTUAL TERMINAL</td>
<td></td>
</tr>
<tr>
<td>VT100/MOV</td>
<td>MOVE CURSOR (INTERNAL)</td>
<td></td>
</tr>
<tr>
<td>VT100/SET</td>
<td>SET ATTRIBUTES (INTERNAL)</td>
<td></td>
</tr>
</tbody>
</table>

CI600.C"

<table>
<thead>
<tr>
<th>Module Name</th>
<th>Module Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>TRMCHK</td>
<td>TERMINAL CHECK</td>
</tr>
<tr>
<td>TRMEND</td>
<td>TERMINAL END</td>
</tr>
<tr>
<td>TRMFLS</td>
<td>TERMINAL FLUSH</td>
</tr>
<tr>
<td>TRMGET</td>
<td>TERMINAL GET</td>
</tr>
<tr>
<td>TRMINI</td>
<td>TERMINAL INITIALIZE</td>
</tr>
<tr>
<td>TRMPUT</td>
<td>TERMINAL PUT</td>
</tr>
<tr>
<td>VT100/MOV</td>
<td>MOVE CURSOR (INTERNAL)</td>
</tr>
<tr>
<td>VT100/SET</td>
<td>SET ATTRIBUTES (INTERNAL)</td>
</tr>
</tbody>
</table>

CTLCHR

<table>
<thead>
<tr>
<th>Build Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLDMSG</td>
</tr>
<tr>
<td>BLDMSG/BL</td>
</tr>
<tr>
<td>BLDMSG/RE</td>
</tr>
<tr>
<td>DRIVER/MA</td>
</tr>
<tr>
<td>GVTICMD</td>
</tr>
<tr>
<td>INTVT</td>
</tr>
<tr>
<td>PVTICMD</td>
</tr>
<tr>
<td>PVTICMD/P</td>
</tr>
<tr>
<td>TRMVT</td>
</tr>
</tbody>
</table>

CTYPE

<table>
<thead>
<tr>
<th>Define Field</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEFFLD</td>
</tr>
</tbody>
</table>
VIRTUAL TERMINAL Where-include-file-used List

<table>
<thead>
<tr>
<th>Include File</th>
<th>Module Name</th>
<th>Module Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>GVTICMD</td>
<td>GET VIRTUAL TERMINAL INTERFACE COMMAND</td>
</tr>
<tr>
<td></td>
<td>TRMCHK</td>
<td>TERMINAL CHECK</td>
</tr>
<tr>
<td></td>
<td>TRMEND</td>
<td>TERMINAL END</td>
</tr>
<tr>
<td></td>
<td>TRMFLS</td>
<td>TERMINAL FLUSH</td>
</tr>
<tr>
<td></td>
<td>TRMGET</td>
<td>TERMINAL GET</td>
</tr>
<tr>
<td></td>
<td>TRMINI</td>
<td>TERMINAL INITIALIZE</td>
</tr>
<tr>
<td></td>
<td>TRMPUT</td>
<td>TERMINAL PUT</td>
</tr>
<tr>
<td></td>
<td>VT100/MOV</td>
<td>MOVE CURSOR (INTERNAL)</td>
</tr>
<tr>
<td></td>
<td>VT100/SET</td>
<td>SET ATTRIBUTES (INTERNAL)</td>
</tr>
</tbody>
</table>

DEVICE

<table>
<thead>
<tr>
<th>DEVICE</th>
<th>ABSPOS</th>
<th>ABSOLUTIZE CURSOR POSITION OF FIELD</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLDMSG</td>
<td>BUILD MESSAGE</td>
<td></td>
</tr>
<tr>
<td>BLDMSG/BL</td>
<td>BUILD BUFFER</td>
<td></td>
</tr>
<tr>
<td>BLDMSG/RE</td>
<td>READ FLAG TURNED OFF</td>
<td></td>
</tr>
<tr>
<td>BVTIDS</td>
<td>BUILD VTI DATA STRUCTURE</td>
<td></td>
</tr>
<tr>
<td>BVTIDS/BV</td>
<td>BUILD VTI FIELD MAP</td>
<td></td>
</tr>
<tr>
<td>BVTIDS/CL</td>
<td>CLEAR FLAGS</td>
<td></td>
</tr>
<tr>
<td>BVTIDS/CV</td>
<td>CLEAR VTI FIELD MAP</td>
<td></td>
</tr>
<tr>
<td>BVTIDS/IN</td>
<td>INSERT FIELD</td>
<td></td>
</tr>
<tr>
<td>BVTIDS/RV</td>
<td>REBUILD VTI FIELD MAP</td>
<td></td>
</tr>
<tr>
<td>DEFFLD</td>
<td>DEFINE FIELD</td>
<td></td>
</tr>
<tr>
<td>DEFWND</td>
<td>DEFINE WINDOW</td>
<td></td>
</tr>
<tr>
<td>DRIVER/MA</td>
<td>MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER</td>
<td></td>
</tr>
<tr>
<td>ERAWND</td>
<td>ERASE WINDOW</td>
<td></td>
</tr>
<tr>
<td>FNDWND</td>
<td>FIND WINDOW</td>
<td></td>
</tr>
<tr>
<td>GETVT</td>
<td>GET DATA FROM VIRTUAL TERMINAL</td>
<td></td>
</tr>
<tr>
<td>PCHVTI</td>
<td>PUT SCREEN CHARACTERS TO VTI DATA STRUCTURE</td>
<td></td>
</tr>
<tr>
<td>PUTVT</td>
<td>PUT DATA TO VIRTUAL TERMINAL</td>
<td></td>
</tr>
<tr>
<td>RMVWND</td>
<td>REMOVE WINDOW</td>
<td></td>
</tr>
<tr>
<td>STFMTF</td>
<td>SET FORMAT FLAG FOR ALL CHILDREN WINDOWS AND FIELDS</td>
<td></td>
</tr>
<tr>
<td>STRDPN</td>
<td>SET READ PENDING FLAGS</td>
<td></td>
</tr>
<tr>
<td>STRDPN/ST</td>
<td>SET FIELD READ PENDING</td>
<td></td>
</tr>
<tr>
<td>SWNPRC</td>
<td>SET WINDOW PRECEDENCE</td>
<td></td>
</tr>
</tbody>
</table>
VIRTUAL TERMINAL Where-include-file-used List

<table>
<thead>
<tr>
<th>Include File</th>
<th>Module Name</th>
<th>Module Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

DEVINI

DRIVER/MA MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER

FUNCTS

BLDMSG BUILD MESSAGE
BLDMSG/BL BUILD BUFFER
BLDMSG/RE READ FLAG TURNED OFF
BVTIDS BUILD VTI DATA STRUCTURE
BVTIDS/BV BUILD VTI FIELD MAP
BVTIDS/CL CLEAR FLAGS
BVTIDS/CV CLEAR VTI FIELD MAP
BVTIDS/IN INSERT FIELD
BVTIDS/RV REBUILD VTI FIELD MAP
DEFFLD DEFINE FIELD
DEFWND DEFINE WINDOW
DOSCR/ERA ERASE PART OF SCREEN
DOSCR/HSC HORIZONTAL SCROLL
DOSCR/VSC VERTICAL SCROLL
DOSCREEN DO COMMAND TO INTERNAL SCREEN
DRIVER/MA MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
GETVT GET DATA FROM VIRTUAL TERMINAL
GVTICMD GET VIRTUAL TERMINAL INTERFACE COMMAND
INTVT INITIALIZE VIRTUAL TERMINAL
PCHVTI PUT SCREEN CHARACTERS TO VTI DATA STRUCTURE
PRCCMDS PROCESS COMMAND
PUTVT PUT DATA TO VIRTUAL TERMINAL
PVTICMD PUT VTI COMMAND
PVTICMD/P PUT NUMBER
REFRESH REFRESH TERMINAL
REFTERM REFRESH TERMINAL
VIRTUAL TERMINAL Where-include-file-used List

<table>
<thead>
<tr>
<th>Include File</th>
<th>Module Name</th>
<th>Module Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>STFMTF</td>
<td>SET FORMAT FLAG FOR ALL CHILDREN WINDOWS AND FIELDS</td>
</tr>
<tr>
<td></td>
<td>SWNPRC</td>
<td>SET WINDOW PRECEDENCE</td>
</tr>
<tr>
<td></td>
<td>TRMCHK</td>
<td>TERMINAL CHECK</td>
</tr>
<tr>
<td></td>
<td>TRMEND</td>
<td>TERMINAL END</td>
</tr>
<tr>
<td></td>
<td>TRMFLS</td>
<td>TERMINAL FLUSH</td>
</tr>
<tr>
<td></td>
<td>TRMGET</td>
<td>TERMINAL GET</td>
</tr>
<tr>
<td></td>
<td>TRMINI</td>
<td>TERMINAL INITIALIZE</td>
</tr>
<tr>
<td></td>
<td>TRMPUT</td>
<td>TERMINAL PUT</td>
</tr>
<tr>
<td></td>
<td>TRMVT</td>
<td>TERMINATE VIRTUAL TERMINAL</td>
</tr>
<tr>
<td></td>
<td>VT100/MOV</td>
<td>MOVE CURSOR (INTERNAL)</td>
</tr>
<tr>
<td></td>
<td>VT100/SET</td>
<td>SET ATTRIBUTES (INTERNAL)</td>
</tr>
</tbody>
</table>

NTM

DRIVER/MA MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER

SCREEN

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>BLDMSG</td>
<td>BUILD MESSAGE</td>
</tr>
<tr>
<td>BLDMSG/BL</td>
<td>BUILD BUFFER</td>
</tr>
<tr>
<td>BLDMSG/RE</td>
<td>READ FLAG TURNED OFF</td>
</tr>
<tr>
<td>BVTIDS</td>
<td>BUILD VTI DATA STRUCTURE</td>
</tr>
<tr>
<td>BVTIDS/BV</td>
<td>BUILD VTI FIELD MAP</td>
</tr>
<tr>
<td>BVTIDS/CL</td>
<td>CLEAR FLAGS</td>
</tr>
<tr>
<td>BVTIDS/CV</td>
<td>CLEAR VTI FIELD MAP</td>
</tr>
<tr>
<td>BVTIDS/IN</td>
<td>INSERT FIELD</td>
</tr>
<tr>
<td>BVTIDS/RV</td>
<td>REBUILD VTI FIELD MAP</td>
</tr>
<tr>
<td>CLRMOD</td>
<td>CLEAR MODIFY FLAGS</td>
</tr>
<tr>
<td>DOSCR/ERA</td>
<td>ERASE PART OF SCREEN</td>
</tr>
<tr>
<td>DOSCR/HSC</td>
<td>HORIZONTAL SCROLL</td>
</tr>
<tr>
<td>DOSCR/VSC</td>
<td>VERTICAL SCROLL</td>
</tr>
<tr>
<td>DOSCREEN</td>
<td>DO COMMAND TO INTERNAL SCREEN</td>
</tr>
<tr>
<td>DRIVER/MA</td>
<td>MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER</td>
</tr>
</tbody>
</table>

3-21
### VIRTUAL TERMINAL Where-include-file-used List

<table>
<thead>
<tr>
<th>Include File</th>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>GETVT</td>
<td>GET DATA FROM VIRTUAL TERMINAL</td>
<td></td>
</tr>
<tr>
<td>INTVT</td>
<td>INITIALIZE VIRTUAL TERMINAL</td>
<td></td>
</tr>
<tr>
<td>INVIS</td>
<td>CHECK FOR INVISIBILITY</td>
<td></td>
</tr>
<tr>
<td>PCHVTI</td>
<td>PUT SCREEN CHARACTERS TO VTI DATA STRUCTURE</td>
<td></td>
</tr>
<tr>
<td>PRCCMDs</td>
<td>PROCESS COMMAND</td>
<td></td>
</tr>
<tr>
<td>PUTVT</td>
<td>PUT DATA TO VIRTUAL TERMINAL</td>
<td></td>
</tr>
<tr>
<td>PVTICMD</td>
<td>PUT VTI COMMAND</td>
<td></td>
</tr>
<tr>
<td>PVTICMD/P</td>
<td>PUT NUMBER</td>
<td></td>
</tr>
<tr>
<td>REFRESH</td>
<td>REFRESH TERMINAL</td>
<td></td>
</tr>
<tr>
<td>REFTERM</td>
<td>REFRESH TERMINAL</td>
<td></td>
</tr>
<tr>
<td>SLINEND</td>
<td>FIND SCREEN LINE END</td>
<td></td>
</tr>
<tr>
<td>TRMCHK</td>
<td>TERMINAL CHECK</td>
<td></td>
</tr>
<tr>
<td>TRMEND</td>
<td>TERMINAL END</td>
<td></td>
</tr>
<tr>
<td>TRMFLS</td>
<td>TERMINAL FLUSH</td>
<td></td>
</tr>
<tr>
<td>TRMGET</td>
<td>TERMINAL GET</td>
<td></td>
</tr>
<tr>
<td>TRMINI</td>
<td>TERMINAL INITIALIZE</td>
<td></td>
</tr>
<tr>
<td>TRMPUT</td>
<td>TERMINAL PUT</td>
<td></td>
</tr>
<tr>
<td>TRMVVT</td>
<td>TERMINATE VIRTUAL TERMINAL</td>
<td></td>
</tr>
<tr>
<td>VT100/MOV</td>
<td>MOVE CURSOR (INTERNAL)</td>
<td></td>
</tr>
<tr>
<td>VT100/SET</td>
<td>SET ATTRIBUTES (INTERNAL)</td>
<td></td>
</tr>
</tbody>
</table>

### SIGNAL

| DRIVER/MA | MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER |

### STDIO

| DRIVER/MA | MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER |
| FATAL      | REPORT FATAL ERROR |
| TRMCHK     | TERMINAL CHECK |
| TRMEND     | TERMINAL END |
| TRMFLS     | TERMINAL FLUSH |
| TRMGET     | TERMINAL GET |
### VIRTUAL TERMINAL Where-include-file-used List

<table>
<thead>
<tr>
<th>Include File</th>
<th>Module</th>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>PVTICMD/P</td>
<td>PUT</td>
<td>NUMBER</td>
<td>PUT NUMBER</td>
</tr>
<tr>
<td>REFRESH</td>
<td>REFRESH</td>
<td>TERMINAL</td>
<td>REFRESH TERMINAL</td>
</tr>
<tr>
<td>REFTERM</td>
<td>REFRESH</td>
<td>TERMINAL</td>
<td>REFRESH TERMINAL</td>
</tr>
<tr>
<td>RMVWND</td>
<td>REMOVE</td>
<td>WINDOW</td>
<td>REMOVE WINDOW</td>
</tr>
<tr>
<td>SLINEND</td>
<td>FIND</td>
<td>SCREEN LINE END</td>
<td>FIND SCREEN LINE END</td>
</tr>
<tr>
<td>STFMTF</td>
<td>SET</td>
<td>FORMAT FLAG FOR ALL CHILDREN WINDOWS AND FIELDS</td>
<td>SET FORMAT FLAG FOR ALL CHILDREN WINDOWS AND FIELDS</td>
</tr>
<tr>
<td>STRDPN</td>
<td>SET</td>
<td>READ PENDING FLAGS</td>
<td>SET READ PENDING FLAGS</td>
</tr>
<tr>
<td>STRDPN/ST</td>
<td>SET</td>
<td>FIELD READ PENDING</td>
<td>SET FIELD READ PENDING</td>
</tr>
<tr>
<td>SWNPCR</td>
<td>SET</td>
<td>WINDOW PRECEDENCE</td>
<td>SET WINDOW PRECEDENCE</td>
</tr>
<tr>
<td>TPUTNUM</td>
<td>TERMINAL PUT NUMBER</td>
<td>PUT NUMBER</td>
<td>TERMINAL PUT NUMBER</td>
</tr>
<tr>
<td>TPUTS</td>
<td>TERMINAL PUT STRING</td>
<td>PUT STRING</td>
<td>TERMINAL PUT STRING</td>
</tr>
<tr>
<td>TRMCHK</td>
<td>TERMINAL CHECK</td>
<td>CHECK</td>
<td>TERMINAL CHECK</td>
</tr>
<tr>
<td>TRMEND</td>
<td>TERMINAL END</td>
<td>END</td>
<td>TERMINAL END</td>
</tr>
<tr>
<td>TRMFLS</td>
<td>TERMINAL FLUSH</td>
<td>FLUSH</td>
<td>TERMINAL FLUSH</td>
</tr>
<tr>
<td>TRMGET</td>
<td>TERMINAL GET</td>
<td>GET</td>
<td>TERMINAL GET</td>
</tr>
<tr>
<td>TRMINI</td>
<td>TERMINAL INITIALIZE</td>
<td>INITIALIZE</td>
<td>TERMINAL INITIALIZE</td>
</tr>
<tr>
<td>TRMPUT</td>
<td>TERMINAL PUT</td>
<td>PUT</td>
<td>TERMINAL PUT</td>
</tr>
<tr>
<td>TRMVVT</td>
<td>TERMINATE VIRTUAL TERMINAL</td>
<td>PUT</td>
<td>TERMINATE VIRTUAL TERMINAL</td>
</tr>
<tr>
<td>TVTPRC</td>
<td>TERMINATE VT100 PROCESS</td>
<td>VT100</td>
<td>TERMINATE VT100 PROCESS</td>
</tr>
<tr>
<td>VT100/MOV</td>
<td>MOVE CURSOR (INTERNAL)</td>
<td>MOV</td>
<td>MOVE CURSOR (INTERNAL)</td>
</tr>
<tr>
<td>VT100/SET</td>
<td>SET ATributes (INTERNAL)</td>
<td>SET</td>
<td>SET ATributes (INTERNAL)</td>
</tr>
</tbody>
</table>

### TERMIO

<table>
<thead>
<tr>
<th>DRIVER/MA</th>
<th>MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER</th>
</tr>
</thead>
<tbody>
<tr>
<td>PRCCMDS</td>
<td>PROCESS COMMAND</td>
</tr>
<tr>
<td>PUTVT</td>
<td>PUT DATA TO VIRTUAL TERMINAL</td>
</tr>
<tr>
<td>TPUTNUM</td>
<td>TERMINAL PUT NUMBER</td>
</tr>
<tr>
<td>TPUTS</td>
<td>TERMINAL PUT STRING</td>
</tr>
<tr>
<td>TRMCHK</td>
<td>TERMINAL CHECK</td>
</tr>
<tr>
<td>TRMEND</td>
<td>TERMINAL END</td>
</tr>
<tr>
<td>TRMFILS</td>
<td>TERMINAL FLUSH</td>
</tr>
<tr>
<td>TRMGET</td>
<td>TERMINAL GET</td>
</tr>
<tr>
<td>TRMINI</td>
<td>TERMINAL INITIALIZE</td>
</tr>
<tr>
<td>TRMPUT</td>
<td>TERMINAL PUT</td>
</tr>
<tr>
<td>VT100/MOV</td>
<td>MOVE CURSOR (INTERNAL)</td>
</tr>
</tbody>
</table>
VIRTUAL TERMINAL Where-include-file-used List

<table>
<thead>
<tr>
<th>Include File</th>
<th>Module Name</th>
<th>Module Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>TRMINI</td>
<td>TERMINAL INITIALIZE</td>
<td></td>
</tr>
<tr>
<td>TRMPUT</td>
<td>TERMINAL PUT</td>
<td></td>
</tr>
<tr>
<td>VT100/MOV</td>
<td>MOVE CURSOR (INTERNAL)</td>
<td></td>
</tr>
<tr>
<td>VT100/SET</td>
<td>SET ATTRIBUTES (INTERNAL)</td>
<td></td>
</tr>
</tbody>
</table>

STDTYP

<table>
<thead>
<tr>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABPOS</td>
<td>ABSOLUTIZE CURSOR POSITION OF FIELD</td>
</tr>
<tr>
<td>BLDMSG</td>
<td>BUILD MESSAGE</td>
</tr>
<tr>
<td>BLDMSG/BL</td>
<td>BUILD BUFFER</td>
</tr>
<tr>
<td>BLDMSG/RE</td>
<td>READ FLAG TURNED OFF</td>
</tr>
<tr>
<td>BVTIDS</td>
<td>BUILD VTI DATA STRUCTURE</td>
</tr>
<tr>
<td>BVTIDS/BV</td>
<td>BUILD VTI FIELD MAP</td>
</tr>
<tr>
<td>BVTIDS/CL</td>
<td>CLEAR FLAGS</td>
</tr>
<tr>
<td>BVTIDS/CV</td>
<td>CLEAR VTI FIELD MAP</td>
</tr>
<tr>
<td>BVTIDS/IN</td>
<td>INSERT FIELD</td>
</tr>
<tr>
<td>BVTIDS/RV</td>
<td>REBUILD VTI FIELD MAP</td>
</tr>
<tr>
<td>CLRMOD</td>
<td>CLEAR MODIFY FLAGS</td>
</tr>
<tr>
<td>DEFFLD</td>
<td>DEFINE FIELD</td>
</tr>
<tr>
<td>DEFWND</td>
<td>DEFINE WINDOW</td>
</tr>
<tr>
<td>DOSCR/ERA</td>
<td>ERASE PART OF SCREEN</td>
</tr>
<tr>
<td>DOSCR/HSC</td>
<td>HORIZONTAL SCROLL</td>
</tr>
<tr>
<td>DOSCR/VSC</td>
<td>VERTICAL SCROLL</td>
</tr>
<tr>
<td>DOSCREEN</td>
<td>DO COMMAND TO INTERNAL SCREEN</td>
</tr>
<tr>
<td>DRIVER/MA</td>
<td>MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER</td>
</tr>
<tr>
<td>ERAWND</td>
<td>ERASE WINDOW</td>
</tr>
<tr>
<td>FATAL</td>
<td>REPORT FATAL ERROR</td>
</tr>
<tr>
<td>FNDWND</td>
<td>FIND WINDOW</td>
</tr>
<tr>
<td>GETVT</td>
<td>GET DATA FROM VIRTUAL TERMINAL</td>
</tr>
<tr>
<td>GVTCMD</td>
<td>GET VIRTUAL TERMINAL INTERFACE COMMAND</td>
</tr>
<tr>
<td>INTVT</td>
<td>INITIALIZE VIRTUAL TERMINAL</td>
</tr>
<tr>
<td>INVIS</td>
<td>CHECK FOR INVISIBILITY</td>
</tr>
<tr>
<td>PCHVTI</td>
<td>PUT SCREEN CHARACTERS TO VTI DATA STRUCTURE</td>
</tr>
<tr>
<td>PRCCMDS</td>
<td>PROCESS COMMAND</td>
</tr>
<tr>
<td>PUTVT</td>
<td>PUT DATA TO VIRTUAL TERMINAL</td>
</tr>
<tr>
<td>PVTCMMD</td>
<td>PUT VTI COMMAND</td>
</tr>
</tbody>
</table>
VIRTUAL TERMINAL Where-include-file-used List

<table>
<thead>
<tr>
<th>Include File</th>
<th>Module Name</th>
<th>Module Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>VT100/SET</td>
<td>SET ATTRIBUTES (INTERNAL)</td>
</tr>
<tr>
<td></td>
<td>TRMRTN</td>
<td>DRIVER/MA MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER</td>
</tr>
<tr>
<td></td>
<td>GETVT</td>
<td>GET DATA FROM VIRTUAL TERMINAL</td>
</tr>
<tr>
<td></td>
<td>INTVT</td>
<td>INITIALIZE VIRTUAL TERMINAL</td>
</tr>
<tr>
<td></td>
<td>PRCCMDS</td>
<td>PROCESS COMMAND</td>
</tr>
<tr>
<td></td>
<td>PUTVT</td>
<td>PUT DATA TO VIRTUAL TERMINAL</td>
</tr>
<tr>
<td></td>
<td>REFRESH</td>
<td>REFRESH TERMINAL</td>
</tr>
<tr>
<td></td>
<td>REFTERM</td>
<td>REFRESH TERMINAL</td>
</tr>
<tr>
<td></td>
<td>TRMCHK</td>
<td>TERMINAL CHECK</td>
</tr>
<tr>
<td></td>
<td>TRMEND</td>
<td>TERMINAL END</td>
</tr>
<tr>
<td></td>
<td>TRMFLS</td>
<td>TERMINAL FLUSH</td>
</tr>
<tr>
<td></td>
<td>TRMGET</td>
<td>TERMINAL GET</td>
</tr>
<tr>
<td></td>
<td>TRMINI</td>
<td>TERMINAL INITIALIZE</td>
</tr>
<tr>
<td></td>
<td>TRMPUT</td>
<td>TERMINAL PUT</td>
</tr>
<tr>
<td></td>
<td>TRMVT</td>
<td>TERMINATE VIRTUAL TERMINAL</td>
</tr>
<tr>
<td></td>
<td>VT100/MOV</td>
<td>MOVE CURSOR (INTERNAL)</td>
</tr>
<tr>
<td></td>
<td>VT100/SET</td>
<td>SET ATTRIBUTES (INTERNAL)</td>
</tr>
</tbody>
</table>
3.10.6 **Where External Routine Used List**

The following lists each external function or routine listed in 3.10.3 and all the documented modules which call it. The purpose of each module is listed as well.
VIRTUAL TERMINAL Where-external-routine-used List

<table>
<thead>
<tr>
<th>System Module</th>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLDCMD</td>
<td>BVTIDS</td>
<td>BUILD VTI DATA STRUCTURE</td>
</tr>
<tr>
<td></td>
<td>DRIVER/MAIMAIN</td>
<td>MODULE FOR WINDOW MANAGER AND DEVICE DRIVER</td>
</tr>
<tr>
<td></td>
<td>GETVT</td>
<td>GET DATA FROM VIRTUAL TERMINAL</td>
</tr>
<tr>
<td></td>
<td>PRCCMDS</td>
<td>PROCESS COMMAND</td>
</tr>
<tr>
<td></td>
<td>PUTVT</td>
<td>PUT DATA TO VIRTUAL TERMINAL</td>
</tr>
<tr>
<td></td>
<td>REFRESH</td>
<td>REFRESH TERMINAL</td>
</tr>
<tr>
<td></td>
<td>REFTERM</td>
<td>REFRESH TERMINAL</td>
</tr>
<tr>
<td></td>
<td>TRMGET</td>
<td>TERMINAL GET</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CABIT</td>
<td>DOSCREEN</td>
<td>DO COMMAND TO INTERNAL SCREEN</td>
</tr>
<tr>
<td></td>
<td>TRMPUT</td>
<td>TERMINAL PUT</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CALLOC</td>
<td>DEFFLD</td>
<td>DEFINE FIELD</td>
</tr>
<tr>
<td></td>
<td>DEFWND</td>
<td>DEFINE WINDOW</td>
</tr>
<tr>
<td></td>
<td>DRIVER/MAIMAIN</td>
<td>MODULE FOR WINDOW MANAGER AND DEVICE DRIVER</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CBIT</td>
<td>CLRMOD</td>
<td>CLEAR MODIFY FLAGS</td>
</tr>
<tr>
<td></td>
<td>DOSCR/ERASERASE</td>
<td>PART OF SCREEN</td>
</tr>
<tr>
<td></td>
<td>DOSCR/VSCRVERTICAL</td>
<td>SCROLL</td>
</tr>
<tr>
<td></td>
<td>DOSCREEN</td>
<td>DO COMMAND TO INTERNAL SCREEN</td>
</tr>
<tr>
<td></td>
<td>REFRESH</td>
<td>REFRESH TERMINAL</td>
</tr>
<tr>
<td></td>
<td>REFTERM</td>
<td>REFRESH TERMINAL</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COL</td>
<td>BLDMSG</td>
<td>BUILD MESSAGE</td>
</tr>
<tr>
<td></td>
<td>BVTIDS</td>
<td>BUILD VTI DATA STRUCTURE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
VIRTUAL TERMINAL Where-external-routine-used List

<table>
<thead>
<tr>
<th>System</th>
<th>Module</th>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>DOSCR/ERASERASE PART OF SCREEN</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>DOSCR/HSCRHORIZONTAL SCROLL</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>DOSCR/VSCRVERTICAL SCROLL</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>DOSCREEN</td>
<td>DO COMMAND TO INTERNAL SCREEN</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PRCCMDS</td>
<td>PROCESS COMMAND</td>
</tr>
<tr>
<td></td>
<td></td>
<td>REFRESH</td>
<td>REFRESH TERMINAL</td>
</tr>
<tr>
<td></td>
<td></td>
<td>REFTERM</td>
<td>REFRESH TERMINAL</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SLINEND</td>
<td>FIND SCREEN LINE END</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TRMGET</td>
<td>TERMINAL GET</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TRMPUT</td>
<td>TERMINAL PUT</td>
</tr>
<tr>
<td></td>
<td></td>
<td>VT100/MOVCMOVE</td>
<td>MOVE CURSOR (INTERNAL)</td>
</tr>
<tr>
<td></td>
<td>CSTR</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>INTVT</td>
<td>INITIALIZE VIRTUAL TERMINAL</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DELAY</td>
<td>DRIVER/MAIMAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER</td>
<td></td>
</tr>
<tr>
<td></td>
<td>EXIT</td>
<td>DRIVER/MAIMAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER</td>
<td>TERM Expected at this point</td>
</tr>
<tr>
<td></td>
<td>FCLOSE</td>
<td>DRIVER/MAIMAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER</td>
<td></td>
</tr>
<tr>
<td></td>
<td>FFBDA</td>
<td>TRMPUT</td>
<td>TERMINAL PUT</td>
</tr>
</tbody>
</table>

3-28
**VIRTUAL TERMINAL Where-external-routine-used List**

<table>
<thead>
<tr>
<th>System Module</th>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>FFBSA</td>
<td>DOSCREEN</td>
<td>DO COMMAND TO INTERNAL SCREEN</td>
</tr>
<tr>
<td></td>
<td>TRMPUT</td>
<td>TERMINAL PUT</td>
</tr>
<tr>
<td></td>
<td>VT100/SETASET ATTRIBUTES (INTERNAL)</td>
<td></td>
</tr>
<tr>
<td>FFBSB</td>
<td>DOSCREEN</td>
<td>DO COMMAND TO INTERNAL SCREEN</td>
</tr>
<tr>
<td>FIX</td>
<td>DOSCREEN</td>
<td>DO COMMAND TO INTERNAL SCREEN</td>
</tr>
<tr>
<td>FLOOR</td>
<td>DOSCREEN</td>
<td>DO COMMAND TO INTERNAL SCREEN</td>
</tr>
<tr>
<td>FOPEN</td>
<td>DRIVER/MAIMAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER</td>
<td></td>
</tr>
<tr>
<td>FPRINTF</td>
<td>DRIVER/MAIMAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER</td>
<td></td>
</tr>
<tr>
<td>FREE</td>
<td>BVTIDS/CVTCLEAR VTI FIELD MAP</td>
<td></td>
</tr>
<tr>
<td></td>
<td>BVTIDS/INSINSERT FIELD</td>
<td></td>
</tr>
</tbody>
</table>

3-29
## VIRTUAL TERMINAL Where-external-routine-used List

<table>
<thead>
<tr>
<th>System</th>
<th>Module</th>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DRIVER</td>
<td>DRIVER/MAIN MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ERAWND</td>
<td>ERASE WINDOW</td>
<td></td>
</tr>
<tr>
<td></td>
<td>INTVT</td>
<td>INITIALIZE VIRTUAL TERMINAL</td>
<td></td>
</tr>
<tr>
<td></td>
<td>RMVWND</td>
<td>REMOVE WINDOW</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TRMVT</td>
<td>TERMINATE VIRTUAL TERMINAL</td>
<td></td>
</tr>
</tbody>
</table>

**FSEARCH**

DRIVER/MAIN MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER

**FWRITE**

DRIVER/MAIN MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER

**GETCHAR**

TRMGET TERMINAL GET  
TRMPUT TERMINAL PUT

**INITEX**

DRIVER/MAIN MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER

**ISDIGIT**

GVTICMD GET VIRTUAL TERMINAL INTERFACE COMMAND
TRMGET TERMINAL GET

**ISPRINT**
### VIRTUAL TERMINAL Where-external-routine-used List

<table>
<thead>
<tr>
<th>System Module</th>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DEFFLD</td>
<td>DEFINE FIELD</td>
</tr>
<tr>
<td></td>
<td>GVTICMD</td>
<td>GET VIRTUAL TERMINAL INTERFACE COMMAND</td>
</tr>
<tr>
<td></td>
<td>TRMGET</td>
<td>TERMINAL GET</td>
</tr>
</tbody>
</table>

**LIMIT**

- DOSCREEN DO COMMAND TO INTERNAL SCREEN

**malloc**

- BVTIDS/BVTBUILD VTI FIELD MAP
- BVTIDS/CVTCLEAR VTI FIELD MAP
- BVTIDS/INSINSERT FIELD
- DEFFLD DEFINE FIELD
- DRIVER/MAIMAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
- INVT INITIALIZE VIRTUAL TERMINAL

**max**

- BVTIDS BUILD VTI DATA STRUCTURE
- BVTIDS/INSINSERT FIELD
- DEFWND DEFINE WINDOW
- DOSCR/VSCRVERTICAL SCROLL
- DOSCREEN DO COMMAND TO INTERNAL SCREEN
- PCHVTI PUT SCREEN CHARACTERS TO VTI DATA STRUCTURE
- REFTERM REFRESH TERMINAL

**memcmp**

- BVTIDS BUILD VTI DATA STRUCTURE
- DRIVER/MAIMAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Where-external-routine-used List

<table>
<thead>
<tr>
<th>System Module</th>
<th>Module Name</th>
<th>Module Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>MI MCPY</td>
<td>BLDMSG</td>
<td>BUILD MESSAGE</td>
</tr>
<tr>
<td></td>
<td>BLDMSG/BLDBUILD BUFFER</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DRIVER/MAIMAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER</td>
<td></td>
</tr>
<tr>
<td>MEMSET</td>
<td>DEFFLD</td>
<td>DEFINE FIELD</td>
</tr>
<tr>
<td>MIH</td>
<td>BVTIDS/INSINSERT FIELD</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DEFWND</td>
<td>DEFINE WINDOW</td>
</tr>
<tr>
<td></td>
<td>DOSCR/HSCRHORIZONTAL SCROLL</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DOSCR/VSCRVERTICAL SCROLL</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DRIVER/MAIMAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER</td>
<td></td>
</tr>
<tr>
<td></td>
<td>REFTERM</td>
<td>REFRESH TERMINAL</td>
</tr>
<tr>
<td>NSEND</td>
<td>DRIVER/MAIMAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER</td>
<td></td>
</tr>
<tr>
<td>POS</td>
<td>DOSCREEN</td>
<td>DO COMMAND TO INTERNAL SCREEN</td>
</tr>
<tr>
<td></td>
<td>TRMPUT</td>
<td>TERMINAL PUT</td>
</tr>
<tr>
<td>PRINTF</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### VIRTUAL TERMINAL Where-external-routine-used List

<table>
<thead>
<tr>
<th>System Module Name</th>
<th>Module Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>DRIVFZ,'MAIMAIN</td>
<td>MODULE FOR WINDOW MANAGER AND DEVICE DRIVER</td>
</tr>
<tr>
<td>FATAL REPORT</td>
<td>FATAL ERROR</td>
</tr>
<tr>
<td>TRMGET</td>
<td>TERMINAL GET</td>
</tr>
<tr>
<td>TRMPUT</td>
<td>TERMINAL PUT</td>
</tr>
<tr>
<td>PRNEND</td>
<td>TERMINAL END</td>
</tr>
<tr>
<td>PRNFLS</td>
<td>TERMINAL PUT</td>
</tr>
<tr>
<td>PRHINI</td>
<td>TERMINAL INITIALIZE</td>
</tr>
<tr>
<td>PRHPUT</td>
<td>TERMINAL PUT</td>
</tr>
<tr>
<td>PUTC</td>
<td>DRIVER/MAIMAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER</td>
</tr>
<tr>
<td>RCV</td>
<td>DRIVER/MAIMAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER</td>
</tr>
</tbody>
</table>
VIRTUAL TERMINAL Where-external-routine-used List

<table>
<thead>
<tr>
<th>System</th>
<th>Module</th>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>RO1</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>BLDMSG</td>
<td>BUILD MESSAGE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>BVTIDS</td>
<td>BUILD VTI DATA STRUCTURE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DOSCR/VSCR_VERTICAL</td>
<td>SCROLL</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DOSCREEN</td>
<td>DO COMMAND TO INTERNAL SCREEN</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PRCCMDS</td>
<td>PROCESS COMMAND</td>
</tr>
<tr>
<td></td>
<td></td>
<td>REFRESH</td>
<td>REFRESH TERMINAL</td>
</tr>
<tr>
<td></td>
<td></td>
<td>REFTERM</td>
<td>REFRESH TERMINAL</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TRMGET</td>
<td>TERMINAL GET</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TRMPUT</td>
<td>TERMINAL PUT</td>
</tr>
<tr>
<td></td>
<td></td>
<td>VT100/MOVCMOVE</td>
<td>MOVE CURSOR (INTERNAL)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>System</th>
<th>Module</th>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>SBIT</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>DOSCR/ERASE</td>
<td>PART OF SCREEN</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DOSCR/VSCR_VERTICAL</td>
<td>SCROLL</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DOSCREEN</td>
<td>DO COMMAND TO INTERNAL SCREEN</td>
</tr>
<tr>
<td></td>
<td></td>
<td>REFRESH</td>
<td>REFRESH TERMINAL</td>
</tr>
<tr>
<td></td>
<td></td>
<td>REFTERM</td>
<td>REFRESH TERMINAL</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TRMPUT</td>
<td>TERMINAL PUT</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>System</th>
<th>Module</th>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>SIGNAL</td>
<td>DRIVER/MAIMAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SPRINTF</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>BLDMSG</td>
<td>BUILD MESSAGE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>BLDMSG/BLDBUILD</td>
<td>BUFFER</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DRIVER/MAIMAIN</td>
<td>MODULE FOR WINDOW MANAGER AND DEVICE DRIVER</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>System</th>
<th>Module</th>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>STRASN</td>
<td>BVTIDS</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>BUILD VTI DATA STRUCTURE</td>
</tr>
</tbody>
</table>
### VIRTUAL TERMINAL Where-external-routine-used List

<table>
<thead>
<tr>
<th>System Module</th>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>BVTIDS/CVTCLEAR</td>
<td>VTI FIELD MAP</td>
<td></td>
</tr>
<tr>
<td>BVTIDS/INSINSERT</td>
<td>FIELD</td>
<td></td>
</tr>
<tr>
<td>DOSCR/ERASERASE</td>
<td>PART OF SCREEN</td>
<td></td>
</tr>
<tr>
<td>STRCAT</td>
<td>DRIVER/MAIMAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER</td>
<td></td>
</tr>
<tr>
<td>STRCPY</td>
<td>DRIVER/MAIMAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER</td>
<td></td>
</tr>
<tr>
<td>STRLEN</td>
<td>BLDMSG BUILD MESSAGE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>BLDMSG/BLDBUILD BUFFER</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DRIVER/MAIMAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER</td>
<td></td>
</tr>
<tr>
<td>TBIT</td>
<td>DOSCR/ERASERASE PAR OF SCREEN</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DOSCR/HSCRHORIZONTAL SCROLL</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DOSCR/VSCRVERTICAL SCROLL</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DOSCREEN DO COMMAND TO INTERNAL SCREEN</td>
<td></td>
</tr>
<tr>
<td></td>
<td>GETVT GET DATA FROM VIRTUAL TERMINAL</td>
<td></td>
</tr>
<tr>
<td></td>
<td>INVIS CHECK FOR INVISIBILITY</td>
<td></td>
</tr>
<tr>
<td></td>
<td>PCHVTI PUT SCREEN CHARACTERS TO VTI DATA STRUCTURE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>PRCCMDS PROCESS COMMAND</td>
<td></td>
</tr>
<tr>
<td></td>
<td>PUTVT PUT DATA TO VIRTUAL TERMINAL</td>
<td></td>
</tr>
<tr>
<td></td>
<td>REFRESH REFRESH TERMINAL</td>
<td></td>
</tr>
<tr>
<td></td>
<td>REFTERM REFRESH TERMINAL</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TRMGET TERMINAL GET</td>
<td></td>
</tr>
<tr>
<td>System Module Name</td>
<td>Module Name</td>
<td>Purpose</td>
</tr>
<tr>
<td>-------------------</td>
<td>-------------</td>
<td>----------------------------------</td>
</tr>
<tr>
<td>TRMPUT</td>
<td>TRMPUT</td>
<td>TERMINAL PUT</td>
</tr>
<tr>
<td>TOPEN</td>
<td>TRMINI</td>
<td>TERMINAL INITIALIZE</td>
</tr>
<tr>
<td>TCHECK</td>
<td>TRMCHK</td>
<td>TERMINAL CHECK</td>
</tr>
<tr>
<td>TCLOSE</td>
<td>TRMEND</td>
<td>TERMINAL END</td>
</tr>
<tr>
<td>TFLUSH</td>
<td>TRMFLS</td>
<td>TERMINAL FLUSH</td>
</tr>
<tr>
<td>TGETC</td>
<td>TRMGET</td>
<td>TERMINAL GET</td>
</tr>
<tr>
<td>TOLOWER</td>
<td>DRIVER/MAIMAIN</td>
<td>MODULE FOR WINDOW MANAGER AND DEVICE DRIVER</td>
</tr>
<tr>
<td>THURGE</td>
<td>TRMGET</td>
<td>TERMINAL GET</td>
</tr>
</tbody>
</table>
VIRTUAL TERMINAL Where-external-routine-used List

<table>
<thead>
<tr>
<th>System Module</th>
<th>Module Name</th>
<th>Module Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>TPUTC</td>
<td>TPUTNUM</td>
<td>TERMINAL PUT NUMBER</td>
</tr>
<tr>
<td></td>
<td>TPUTS</td>
<td>TERMINAL PUT STRING</td>
</tr>
<tr>
<td></td>
<td>TRMPUT</td>
<td>TERMINAL PUT</td>
</tr>
<tr>
<td></td>
<td>VT100/MOVCMOVE CURSOR (INTERNAL)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>VT100/SETASET ATTRIBUTES (INTERNAL)</td>
<td></td>
</tr>
<tr>
<td>TRMNAT</td>
<td>DRIVER/MAIMAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TVTPRC</td>
<td>TERMINATE VTI PROCESS</td>
</tr>
<tr>
<td>ZERO</td>
<td>DOSCREEN</td>
<td>DO COMMAND TO INTERNAL SCREEN</td>
</tr>
</tbody>
</table>
3.10.7 Main Program Parts List

The following lists each Main Program listed in 3.10.1 and all the modules which are called either by that module itself or by any of the documented modules which it calls. It is possible for a non-main module to be listed more than once if it is called by multiple modules. The called modules, in this case known as program parts, are marked as to whether they are documented here. If so, the phrase "well-defined module" appears by the module name, if not it is an "external routine". The Purpose of the Main Program module is listed as well.
# VIRTUAL TERMINAL Main Program Parts List

<table>
<thead>
<tr>
<th>Main Pgm</th>
<th>Module</th>
<th>Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name</td>
<td>Type</td>
</tr>
<tr>
<td>--------</td>
<td>------</td>
<td>-----------------------</td>
</tr>
<tr>
<td>DRIVER/MAIN</td>
<td>Purpose: MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER</td>
<td></td>
</tr>
<tr>
<td>ABSPOS</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>BLDCMD</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>BLDMSG</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>BLDMSG/BLDBUF</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>BLDMSG/REDOFF</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>BVTIDS</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>BVTIDS/BVTIFM</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>BVTIDS/CLRFLG</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>BVTIDS/CVTIFM</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>BVTIDS/INSFLD</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>BVTIDS/RVTIFM</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>CABIT</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>CALLOC</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>CBIT</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>CLRMOD</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>COL</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>CSTR</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>DEFFLD</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>DEFWND</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>DELAY</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>DOSCR/ERASE</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>DOSCR/HSCR</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>DOSCR/VSCR</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>DOSCREEN</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>ERAWND</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>EXIT</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>FATAL</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>FCLOSE</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>FCLOSE</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>FRFDA</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>FFBSA</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>FFBSB</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>FIX</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>FLOOR</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>FNDWND</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>FOPEN</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>FPRINTF</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>FREE</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>FSEARCH</td>
<td>External routine</td>
<td></td>
</tr>
</tbody>
</table>
VIRTUAL TERMINAL Main Program Parts List

<table>
<thead>
<tr>
<th>Main Pgm Name</th>
<th>Module Name</th>
<th>Module Type</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>FWRITE</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>GETCHAR</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>GETVT</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>GVTCMD</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>INITEX</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>INTVT</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>INVIS</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>ISDIGIT</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>ISPRINT</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>LIMIT</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>MALLOC</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>MAX</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>MEMCMP</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>MEMCPY</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>MEMSET</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>MIN</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>NSEND</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>PCHVTI</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>POS</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>PRCCMDS</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>PRINTF</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>PRNEND</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>PRNFLS</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>PRNICI</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>PRNPUT</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>PUTC</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>PUTVT</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>PVTCMD</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>PVTCMD/PUTNUM</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>RCV</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>REFRESH</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>REFTERM</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>RMVWND</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>ROW</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>SBIT</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>SIGNAL</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>SLINEND</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>SPRINTF</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>STFMTF</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>STRASN</td>
<td>External routine</td>
</tr>
</tbody>
</table>
## VIRTUAL TERMINAL Main Program Parts List

<table>
<thead>
<tr>
<th>Main Pgm Name</th>
<th>Module Name</th>
<th>Module Type</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>STRCAT</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>STRCPY</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>STRDPN</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>STRDPN/STFD RD</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>STRLEN</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>SWNPRC</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>TBIT</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>TBOPEN</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>TCHECK</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>TCLOSE</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>TFIUSH</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>TGETC</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>TOLOWER</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>TPURGE</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>TPUTC</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>TPUTNUM</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>TPUTS</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>TRMCHK</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>TRMEND</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>TRMFLS</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>TRMGET</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>TRMINI</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>TRMNAT</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>TRMPUT</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>TRMVT</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>TVTPRC</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>VT100/MOV CUR</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>VT100/SETATR</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>ZERO</td>
<td>External routine</td>
</tr>
</tbody>
</table>
3.10.8 Module Documentation

The following documentation describes information which is specific to each individual module being documented in this specification as listed in section 3.10.2. It provides a compact way of getting information that would be otherwise buried within each module's source code.

The specific items in this module documentation have the following meanings:

<table>
<thead>
<tr>
<th>Term</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAME</td>
<td>Name of program Module.</td>
</tr>
<tr>
<td>PURPOSE</td>
<td>Purpose of Module as detailed in the source code.</td>
</tr>
<tr>
<td>LANGUAGE</td>
<td>Programming language source code is written in.</td>
</tr>
<tr>
<td></td>
<td>The choices are:</td>
</tr>
<tr>
<td></td>
<td>VAX-11 FORTRAN</td>
</tr>
<tr>
<td></td>
<td>C (I/S-1 Workbench 'C')</td>
</tr>
<tr>
<td></td>
<td>VAX-11 COBOL</td>
</tr>
<tr>
<td>MODULE TYPE</td>
<td>Whether a Program, Subroutine, or Function.</td>
</tr>
<tr>
<td>SOURCE FILE</td>
<td>Name of Source File from file specification.</td>
</tr>
<tr>
<td>SOURCE FILE TYPE</td>
<td>Source File Extension from file specification.</td>
</tr>
<tr>
<td>HOST</td>
<td>Whether this is a host-dependent routine (VAX or IBM) or blank if host-independent.</td>
</tr>
<tr>
<td>SUBSYSTEM</td>
<td>IISS sub-system this file resides in.</td>
</tr>
<tr>
<td>SUBDIRECTORY</td>
<td>Sub-directory of that subsystem in which this file resides.</td>
</tr>
<tr>
<td>DOCUMENTATION GROUP</td>
<td>Name of documentation group of which this source file is a member.</td>
</tr>
<tr>
<td>DESCRIPTION</td>
<td>A description of the module as obtained from the source code.</td>
</tr>
<tr>
<td>ARGUMENTS</td>
<td>The arguments with which this routine is called if it is a Subroutine or a Function.</td>
</tr>
<tr>
<td>INCLUDE FILES</td>
<td>A list of all the files that are included into this module as well as their purposes.</td>
</tr>
</tbody>
</table>
ROUTINES CALLED: Subroutines or Functions, either documented or external, called by this module, if any.

CALLED DIRECTLY BY: The documented routines which call this module, if any.

USED IN MAIN PROGRAM(S): The documented Main Programs which contain this module in their parts list according to the list in section 3.10.7.

The Module Documentation is arranged alphabetically according to Module Name.
VIRTUAL TERMINAL Module Documentation

NAME: abspos
PURPOSE: ABSOLUTIZE CURSOR POSITION OF FIELD
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: abspos
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:

SYNOPSIS
VOID ABSPOS(WNDPTR, ACRPOS)
WND *WNDPTR;
POSITION *ACRPOS;

INPUTS/OUTPUTS:

INPUTS:
WNDPTR - WINDOW WHOSE ROW AND COL WANT TO ABSOLUTIZED
ADDRESS OF STRUCTURE FOR RETURNING VALUES OF:
ABSOLUTE ROW
ABSOLUTE COL

OUTPUTS:
STRUCTURE CONTAINING:
ABSOLUTE ROW OF FIELD
ABSOLUTE COL OF FIELD

DESCRIPTION
THIS MODULE ABSOLUTIZES A FIELD'S ROW AND COL BY GOING
BACK UP
CHILD PARENT TREE AND ADDING EACH SUCCESSIVE PARENT'S ROW
AND COL
TO SUM OF CHILDS'.

ARGUMENTS:
----------
WNDPTR = WND *
ACRPOS = POSITION *

INCLUDE FILES:
---------------
STDTYP - STANDARD TYPE DEFINITIONS
DEVICE - PHYSICAL DEVICE DATA STRUCTURE

3-44
CALLED DIRECTLY BY:

- BVTIDS - BUILD VTI DATA STRUCTURE
- PCHVTI - PUT SCREEN CHARACTERS TO VTI DATA STRUCTURE

USED IN MAIN PROGRAM(S):

- DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: BLDMSG
PURPOSE: BUILD MESSAGE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID
SOURCE FILE: BLDMSG
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:
-------------

SYNOPSIS
VOID BLDMSG(BUFF, MAXLEN, LEN)
CHAR BUFF[];
INT MAXLEN;
INT *LEN;

INPUTS/OUTPUTS:

INPUTS:
ADDRESS OF BUFF WHERE FORMATED MESSAGE TO BE PUT
MAXLEN - LENGTH OF THIS MEMORY AREA
ADDRESS OF LOCATION WHERE LEN OF THIS FORMATTED MESSAGE
TO BE PUT

OUTPUTS:
BUFF - CONTAINES FORMATED MESSAGE
LEN - CONTAINES LENGTH OF THIS FORMATED MESSAGE

DESCRIPTION
THIS MODULE BUILDS A FORMATED MESSAGE (TO BE SENT ACROSS
NTM TO MONITOR)

ARGUMENTS:
-------------
BUFF = CHAR []
MAXLEN = INT
LEN = INT *

INCLUDE FILES:
-------------
STDTYP - STANDARD TYPE DEFINITIONS
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
SCREEN - INTERNAL SCREEN DEFINITIONS
FUNCTS - FUNCTION DEFINITIONS
CTLCHR - CONTROL CHARACTERS
DEVICE - PHYSICAL DEVICE DATA STRUCTURE
ROUTINES CALLED:
-----------------
ROW
COL
BLDMSG/BLDBUF - BUILD BUFFER
PRINTF
STRLEN
MEMCPY
BLDMSG/REDOFF - READ FLAG TURNED OFF
BVTIDS - BUILD VTI DATA STRUCTURE

CALLED DIRECTLY BY:
-------------------
GETVT - GET DATA FROM VIRTUAL TERMINAL

USED IN MAIN PROGRAM(S):
-------------------------
DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: BLDMSG/BLDBUF
PURPOSE: BUILD BUFFER
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID
SOURCE FILE: BLDMSG
SOURCE FILE TYPE: .C
HOST: 
SUBSYSTEM: UI
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:

SYNOPSIS

STATIC VOID BLDBUF(WNDPT, BUFPTR, BUFEND)
REGISTER WND *WNDPT;
REGISTER CHAR **BUFPTR;
CHAR *BUFEND;

INPUTS/OUTPUTS:

INPUTS:
WNDPT - POINTER TO WINDOW FROM WHICH TO GET INFO TO PUT IN MESSAGE
BUFPTR - ADDRESS OF POINTER WHERE FORMATED MESSAGE TO BE PUT
BUFEND - END OF THIS MEMORY AREA

OUTPUTS:
BUFPTR - POINTS TO LAST ENTRAY OF FORMATED MESSAGE

DESCRIPTION

THIS MODULE BUILDS A FORMATED MESSAGE (TO BE SENT ACROSS NTM TO MONITOR)
FROM WINDOW POINTED TO BY WNDPT

ARGUMENTS:

WNDPT = WND *
BUFPTR = CHAR **
BUFEND = CHAR *

INCLUDE FILES:

STDTYP - STANDARD TYPE DEFINITIONS
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
SCREEN - INTERNAL SCREEN DEFINITIONS
FUNCTS - FUNCTION DEFINITIONS
CTLCHR - CONTROL CHARACTERS
DEVICE - PHYSICAL DEVICE DATA STRUCTURE
VIRTUAL TERMINAL Module Documentation

NAME: BLDMSG/REDOFF
PURPOSE: READ FLAG TURNED OFF
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID
SOURCE FILE: BLDMSG
SOURCE FILE TYPE: C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:

SYNOPSIS
STATIC VOID REDOFF(WNDPT)
REGISTER WND *WNDPT;

INPUTS/OUTPUTS:

INPUTS:
WNDPT - POINTER TO WINDOW FROM WHICH DATA WAS READ

OUTPUTS:
NONE

DESCRIPTION
THIS MODULE TURNS OFF ALL READ FLAGS OF CHILD WINDOWS AND
FIELDS WHOSE
DATA HAS BEEN PUT IN FORMATED MESSAGE (TO BE SENT ACROSS
NTM TO MONITOR)
OF WINDOW POINTED TO BY WNDPT

ARGUMENTS:

WNDPT = WND *

INCLUDE FILES:

STDTYP - STANDARD TYPE DEFINITIONS
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
SCREEN - INTERNAL SCREEN DEFINITIONS
FUNCTS - FUNCTION DEFINITIONS
CTLCHR - CONTROL CHARACTERS
DEVICE - PHYSICAL DEVICE DATA STRUCTURE

ROUTINES CALLED:

BLDMSG/REDOFF - READ FLAG TURNED OFF

3-50
ROUTINES CALLED:
---------------------
BLDMSG/BLDBUF - BUILD BUFFER
MEMCPY
STRLEN
SPRINTF

CALLED DIRECTLY BY:
---------------------
BLDMSG/BLDBUF - BUILD BUFFER
BLDMSG - BUILD MESSAGE

USED IN MAIN PROGRAM(S):
------------------------
DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
CALLED DIRECTLY BY:

BLDMSG/REDOFF - READ FLAG TURNED OFF
BLDMSG    - BUILD MESSAGE

USED IN MAIN PROGRAM(S):

DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: BVTIDS
PURPOSE: BUILD VTI DATA STRUCTURE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: BOOL ()
SOURCE FILE: BVTIDS
SOURCE FILE TYPE: .C
HOST: 
SUBSYSTEM: UI
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRTFRM

DESCRIPTION:

SYNOPSIS
BOOL BVTIDS()

INPUTS/OUTPUTS:

INPUTS:
NONE

OUTPUTS:
RETURNS SUCCESS/FAILURE

DESCRIPTION
THIS MODULE (AND STATIC SUBMODULES) BUILDS THE VTI FIELD MAP USED BY VIRTUAL TERMINAL TO PAINT SCREEN ON TERMINAL(DEVICE) FROM THE INTERNAL DATA STRUCTURE.

INCLUDE FILES:

STDTYP - STANDARD TYPE DEFINITIONS
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
FUNCTS - FUNCTION DEFINITIONS
SCREEN - INTERNAL SCREEN DEFINITIONS
DEVICE - PHYSICAL DEVICE DATA STRUCTURE

ROUTINES CALLED:

BLDCMD
PRCCMDS - PROCESS COMMAND
BVTIDS/CLRFLG - CLEAR FLAGS
ROW
COL
ABSPOS - ABSOLUTIZE CURSOR POSITION OF FIELD
MAX
STRASN
MEMCMP
BVTIDS/RVTIFM - REBUILD VTI FIELD MAP

CALLED DIRECTLY BY:
---------------------
BLDMSG      - BUILD MESSAGE
PUTVT       - PUT DATA TO VIRTUAL TERMINAL

USED IN MAIN PROGRAM(S):
------------------------
DRIVER/MAI   - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: BVTIDS/BVTIFM
PURPOSE: BUILD VTI FIELD MAP
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: BOOL
SOURCE FILE: BVTIDS
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:
TRAVERSES THE INTERNAL STRUCTURE ADDING WINDOWS AND
FIELDS TO THE
FIELD MAP

ARGUMENTS:
WNDPT = WND *
BNDRY = INT []

INCLUDE FILES:
STDTYPE - STANDARD TYPE DEFINITIONS
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
FUNCTS - FUNCTION DEFINITIONS
SCREEN - INTERNAL SCREEN DEFINITIONS
DEVICE - PHYSICAL DEVICE DATA STRUCTURE

ROUTINES CALLED:
BVTIDS/BVTIFM - BUILD VTI FIELD MAP
BVTIDS/INSFLD - INSERT FIELD
MALLOC

CALLED DIRECTLY BY:
BVTIDS/RVTIFM - REBUILD VTI FIELD MAP
BVTIDS/BVTIFM - BUILD VTI FIELD MAP

USED IN MAIN PROGRAM(S):
DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: BVTIDS/CLRFLG
PURPOSE: CLEAR FLAGS
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID
SOURCE FILE: BVTIDS
SOURCE FILE TYPE: C
HOST: UI
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:

THIS MODULE CLEARS ALL FORMAT CHANGE AND CHANGE OUTPUT FLAGS

ARGUMENTS:

---------
WNDPT = WND *

INCLUDE FILES:

---------
STDTYP - STANDARD TYPE DEFINITIONS
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
FUNCTS - FUNCTION DEFINITIONS
SCREEN - INTERNAL SCREEN DEFINITIONS
DEVICE - PHYSICAL DEVICE DATA STRUCTURE

ROUTINES CALLED:

---------
BVTIDS/CLRFLG - CLEAR FLAGS

CALLED DIRECTLY BY:

---------
BVTIDS/CLRFLG - CLEAR FLAGS
BVTIDS - BUILD VTI DATA STRUCTURE

USED IN MAIN PROGRAM(S):

---------
DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME:          BVTIDS/CVTIFM
PURPOSE:       CLEAR VTI FIELD MAP
LANGUAGE:      C
MODULE TYPE:   FUNCTION
FUNCTION TYPE: BOOL ()
SOURCE FILE:   BVTIDS
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM:     UI
SUBDIRECTORY:  DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:
--------------
DESCRIPTION
REMOVES ALL OLD VTI FIELDS FROM THE MAP AND FREES THEM

INCLUDE FILES:
----------------
STDTYP - STANDARD TYPE DEFINITIONS
BITS  - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
FUNCTS - FUNCTION DEFINITIONS
SCREEN - INTERNAL SCREEN DEFINITIONS
DEVICE - PHYSICAL DEVICE DATA STRUCTURE

ROUTINES CALLED:
----------------
STRASN
MALLOC
FREE

CALLED DIRECTLY BY:
---------------------
BVTIDS/RVTIFM - REBUILD VTI FIELD MAP

USED IN MAIN PROGRAM(S):
------------------------
DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: BVTIDS/INSFLD
PURPOSE: INSERT FIELD
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: BOOL ()
SOURCE FILE: BVTIDS
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:
-----------------
DESCRIPTION
INSERTS A FIELD (OR WINDOW) INTO THE FIELD MAP TAKING CARE TO TRUNCATE,
SPLIT, OR REMOVE FIELDS ALREADY IN THE FIELD MAP WHICH ARE PARTIALLY OR
TOTALLY OBSCURED BY THE NEW FIELD. NOTE THAT THIS ASSUMES FIELDS ARE
INSERTED IN A BACK-TO-FRONT ORDER.

ARGUMENTS:
-----------
IVTIPT = VTIFLD *
WBNDRY = INT []

INCLUDE FILES:
-----------------
STDTYP - STANDARD TYPE DEFINITIONS
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
FUNCTS - FUNCTION DEFINITIONS
SCREEN - INTERNAL SCREEN DEFINITIONS
DEVICE - PHYSICAL DEVICE DATA STRUCTURE

ROUTINES CALLED:
--------------------
STRASN
MALLOC
FREE
MIN
MAX

CALLED DIRECTLY BY:
---------------------
BVTIDS/BVTIFM - BUILD VTI FIELD MAP
USED IN MAIN PROGRAM(S):

-----------------

DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: CLRMOD
PURPOSE: CLEAR MODIFY FLAGS
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: CLRMOD
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:

SYNOPSIS
VOID CLRMOD()

DESCRIPTION
CLEARS ALL THE MODIFY FLAGS IN THE INTERNAL SCREEN

INCLUDE FILES:

STDDTYP - STANDARD TYPE DEFINITIONS
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
SCREEN - INTERNAL SCREEN DEFINITIONS

ROUTINES CALLED:

CBIT

CALLED DIRECTLY BY:

GETVT - GET DATA FROM VIRTUAL TERMINAL
PUTVT - PUT DATA TO VIRTUAL TERMINAL

USED IN MAIN PROGRAM(S):

DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: BVTIDS/RVTIFM
PURPOSE: REBUILD VTI FIELD MAP
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: BOOL ()
SOURCE FILE: BVTIDS
SOURCE FILE TYPE: .C
HOST: UI
SUBSYSTEM: DRIVER
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:
-------------------------------
DESCRIPTION
REBUILD VTI FIELD MAP

INCLUDE FILES:
---------------------------
STDTYP - STANDARD TYPE DEFINITIONS
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
FUNCTS - FUNCTION DEFINITIONS
SCREEN - INTERNAL SCREEN DEFINITIONS
DEVICE - PHYSICAL DEVICE DATA STRUCTURE

ROUTINES CALLED:
---------------------
BVTIDS/BVTIFM - BUILD VTI FIELD MAP
BVTIDS/CVTIFM - CLEAR VTI FIELD MAP

CALLED DIRECTLY BY:
---------------------
BVTIDS - BUILD VTI DATA STRUCTURE

USED IN MAIN PROGRAM(S):
------------------------
DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: DEFFLD
PURPOSE: DEFINE FIELD
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: BOOL
SOURCE FILE: DEFFLD
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:

SYNOPSIS
BOOL DEFFLD(CMD,PTR)
  STRUCT COMMAND *CMD;
  CHAR **PTR;

INPUTS/OUTPUTS:

INPUTS:
  CMD - ADDRESS OF COMMAND STRUCTURE USED TO MODIFY DATA STRUCTURE
  PTR - POINTS TO END OF MESSAGE BUFFER BEING PROCESSED

OUTPUTS:
  PTR - WILL POINT TO END PROCESSED DATA IN BUFFER
  RETURNS SUCCESS / FAILURE

DESCRIPTION
THIS MODULE USING DATA IN COMMAND STRUCTURE AS WELL AS DATA STILL IN MESSAGE BUFFER MODIFIES INTERNAL DATA STRUCTURE OF FIELD SPECIFIED BY CURRENT WINDOW AND ROW AND COL OF FIELD FIELD BEING DEFINED - IF NO FIELD IS FOUND TO MODIFY THEN ONE IS CREATED.

ARGUMENTS:

CMD = STRUCT COMMAND *
PTR = CHAR **

INCLUDE FILES:

S;TDTYP - STANDARD TYPE DEFINITIONS
CTYPE - **** PURPOSE NOT FOUND BY STRIPPER ****
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
FUNCTS - FUNCTION DEFINITIONS
DEVICE - PHYSICAL DEVICE DATA STRUCTURE

ROUTINES CALLED:
-------------------
CALLOC
MALLOC
MEMSET
ISPRINT

CALLED DIRECTLY BY:
---------------------
PUTVT - PUT DATA TO VIRTUAL TERMINAL

USED IN MAIN PROGRAM(S):
------------------------
DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: DEFWND
PURPOSE: DEFINE WINDOW
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: BOOL ()
SOURCE FILE: DEFWND
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:

SYNOPSIS
BOOL DEFWND(CMD)
   STRUCT COMMAND *CMD;

INPUTS/OUTPUTS:

INPUTS:
   CMD - ADDRESS OF COMMAND STRUCTURE USED TO MODIFY DATA STRUCTURE

OUTPUTS:
   RETURNS SUCCESS / FAILURE

DESCRIPTION
   THIS MODULE USING DATA IN COMMAND STRUCTURE MODIFIES INTERNAL DATA STRUCTURE OF WINDOW SPECIFIED BY CURRENT WINDOW AND WNDID OF WINDOW BEING DEFINED - IF NO WINDOW IS FOUND TO MODIFY THEN ONE IS CREATED

ARGUMENTS:

   CMD = STRUCT COMMAND *

INCLUDE FILES:

   STDTYP - STANDARD TYPE DEFINITIONS
   BITS   - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
   FUNCTS - FUNCTION DEFINITIONS
   DEVICE - PHYSICAL DEVICE DATA STRUCTURE

3-63
ROUTINES CALLED:

- FNDWND - FIND WINDOW
- CALLOC
- MAX
- MIN
- STFMTE - SET FORMAT FLAG FOR ALL CHILDREN WINDOWS AND FIELDS

CALLED DIRECTLY BY:

- PUTVT - PUT DATA TO VIRTUAL TERMINAL

USED IN MAIN PROGRAM(S):

- DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: DOSCR/ERASE
PURPOSE: ERASE PART OF SCREEN
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: DOSCR
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:

ARGUMENTS:

-----------------
LO = INT
HI = INT

INCLUDE FILES:

-----------------
STDTYP - STANDARD TYPE DEFINITIONS
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
FUNCTS - FUNCTION DEFINITIONS
SCREEN - INTERNAL SCREEN DEFINITIONS

ROUTINES CALLED:

-----------------
SBIT
COL
TBIT
CBIT
STRASN

CALLED DIRECTLY BY:

-----------------
DOSCR/HSCR - HORIZONTAL SCROLL
DOSCR/VSCR - VERTICAL SCROLL
DOSCREEN - DO COMMAND TO INTERNAL SCREEN

USED IN MAIN PROGRAM(S):

-----------------
DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: DOSCR/HSCR
PURPOSE: HORIZONTAL SCROLL
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: DOSCR
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:

ARGUMENTS:

POS = INT
DIR = INT
N = INT*

INCLUDE FILES:

STDTYP - STANDARD TYPE DEFINITIONS
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
FUNCTS - FUNCTION DEFINITIONS
SCREEN - INTERNAL SCREEN DEFINITIONS

ROUTINES CALLED:

DOSCR/ERASE - ERASE PART OF SCREEN
MIN
COL
TBIT

CALLED DIRECTLY BY:

DOSCREEN - DO COMMAND TO INTERNAL SCREEN

USED IN MAIN PROGRAM(S):

DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME:              DOSCR/VSCR
PURPOSE:           VERTICAL SCROLL
LANGUAGE:          C
MODULE TYPE:       SUBROUTINE
FUNCTION TYPE:     VOID ()
SOURCE FILE:       DOSCR
SOURCE FILE TYPE:  .C
HOST:
SUBSYSTEM:         UI
SUBDIRECTORY:      DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:

ARGUMENTS:

POS = INT
N = INT

INCLUDE FILES:

STD Typ - STANDARD TYPE DEFINITIONS
BITs - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
FUNCTS - FUNCTION DEFINITIONS
SCREEN - INTERNAL SCREEN DEFINITIONS

ROUTINES CALLED:

SBIT
MAX
DOSCR/ERASE - ERASE PART OF SCREEN
ROW
MIN
COL
CBit
TBIT

CALLED DIRECTLY BY:

DOSCREEN - DO COMMAND TO INTERNAL SCREEN

USED IN MAIN PROGRAM(S):

DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: DOSCREEN
PURPOSE: DO COMMAND TO INTERNAL SCREEN
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: DOSCR
SOURCE FILE TYPE: .C
HOST: 
SUBSYSTEM: UI
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:

SYNOPSIS
INT DOSCREEN(CMD)
   STRUCT COMMAND *CMD;

DESCRIPTION
EXECUTES CMD ON THE INTERNAL SCREEN AND FIXES UP ITS PARAMETERS.
RETURNS -1 FOR ERRORS, 0 FOR NO ACTION, 1 FOR NORMAL COMMAND, AND 2 FOR MOVE THE CURSOR AND RETRY.

ARGUMENTS:

CMD = STRUCT COMMAND *

INCLUDE FILES:

STDTYP - STANDARD TYPE DEFINITIONS
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
FUNCTS - FUNCTION DEFINITIONS
SCREEN - INTERNAL SCREEN DEFINITIONS

ROUTINES CALLED:

TBIT
CBIT
FFBSA
DOSCR/HSCR - HORIZONTAL SCROLL
COL
MAX
FFBSB
ROW
DOSCR/VSCR - VERTICAL SCROLL
DOSCR/ERASE - ERASE PART OF SCREEN
FIX
LIMIT
POS
CABIT
FLOOR
SBIT
ZERO

CALLED DIRECTLY BY:

PRCCMDS - PROCESS COMMAND
TRMGET - TERMINAL GET

USED IN MAIN PROGRAM(S):

DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: DRIVER/MAIN
PURPOSE: MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: DRIVER
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: DEVDRV
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:

SYNOPSIS
MAIN()

DESCRIPTION
THIS IS THE MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER. IT SITS IN A POOLING LOOP GETTING MESSAGES FROM NTM AND PROCESSING THEM AND GETTING TERMINAL INPUT AND PROCESSING THAT. WHEN RUN, THREE OPTIONAL ARGUMENTS MAY BE SPECIFIED FOR SCRIPTING: -W<FILE> TO WRITE A SCRIPT FILE, -R<FILE> TO READ A SCRIPT FILE, AND -S<FILE> TO SAVE OUTPUT IN A FILE.

ARGUMENTS:
----------
ARGC = INT
ARGV = CHAR * []

INCLUDE FILES:
----------
STDTYP - STANDARD TYPE DEFINITIONS
STDLIO - **** PURPOSE NOT FOUND BY STRIPPER ****
TERMIO - TRANSPARENT TERMINAL I/O DEFINITIONS
DEVICE - PHYSICAL DEVICE DATA STRUCTURE
DEVINI - DEVICE INITIALIZATIONS
NTM - NTM INTERFACE INCLUDE FILE
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
FUNCTS - FUNCTION DEFINITIONS
SCREEN - INTERNAL SCREEN DEFINITIONS
CTLCHR - CONTROL CHARACTERS
SIGNAL - **** PURPOSE NOT FOUND BY STRIPPER ****
TRMRTN - TERMINAL (DEVICE DRIVER) ROUTINES

3-70
ROUTINES CALLED:

- BLDCMD
- TVTPRC - TERMINATE VTI PROCESS
- INITEX
- MEMCMP
- PUTVT - PUT DATA TO VIRTUAL TERMINAL
- FPRINTF
- PRINTF
- TRMNAT
- EXIT
- INTVT - INITIALIZE VIRTUAL TERMINAL
- INITLOC
- MALLOC
- FATAL - REPORT FATAL ERROR
- DELAY
- TRMVT - TERMINATE VIRTUAL TERMINAL
- RCV
- TRMCHK - TERMINAL CHECK
- GETVT - GET DATA FROM VIRTUAL TERMINAL
- SIGNAL
- MEMCPY
- STRCPY
- FREE
- STRCAT
- SPRINTF
- STRELEN
- NSEND
- FWRITE
- MII
- PUTC
- FCLOSE
- FSEARCH
- FOPEN
- PRCCMDS - PROCESS COMMAND
- TOLOWER
VIRTUAL TERMINAL Module Documentation

NAME: ERAWND
PURPOSE: ERASE WINDOW
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID
SOURCE FILE: ERAWND
SOURCE FILE TYPE: C
HOST: 
SUBSYSTEM: UI
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRTTERM

DESCRIPTION:

SYNOPSIS
VOID ERAWND(WNDPT)
WND *WNDPT;

INPUTS/OUTPUTS:

INPUTS:
WNDPT - POINTER TO WINDOW WISH TO FREE

OUTPUTS:
NONE

DESCRIPTION
THIS MODULE FREES ALL WINDOW'S CHILDREN WINDOWS AS WELL AS ALL DEPENDENT FIELDS.

ARGUMENTS:

WNDPT = WND *

INCLUDE FILES:

STDTYP - STANDARD TYPE DEFINITIONS
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
DEVICE - PHYSICAL DEVICE DATA STRUCTURE

ROUTINES CALLED:

ERAWND - ERASE WINDOW
FREE
CALLED DIRECTLY BY:

- ERAWND - ERASE WINDOW
- PUTVT - PUT DATA TO VIRTUAL TERMINAL
- RMVWND - REMOVE WINDOW

USED IN MAIN PROGRAM(S):

- DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: FATAL
PURPOSE: REPORT FATAL ERROR
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID
SOURCE FILE: FATAL
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:

SYNOPSIS

VOID FATAL(MSG)
CHAR MSG[];

INPUTS:
MSG - ERROR MESSAGE TO BE DISPLAYED (ERROR - %S\N)

DESCRIPTION
DISPLAYS THE SPECIFIED ERROR MESSAGE AND EXITS.

ARGUMENTS:

MSG = CHAR []

INCLUDE FILES:

STDTYP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****

ROUTINES CALLED:

TVTPRC - TERMINATE VTI PROCESS
PRINTF

CALLED DIRECTLY BY:

DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER

USED IN MAIN PROGRAM(S):

DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: FNDWND
PURPOSE: FIND WINDOW
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: WND * ()
SOURCE FILE: FNDWND
SOURCE FILE TYPE: .C
HOST: 
SUBSYSTEM: UI
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:

SYNOPSIS
WND *FNDWND(WNDID, FWNDPT)
REGISTER INT WNDID;
REGISTER WND *FWNDPT;

INPUTS/OUTPUTS:

INPUTS:
WNDID - ID OF WINDOW SEARCHING FOR
FWNDPT - POINTER TO FIRST WINDOW IN LIST TO BE SEARCHED

OUTPUTS:
RETURNS A POINTER TO WINDOW FOUND OR A NULL

DESCRIPTION
THIS MODULE SEARCHES FOR A WINDOW WITH THE ID GIVEN AND EITHER RETURNS A POINTER TO THE WINDOW FOUND OR A NULL.

ARGUMENTS:

WNDID = INT
FWNDPT = WND *

INCLUDE FILES:

STDTPY - STANDARD TYPE DEFINITIONS
DEVICE - PHYSICAL DEVICE DATA STRUCTURE

ROUTINES CALLED:

FNDWND - FIND WINDOW
CALLED DIRECTLY BY:
---------------------
DEFWND - DEFINE WINDOW
FNDWND - FIND WINDOW
PUTVT - PUT DATA TO VIRTUAL TERMINAL
RMVWND - REMOVE WINDOW

USED IN MAIN PROGRAM(S):
------------------------
DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: GETVT
PURPOSE: GET DATA FROM VIRTUAL TERMINAL
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: GETVT
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:

SYNOPSIS
VOID GETVT(BUFF, MAXLEN, LEN)
  CHAR *BUFF;
  INT *MAXLEN, *LEN;

DESCRIPTION
PERFORMS A READ FROM THE VIRTUAL TERMINAL. IF IN FORMS
MODE, BUFF
WILL CONTAIN A FORMATTED SCREEN, OTHERWISE IT WILL
CONSIST OF ALL THE
PRINTABLE CHARACTERS ENTERED PRIOR TO A COMMAND; IF IN
CONTROL TRANSFER
MODE, THE COMMAND WILL ALSO BE STORED. MAXLEN IS THE
LENGTH OF BUFF,
LEN IS THE NUMBER OF CHARACTER READ IN.

ARGUMENTS:

BUFF = CHAR *
MAXLEN = INT *
LEN = INT *

INCLUDE FILES:

STDTVTP - STANDARD TYPE DEFINITIONS
BITS - INCLUDE FILE FOR BIT MANIPULATION Routines
FUNCTS - FUNCTION DEFINITIONS
SCREEN - INTERNAL SCREEN DEFINITIONS
DEVICE - PHYSICAL DEVICE DATA STRUCTURE
TRMRRTN - TERMINAL (DEVICE DRIVER) ROUTINES

ROUTINES CALLED:

BLDCMD
CLRMOD - CLEAR MODIFY FLAGS
TBIT
PVTICMD - PUT VTI COMMAND
TRMGET - TERMINAL GET
PCHVTI - PUT SCREEN CHARACTERS TO VTI DATA STRUCTURE
BLDMSG - BUILD MESSAGE

CALLED DIRECTLY BY:
---------------------
DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER

USED IN MAIN PROGRAM(S):
------------------------
DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: GVTICMD
PURPOSE: GET VIRTUAL TERMINAL INTERFACE COMMAND
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: GVTICMD
SOURCE FILE TYPE: .C
HOST: SUBSYSTEM: UI
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:

SYNOPSIS

VOID GVTICMD(CMD, PTR, END)
  STRUCT COMMAND *CMD;
  CHAR **PTR, *END;

DESCRIPTION

PARSES THE NEXT VIRTUAL TERMINAL COMMAND INTO CMD AND
UPDATES PTR TO
POINT TO THE CHARACTER FOLLOWING IT. END IS A POINTER TO
THE CHARACTER
FOLLOWING THE END OF THE COMMAND STRING.

ARGUMENTS:

-------------

CMD = STRUCT COMMAND *
PTR = CHAR **
END = CHAR *

INCLUDE FILES:

-------------

STDTYP - STANDARD TYPE DEFINITIONS
CTYPE - **** PURPOSE NOT FOUND BY STRIPPER ****
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
FUNCTS - FUNCTION DEFINITIONS
CTLCHR - CONTROL CHARACTERS

ROUTINES CALLED:

--------------

ISPRINT
ISDIGIT

CALLED DIRECTLY BY:

-------------

PUTVT - PUT DATA TO VIRTUAL TERMINAL.
USED IN MAIN PROGRAM(S):

------------------

DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: INTVT
PURPOSE: INITIALIZE VIRTUAL TERMINAL
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID
SOURCE FILE: INTVT
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:

SYNOPSIS
VOID INTVT(TNAME, LEN)
CHAR *TNAME;
INT *LEN;

DESCRIPTION
OPENS THE VTI FOR THE TERMINAL SPECIFIED BY TNAME. LEN
IS THE NUMBER OF
CHARACTERS IN TNAME.

ARGUMENTS:

TNAME = CHAR *
LEN = INT *

INCLUDE FILES:

STDTYP - STANDARD TYPE DEFINITIONS
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
SCREEN - INTERNAL SCREEN DEFINITIONS
FUNCTS - FUNCTION DEFINITIONS
CTLCHR - CONTROL CHARACTERS
TRMRTN - TERMINAL (DEVICE DRIVER) ROUTINES

ROUTINES CALLED:

CSTR
MALLOC
FREE
PUTVT - PUT DATA TO VIRTUAL TERMINAL.
TRMINI - TERMINAL INITIALIZE

CALLED DIRECTLY BY:

DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
USED IN MAIN PROGRAM(S):

---------------------

DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: INVIS
PURPOSE: CHECK FOR INVISIBILITY
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: BOOL()
SOURCE FILE: INVIS
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:

SYNOPSIS

BOOL INVIS(POS)
INT POS;

DESCRIPTION

RETURNS TRUE IF THE CHARACTER AT POSITION POS ON THE INTERNAL SCREEN IS INVISIBLE, FALSE OTHERWISE.

ARGUMENTS:

POS = INT

INCLUDE FILES:

STDTYP - STANDARD TYPE DEFINITIONS
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
SCREEN - INTERNAL SCREEN DEFINITIONS

ROUTINES CALLED:

TBIT

CALLED DIRECTLY BY:

REFRESH - REFRESH TERMINAL
SLINEND - FIND SCREEN LINE END

USED IN MAIN PROGRAM(S):

DRIVER/MAI - MAIN MODULL FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: PCHVTI
PURPOSE: PUT SCREEN CHARACTERS TO VTI DATA STRUCTURE

LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: BOOL ()
SOURCE FILE: PCHVTI
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:

SYNOPSIS
BOOL PCHVTI()

INPUTS/OUTPUTS:

INPUTS:
NONE
OUTPUTS:
RETURNS FAILURE/SUCCESS

DESCRIPTION
THIS MODULE TAKES VT DATA IN "SCREEN" BUFFER AND PUTS IT INTO
VTI INTERNAL DATA STRUCTURE

INCLUDE FILES:

STDTYP - STANDARD TYPE DEFINITIONS
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
SCREEN - INTERNAL SCREEN DEFINITIONS
FUNCTS - FUNCTION DEFINITIONS
DEVICE - PHYSICAL DEVICE DATA STRUCTURE

ROUTINES CALLED:

TBIT
ABSPOS - ABSOLUTIZE CURSOR POSITION OF FIELD
MAX
CALLED DIRECTLY BY:

- GETVT       - GET DATA FROM VIRTUAL TERMINAL
- PUTVT       - PUT DATA TO VIRTUAL TERMINAL

USED IN MAIN PROGRAM(S):

- DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: PRCCMDS
PURPOSE: PROCESS COMMAND
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID
SOURCE FILE: PRCCMDS
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRT-RM

DESCRIPTION:

SYNOPSIS
VOID PRCCMDS(CMD)
STRUCT COMMAND *CMD;

INPUTS/OUTPUTS:

INPUTS:
 CMD - COMMAND TO BE PROCESSED

OUTPUTS:
 NONE

DESCRIPTION
 PROCESSES INDIVIDUAL VIRTUAL TERMINAL COMMANDS

ARGUMENTS:

CMD = STRUCT COMMAND *

INCLUDE FILES:

STDTYPE - STANDARD TYPE DEFINITIONS
TERMIO - TRANSPARENT TERMINAL I/O DEFINITIONS
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
FUNCTS - FUNCTION DEFINITIONS
SCREEN - INTERNAL SCREEN DEFINITIONS
TRMRTN - TERMINAL (DEVICE DRIVER) ROUTINES

ROUTINES CALLED:

BLDCMD
DOSCREEN - DO COMMAND TO INTERNAL SCREEN
TRMPUT - TERMINAL PUT
TBIT
ROW
COL

CALLED DIRECTLY BY:
---------------------
BVTIDS - BUILD VTI DATA STRUCTURE
DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
PUTVT - PUT DATA TO VIRTUAL TERMINAL

USED IN MAIN PROGRAM(S):
---------------------
DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: PUTVT
PURPOSE: PUT DATA TO VIRTUAL TERMINAL
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID (

SOURCE FILE: PUTVT
SOURCE FILE TYPE: .C

HOST: 
SUBSYSTEM: UI
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:

SYNOPSIS
VOID PUTVT(BUFF, LEN)
    CHAR *BUFF;
    INT *LEN;

DESCRIPTION
PERFORMS A WRITE TO THE VIRTUAL TERMINAL. LEN IS THE NUMBER OF CHARACTERS IN BUFF TO BE WRITTEN.

ARGUMENTS:

BUFF = CHAR *
LEN = INT *

INCLUDE FILES:

STDTYP - STANDARD TYPE DEFINITIONS
TERMIO - TRANSPARENT TERMINAL I/O DEFINITIONS
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
FUNCTS - FUNCTION DEFINITIONS
SCREEN - INTERNAL SCREEN DEFINITIONS
DEVICE - PHYSICAL DEVICE DATA STRUCTURE
TRMRTN - TERMINAL (DEVICE DRIVER) ROUTINES

ROUTINES CALLED:

BLDCMD
FNDWND - FIND WINDOW
PRCCMDS - PROCESS COMMAND
TBIT
TRMFLS - TERMINAL FLUSH
RMVWND - REMOVE WINDOW
DEFWND - DEFINE WINDOW
SWMRPC - SET WINDOW PRECEDENCE
DEFFLD - DEFINE FIELD
BVTIDS    - BUILD VTI DATA STRUCTURE
ERAWND    - ERASE WINDOW
PCHVTI    - PUT SCREEN CHARACTERS TO VTI DATA STRUCTURE
CLRMOD    - CLEAR MODIFY FLAGS
GVTICMD   - GET VIRTUAL TERMINAL INTERFACE COMMAND
STRDPN    - SET READ PENDING FLAGS

CALLED DIRECTLY BY:
---------
DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
INTVT     - INITIALIZE VIRTUAL TERMINAL
TRMVT     - TERMINATE VIRTUAL TERMINAL

USED IN MAIN PROGRAM(S):
---------
DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: PVTICMD
PURPOSE: PUT VTI COMMAND
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: PVTICMD
SOURCE FILE TYPE: .C
HOST: 
SUBSYSTEM: UI
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:

SYNOPSIS
VOID PVTICMD(CMD, BUFF, END)
  STRUCT COMMAND *CMD;
  CHAR **BUF, *END;

DESCRIPTION
CONVERTS CMD TO CHARACTER FORM AND UPDATE BUFF TO POINT TO THE CHARACTER FOLLOWING THE CONVERTED STRING. END IS A POINTER TO THE CHARACTER FOLLOWING THE BUFFER.

ARGUMENTS:

CMD = STRUCT COMMAND *
BUFF = CHAR **
END = CHAR *

INCLUDE FILES:

STDYP - STANDARD TYPE DEFINITIONS
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
SCREEN - INTERNAL SCREEN DEFINITIONS
FUNCTS - FUNCTION DEFINITIONS
CTLCHR - CONTROL CHARACTERS

ROUTINES CALLED:

PVTICMD/PUTNUM - PUT NUMBER

CALLED DIRECTLY BY:

GETVT - GET DATA FROM VIRTUAL TERMINAL
USED IN MAIN PROGRAM(S):

-------------

DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: PVTICMD/PUTNUM
PURPOSE: PUT NUMBER
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: PVTICMD
SOURCE FILE TYPE: C
HOST: SUBSYSTEM: UI
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:

ARGUMENTS:

NUM = INT
BUFF = CHAR **
END = CHAR *

INCLUDE FILES:

STDTYP - STANDARD TYPE DEFINITIONS
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
SCREEN - INTERNAL SCREEN DEFINITIONS
FUNCTS - FUNCTION DEFINITIONS
CTLCHR - CONTROL CHARACTERS

CALLED DIRECTLY BY:

PVTICMD - PUT VTI COMMAND

USED IN MAIN PROGRAM(S):

DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: REFRESH
PURPOSE: REFRESH TERMINAL
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: REFRESH
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:
--------------

SYNOPSIS
VOID REFRESH()

DESCRIPTION
CLEARS THE TERMINAL SCREEN AND REWRITES IT FROM THE INTERNAL SCREEN.

INCLUDE FILES:
---------------
STDTYP - STANDARD TYPE DEFINITIONS
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
SCREEN - INTERNAL SCREEN DEFINITIONS
FUNCTS - FUNCTION DEFINITIONS
TRMRTN - TERMINAL (DEVICE DRIVER) ROUTINES

ROUTINES CALLED:
----------------
BLDCMD
INVIS - CHECK FOR INVISIBILITY
ROW
COL
TBIT
TRMPUT - TERMINAL PUT
SBIT
TRMFLS - TERMINAL FLUSH
CBIT

CALLED DIRECTLY BY:
-------------------
TRMPUT - TERMINAL PUT

USED IN MAIN PROGRAM(S):
------------------------
DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER

3-93
VIRTUAL TERMINAL Module Documentation

NAME: REFTERM
PURPOSE: REFRESH TERMINAL
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: REFTERM
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:

SYNOPSIS
VOID REFTERM(MIN, MAX)
INT MIN, MAX;

DESCRIPTION
REFRESHES THE SPECIFIED PORTION OF THE TERMINAL SCREEN
FROM THE INTERNAL SCREEN.

ARGUMENTS:

MIN = INT
MAX = INT

INCLUDE FILES:

STDTYP - STANDARD TYPE DEFINITIONS
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
SCREEN - INTERNAL SCREEN DEFINITIONS
FUNCTS - FUNCTION DEFINITIONS
TRMRRTN - TERMINAL (DEVICE DRIVER) ROUTINES

ROUTINES CALLED:

BLDCMD
TBIT
TRMPUT - TERMINAL PUT
SBIT
SLINEND - FIND SCREEN LINE END
CBIT
COL
ROW
MAX
MIN
CALLED DIRECTLY BY:

-------------------
TRMPUT - TERMINAL PUT

USED IN MAIN PROGRAM(S):

-----------------------
DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME:          RMVWND
PURPOSE:       REMOVE WINDOW
LANGUAGE:      C
MODULE TYPE:   SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE:   RMVWND
SOURCE FILE TYPE: .C
HOST:          
SUBSYSTEM:     UI
SUBDIRECTORY:  DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:

---

SYNOPSIS

VOID RMVWND(WNDID)
    INT WNDID;

INPUTS/OUTPUTS:

INPUTS:
    WNDID - ID OF WINDOW WISH TO REMOVE

OUTPUTS:
    NONE

DESCRIPTION

THIS MODULE AFTER CALLING FNDWND TO GET POINTER TO WINDOW
INTERESTED IN REMOVING, UNLINKS IT FROM DATA STRUCTURE AND CALLS FREWND
TO FREE IT AND ALL ITS CHILDREN WINDOWS AS WELL AS ALL DEPENDENT
FIELDS.

ARGUMENTS:

---------

WNDID = INT

INCLUDE FILES:

-----------

STDTYP - STANDARD TYPE DEFINITIONS
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
DEVICE - PHYSICAL DEVICE DATA STRUCTURE

3-96
ROUTINES CALLED:
---------------
FNDWND       - FIND WINDOW
ERAWND       - ERASE WINDOW
FREE

CALLED DIRECTLY BY:
-------------------
PUTVT         - PUT DATA TO VIRTUAL TERMINAL

USED IN MAIN PROGRAM(S):
------------------------
DRIVER/MAI    - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: SLINEND
PURPOSE: FIND SCREEN LINE END
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: SLINEND
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:

SYNOPSIS

INT SLINEND(POS)
INT POS;

DESCRIPTION

RETURNS THE POSITION OF THE LAST VISIBLE CHARACTER ON THE
LINE CONTAINING
THE SPECIFIED POSITION.

ARGUMENTS:

POS = INT

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
SCREEN - INTERNAL SCREEN DEFINITIONS

ROUTINES CALLED:

INVIS - CHECK FOR INVISIBILITY
COL

CALLED DIRECTLY BY:

REFTERM - REFRESH TERMINAL

USED IN MAIN PROGRAM(S):

DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: STFMTF
PURPOSE: SET FORMAT FLAG FOR ALL CHILDREN WINDOWS AND FIELDS
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID (
SOURCE FILE: STFMTF
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:

SYNOPSIS

VOID STFMTF(WNDPT);
WND *WNDPT;

INPUTS/OUTPUTS:

INPUTS:
WNDPT - POINTER TO WINDOW SETTING FLAGS FOR

OUTPUTS:
NONE

DESCRIPTION

THIS MODULE SETS ALL FORMAT CHANGE FLAGS FOR WINDOW AND ITS CHILDREN WINDOWS AND FIELDS

ARGUMENTS:

WNDPT = WND *

INCLUDE FILES:

STDTYPE - STANDARD TYPE DEFINITIONS
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
FUNCTS - FUNCTION DEFINITIONS
DEVICE - PHYSICAL DEVICE DATA STRUCTURE

ROUTINES CALLED:

STFMTF - SET FORMAT FLAG FOR ALL CHILDREN WINDOWS AND FIELDS
CALLED DIRECTLY BY:

-------------
DEFWND  - DEFINE WINDOW
STFMTF  - SET FORMAT FLAG FOR ALL CHILDREN WINDOWS AND
          FIELDS
SWNPC   - SET WINDOW PRECEDENCE

USED IN MAIN PROGRAM(S):

-------------
DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: STRDPN
PURPOSE: SET READ PENDING FLAGS
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID
SOURCE FILE: STRDPN
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:

SYNOPSIS

VOID STRDPN(WNDPT)

REGISTER WND *WNDPT;

INPUTS/OUTPUTS:

INPUTS:
WNDPT - POINTER TO WINDOW FROM WHICH DATA IS TO BE READ

OUTPUTS:
NONE

DESCRIPTION

THIS MODULE TURNS ON ALL READ FLAGS OF CHILD WINDOWS AND FIELDS WHOSE DATA HAS IS TO BE PUT INTO FORMATED MESSAGE (TO BE SENT ACROSS NTM TO MONITOR) OF WINDOW POINTED TO BY WNDPT

ARGUMENTS:

-----------
WNDPT = WND *

INCLUDE FILES:

-----------
STDTYP - STANDARD TYPE DEFINITIONS
DEVICE - PHYSICAL DEVICE DATA STRUCTURE

ROUTINES CALLED:

-----------
STRDPN/STFDRD - SET FIELD READ PENDING

3-101
CALLED DIRECTLY BY:

---------
PUTVT - PUT DATA TO VIRTUAL TERMINAL

USED IN MAIN PROGRAM(S):

---------
DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: STRDPN/STFDRD
PURPOSE: SET FIELD READ PENDING
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: STRDPN
SOURCE FILE TYPE: .C
HOST: 
SUBSYSTEM: UI
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:

ARGUMENTS:

--------
WNDPT = WND *

INCLUDE FILES:

---------------
STDTYP - STANDARD TYPE DEFINITIONS
DEVICE - PHYSICAL DEVICE DATA STRUCTURE

ROUTINES CALLED:

---------------
STRDPN/STFDRD - SET FIELD READ PENDING

CALLED DIRECTLY BY:

---------------
STRDPN/STFDRD - SET FIELD READ PENDING
STRDPN - SET READ PENDING FLAGS

USED IN MAIN PROGRAM(S):

---------------
DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER

3-103
VIRTUAL TERMINAL Module Documentation

NAME: SWNPRC
PURPOSE: SET WINDOW PRECEDENCE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID()
SOURCE FILE: SWNPRC
SOURCE FILE TYPE: .C
HOST: 
SUBSYSTEM: UI
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:

SYNOPSIS
VOID SWNPRC(CMD)
    STRUCT COMMAND *CMD;

INPUTS/OUTPUTS:

INPUTS:
    CMD - ADDRESS OF COMMAND STRUCTURE USED TO SET
    PRECEDENCE OF
    WINDOWS

OUTPUTS:
    NONE

DESCRIPTION
THIS MODULE REORDERS PRECEDENCE OF WINDOWS, TAKING FIRST
WINDOW OUT OF
LIST AND PUTTING IT AT THE HEAD OF THE LIST, THEN TAKING
THE NEXT WINDOW
AND DOING THE SAME TING AND SO ON UNTIL ALL WINDOW PASSED
IN COMMAND
STRUCTURE HAVE BEEN PROCESSED.

ARGUMENTS:
--------
CMD = STRUCT COMMAND *

INCLUDE FILES:
--------
STDTYP - STANDARD TYPE DEFINITIONS
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
FUNCTS - FUNCTION DEFINITIONS
DEVICE - PHYSICAL DEVICE DATA STRUCTURE

3-104
ROUTINES CALLED:
---------------

STFMTF - SET FORMAT FLAG FOR ALL CHILDREN WINDOWS AND FIELDS

CALLED DIRECTLY BY:
---------------------

PUTVT - PUT DATA TO VIRTUAL TERMINAL

USED IN MAIN PROGRAM(S):
------------------------

DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: TPUTNUM
PURPOSE: TERMINAL PUT NUMBER
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: TPUTNUM
SOURCE FILE TYPE: .C
HOST: SUBSYSTEM: UI
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:

SYNOPSIS
VOID TPUTNUM(I, CHAN)

INT I;
TERM *CHAN;

DESCRIPTION
CONVERTS I TO CHARACTER FORM AND WRITES IT TO THE SPECIFIED TERMINAL.

ARGUMENTS:

I = INT
CHAN = TERM *

INCLUDE FILES:

STDTYP - STANDARD TYPE DEFINITIONS
TERMIO - TRANSPARENT TERMINAL I/O DEFINITIONS

ROUTINES CALLED:

TPUTC

CALLED DIRECTLY BY:

VT100/MOVCUR - MOVE CURSOR (INTERNAL)
VT100/SETATR - SET ATTRIBUTES (INTERNAL)
TRMPUT - TERMINAL PUT

USED IN MAIN PROGRAM(S):

DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER

3-106
VIRTUAL TERMINAL Module Documentation

NAME: TPUTS
PURPOSE: TERMINAL PUT STRING
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: TPUTS
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:
-------------

SYNOPSIS

VOID TPUTS(S, CHAN)
CHAR *S;
TERM *CHAN;

DESCRIPTION

WRITES THE SPECIFIED STRING TO THE SPECIFIED TERMINAL.

ARGUMENTS:
-------------

S = CHAR *
CHAN = TERM *

INCLUDE FILES:
-------------

STDTYP - STANDARD TYPE DEFINITIONS
TERMIO - TRANSPARENT TERMINAL I/O DEFINITIONS

ROUTINES CALLED:
-------------

TPUTC

CALLED DIRECTLY BY:
-------------

VT100/MOVCUR - MOVE CURSOR (INTERNAL)
VT100/SETATR - SET ATTRIBUTES (INTERNAL)
TRMPUT - TERMINAL PUT
TRMEND - TERMINAL END

USED IN MAIN PROGRAM(S):
-------------

DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: TRMCHK
PURPOSE: TERMINAL CHECK
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT
SOURCE FILE: VT100
SOURCE FILE TYPE: .C
HOST: SUBSYSTEM: UI SUBDIRECTORY: DEVDRV DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:

SYNOPSIS

INT TRMCHK()

DESCRIPTION

THIS MODULE RETURNS THE NUMBER OF CHARACTERS IN THE TYPE-AHEAD BUFFER.

INCLUDE FILES:

STDKIO - **** PURPOSE NOT FOUND BY STRIPPER ****
STDTYP - STANDARD TYPE DEFINITIONS
CTYPE - **** PURPOSE NOT FOUND BY STRIPPER ****
TERMIO - TRANSPARENT TERMINAL I/O DEFINITIONS
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
SCREEN - INTERNAL SCREEN DEFINITIONS
FUNCTS - FUNCTION DEFINITIONS
TRMRTN - TERMINAL (DEVICE DRIVER) ROUTINES
CI600.C" - **** PURPOSE NOT FOUND BY STRIPPER ****

ROUTINES CALLED:

---------
TCHECK

CALLED DIRECTLY BY:

---------
DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER

USED IN MAIN PROGRAM(S):

---------
DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: TRMEND
PURPOSE: TERMINAL END
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: VT100
SOURCE FILE TYPE: .C
HOST: 
SUBSYSTEM: UI
SUBDIRECTORY: DEVDRV
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:

SYNOPSIS
VOID TRMEND()

DESCRIPTION
RESETS THE CURRENTLY OPEN TERMINAL AND CLOSES IT.

INCLUDE FILES:

STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
STDTYP - STANDARD TYPE DEFINITIONS
CTYPE - **** PURPOSE NOT FOUND BY STRIPPER ****
TERMIO - TRANSPARENT TERMINAL I/O DEFINITIONS
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
SCREEN - INTERNAL SCREEN DEFINITIONS
FUNCTS - FUNCTION DEFINITIONS
TRMRTN - TERMINAL (DEVICE DRIVER) ROUTINES
C1600.C" - **** PURPOSE NOT FOUND BY STRIPPER ****

ROUTINES CALLED:

PRNEND
TPUTS - TERMINAL PUT STRING
TCLOSE

CALLED DIRECTLY BY:

TRMVT - TERMINATE VIRTUAL TERMINAL

USED IN MAIN PROGRAM(S):

DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER

3-109
VIRTUAL TERMINAL Module Documentation

NAME: TRMFLS
PURPOSE: TERMINAL FLUSH
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: VT100
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: DEVDRV
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:
---------------------
SYNOPSIS
VOID TRMFLS()

DESCRIPTION
FLUSH ANY TERMINAL BUFFERS.

INCLUDE FILES:
---------------------
STDDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
STDTyp - STANDARD TYPE DEFINITIONS
CTYpe - **** PURPOSE NOT FOUND BY STRIPPER ****
TERMIO - TRANSPARENT TERMINAL I/O DEFINITIONS
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
SCREEN - INTERNAL SCREEN DEFINITIONS
Functs - FUNCTION DEFINITIONS
TRMRTN - TERMINAL (DEVICE DRIVER) ROUTINES
CI600.C" - **** PURPOSE NOT FOUND BY STRIPPER ****

ROUTINES CALLED:
---------------------
VT100/MOVCUR - MOVE CURSOR (INTERNAL)
TFLUSH

CALLED DIRECTLY BY:
---------------------
PUTVT - PUT DATA TO VIRTUAL TERMINAL
REFRESH - REFRESH TERMINAL
TRMGET - TERMINAL GET

USED IN MAIN PROGRAM(S):
---------------------
DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER

3-110
VIRTUAL TERMINAL Module Documentation

NAME: TRMGET
PURPOSE: TERMINAL GET
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID
SOURCE FILE: VT100
SOURCE FILE TYPE: .C

DESCRIPTION:
SYNOPSIS
VOID TRMGET(CMD)
    STRUCT COMMAND *CMD;

DESCRIPTION
GETS THE NEXT COMMAND FROM THE TERMINAL AND CONVERTS IT TO INTERNAL FORM.

ARGUMENTS:
--------
CMD = STRUCT COMMAND *

INCLUDE FILES:
-------------
STDDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
STDTP - STANDARD TYPE DEFINITIONS
CTYPE - **** PURPOSE NOT FOUND BY STRIPPER ****
TERMIO - TRANSPARENT TERMINAL I/O DEFINITIONS
BITS - INCLUDE FILE FOR BIT MANIPULATION Routines
SCREEN - INTERNAL SCREEN DEFINITIONS
FUNCTS - FUNCTION DEFINITIONS
TRMRRTN - TERMINAL (DEVICE DRIVER) ROUTINES
CI600.C" - **** PURPOSE NOT FOUND BY STRIPPER ****

ROUTINES CALLED:
-------------------
BLDCMD
PRINTF
GETCHAR
TPURGE
ISPRINT
TGETC
DOSCREEN - DO COMMAND TO INTERNAL SCREEN
ROW
COL
TRMPUT - TERMINAL PUT
TRMFLS - TERMINAL FLUSH
TBIT
ISDIGIT

CALLED DIRECTLY BY:
-------------------------
GETVT - GET DATA FROM VIRTUAL TERMINAL

USED IN MAIN PROGRAM(S):
-------------------------
DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: TRMINI
PURPOSE: TERMINAL INITIALIZE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID
SOURCE FILE: VT100
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: DEVDRV
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:

SYNOPSIS
VOID TRMINI(TNAME)
   CHAR *TNAME;

DESCRIPTION
OPENS THE TERMINAL SPECIFIED BY TNAME AND Initializes IT.

ARGUMENTS:

TNAME = CHAR *

INCLUDE FILES:

STDXO - **** PURPOSE NOT FOUND BY STRIPPER ****
STDTYP - STANDARD TYPE DEFINITIONS
CTYPE - **** PURPOSE NOT FOUND BY STRIPPER ****
TERMIO - TRANSPARENT TERMINAL I/O DEFINITIONS
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
SCREEN - INTERNAL SCREEN DEFINITIONS
FUNCTS - FUNCTION DEFINITIONS
TRMRTN - TERMINAL (DEVICE DRIVER) ROUTINES
CI600.C" - **** PURPOSE NOT FOUND BY STRIPPER ****

ROUTINES CALLED:

TBOPEN
PRNINI

CALLED DIRECTLY BY:

INTVT - INITIALIZE VIRTUAL TERMINAL

USED IN MAIN PROGRAM(S):

DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: TRMPUT
PURPOSE: TERMINAL PUT
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID
SOURCE FILE: VT100
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: DEVDRV
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:
---------
SYNOPSIS
VOID TRMPUT(CMD)
STRUCT COMMAND *CMD;

DESCRIPTION
PUTS AN INTERNAL FORMAT COMMAND TO THE TERMINAL.

ARGUMENTS:
----------
CMD = STRUCT COMMAND *

INCLUDE FILES:
-------------
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
STDTYP - STANDARD TYPE DEFINITIONS
CTYPE - **** PURPOSE NOT FOUND BY STRIPPER ****
TERMIO - TRANSPARENT TERMINAL I/O DEFINITIONS
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
SCREEN - INTERNAL SCREEN DEFINITIONS
FUNCTS - FUNCTION DEFINITIONS
TRMRTN - TERMINAL (DEVICE DRIVER) ROUTINES
CI600.C" - **** PURPOSE NOT FOUND BY STRIPPER ****

ROUTINES CALLED:
-----------------
GETCHAR
PRINTF
PRNFLS
PRNPUT
REFTERM - REFRESH TERMINAL
TPUTNUM - TERMINAL PUT NUMBER
POS
TPUTS - TERMINAL PUT STRING
REFRESH - REFRESH TERMINAL
ROW
COL
TPUTC
VT100/SETATR - SET ATTRIBUTES (INTERNAL)
SBIT
FFBSA
CABIT
FFBDA
VT100/MOVCUR - MOVE CURSOR (INTERNAL)
TBIT

CALLED DIRECTLY BY:
-----------------------
PRCCMDS - PROCESS COMMAND
REFRESH - REFRESH TERMINAL
REFTERM - REFRESH TERMINAL
TRMGET - TERMINAL GET

USED IN MAIN PROGRAM(S):
-----------------------
DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: TRMVT
PURPOSE: TERMINATE VIRTUAL TERMINAL
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: TRMVT
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: DRIVER
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:

SYNOPSIS
VOID TRMVT();

DESCRIPTION
Closes the VTI.

INCLUDE FILES:

STDTYP - STANDARD TYPE DEFINITIONS
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
SCREEN - INTERNAL SCREEN DEFINITIONS
FUNCTS - FUNCTION DEFINITIONS
CTLCHR - CONTROL CHARACTERS
TRMRTN - TERMINAL (DEVICE DRIVER) ROUTINES

ROUTINES CALLED:

TRMEND - TERMINAL END
FREE
PUTVT - PUT DATA TO VIRTUAL TERMINAL

CALLED DIRECTLY BY:

DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
TVTPRC - TERMINATE VTI PROCESS

USED IN MAIN PROGRAM(S):

DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME:     TVTPRC
PURPOSE:  TERMINATE VTI PROCESS
LANGUAGE:  C
MODULE TYPE:  SUBROUTINE
FUNCTION TYPE:  VOID ()
SOURCE FILE:  TVTPRC
SOURCE FILE TYPE:  .C
HOST:
SUBSYSTEM:  UI
SUBDIRECTORY:  DRIVER
DOCUMENTATION GROUP:  VIRTERM

DESCRIPTION:

SYNOPSIS

VOID TVTPRC()

DESCRIPTION

THIS IS AN EXIT HANDLER FOR ABNORMAL TERMINATIONS

INCLUDE FILES:

STDTYP  - STANDARD TYPE DEFINITIONS

ROUTINES CALLED:

TRMNAT
TRMVT  - TERMINATE VIRTUAL TERMINAL
EXIT

CALLED DIRECTLY BY:

DRIVER/MAI  - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
FATAL  - REPORT FATAL ERROR

USED IN MAIN PROGRAM(S):

DRIVER/MAI  - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: VT100/MOVCUR
PURPOSE: MOVE CURSOR (INTERNAL)
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID
SOURCE FILE: VT100
SOURCE FILE TYPE: .c
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: DEVDRV
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:

SYNOPSIS
    STATIC VOID MOVCUR(NEWPOS)
    INT NEWPOS;

DESCRIPTION
    MOVES THE TERMINAL CURSOR TO THE SPECIFIED POSITION AND
    RESETS ANY
    PENDING POSITION.

ARGUMENTS:

NEWPOS = INT

INCLUDE FILES:

STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
STDTYP - STANDARD TYPE DEFINITIONS
CTYPE - **** PURPOSE NOT FOUND BY STRIPPER ****
TERMIO - TRANSPARENT TERMINAL I/O DEFINITIONS
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
SCREEN - INTERNAL SCREEN DEFINITIONS
FUNCTS - FUNCTION DEFINITIONS
TRMRRTN - TERMINAL (DEVICE DRIVER) ROUTINES
CI600.C" - **** PURPOSE NOT FOUND BY STRIPPER ****

ROUTINES CALLED:

TPUTC
TPUTNUM - TERMINAL PUT NUMBER
TPUTS - TERMINAL PUT STRING
COL
ROW
CALLED DIRECTLY BY:

---------------
TRMPUT - TERMINAL PUT
TRMFLS - TERMINAL FLUSH

USED IN MAIN PROGRAM(S):

---------------
DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
VIRTUAL TERMINAL Module Documentation

NAME: VT100/SETATR
PURPOSE: SET ATTRIBUTES (INTERNAL)
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID()
SOURCE FILE: VT100
SOURCE FILE TYPE: C
HOST: 
SUBSYSTEM: UI
SUBDIRECTORY: DEVDRV
DOCUMENTATION GROUP: VIRTERM

DESCRIPTION:

SYNOPSIS
VOID SETATR(ATR)
   INT ATR;

DESCRIPTION
SETS THE SPECIFIED TERMINAL ATTRIBUTES.

ARGUMENTS:

ATR = INT

INCLUDE FILES:

STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
STDTYP - STANDARD TYPE DEFINITIONS
CTYPE - **** PURPOSE NOT FOUND BY STRIPPER ****
TERMIO - TRANSPARENT TERMINAL I/O DEFINITIONS
BITS - INCLUDE FILE FOR BIT MANIPULATION ROUTINES
SCREEN - INTERNAL SCREEN DEFINITIONS
FUNCTS - FUNCTION DEFINITIONS
TRMRTN - TERMINAL (DEVICE DRIVER) ROUTINES
C1600.C" - **** PURPOSE NOT FOUND BY STRIPPER ****

ROUTINES CALLED:

TPUTNUM - TERMINAL PUT NUMBER
TPUTC
FFBSA
TPUTS - TERMINAL PUT STRING

CALLED DIRECTLY BY:

TRMPUT - TERMINAL PUT
USED IN MAIN PROGRAM(S):
------------------------
DRIVER/MAI - MAIN MODULE FOR WINDOW MANAGER AND DEVICE DRIVER
3.10.9 Include File Descriptions

The following list contains a purpose and description of each include file listed in 3.10.4 as specified in the source code. The language it is written in is also given.
VIRTUAL TERMINAL Include File Description

FILE NAME: BITS
PURPOSE: INCLUDE FILE FOR BIT MANIPULATION ROUTINES
LANGUAGE: C

DESCRIPTION:
-----------
VIRTUAL TERMINAL Include File Description

FILE NAME: CTLCHR
PURPOSE: CONTROL CHARACTERS
LANGUAGE: C

DESCRIPTION:
---------------

DESCRIPTION
DEFINITIONS OF ALL CONTROL CHARACTERS TO AVOID CHARACTER SET
DEPENDENCIES.
VIRTUAL TERMINAL Include File Description

FILE NAME: DEVICE
PURPOSE: PHYSICAL DEVICE DATA STRUCTURE
LANGUAGE: C

DESCRIPTION:

DESCRIPTION
THIS IS INCLUDE FILE FOR WINDOW MANAGER. IT CONTAINS DATA STRUCTURE
FOR THE PHYSICAL DEVICE AND ITS LOGICAL DEVICES AND WINDOWS.
VIRTUAL TERMINAL Include File Description

FILE NAME: DEVINI
PURPOSE: DEVICE INITIALIZATIONS
LANGUAGE: C

DESCRIPTION:

EXTERNAL DEFINITION AND INITIALIZING INCLUDE FILE FOR DEVICE.H
VIRTUAL TERMINAL Include File Description

FILE NAME: FUNCTS
PURPOSE: FUNCTION DEFINITIONS
LANGUAGE: C

DESCRIPTION:

DESCRIPTION

DEFINES THE MNEMONIC VIRTUAL TERMINAL COMMAND FUNCTIONS.
AND DEFINES STRUCTURE FOR PARSING VT1 MESSAGE BUFFER.
VIRTUAL TERMINAL Include File Description

FILE NAME: NTM
PURPOSE: NTM INTERFACE INCLUDE FILE
LANGUAGE: C

DESCRIPTION:
-----------------

DESCRIPTION
INCLUDE FILE FOR NTM INTERFACE
VIRTUAL TERMINAL Include File Description

ABS(A) - ABSOLUTE VALUE OF A
STRASN(A, B) - TRANSPORTABLE A = B FOR STRUCTURES
NULL - NULL POINTER VALUE (0)
TRUE - 1
FALSE - 0
SUCCESS - EXIT(SUCCESS) INDICATES SUCCESSFUL COMPLETION
FAILURE - EXIT(Failure) INDICATES ERRORS

THE FOLLOWING SYMBOLS SHOULD BE DEFINED BASED ON THE COMPILER BEING USED:
USHORT - COMPILER SUPPORTS UNSIGNED SHORT
TINY - COMPILER TREATS CHAR AS SIGNED
UTINY - CHAR IS SIGNED AND COMPILER SUPPORTS UNSIGNED CHAR
VOID - COMPILER SUPPORTS VOID
FORTRAN - COMPILER SUPPORTS FORTRAN
STRASN - DEFINE APPROPRIATE MACRO
SUCCESS - DEFINE APPROPRIATE VALUE IF NOT 0
FAILURE - DEFINE APPROPRIATE VALUE IF NOT 1
VIRTUAL TERMINAL Include File Description

FILE NAME: TERMIO
PURPOSE: TRANSPARENT TERMINAL I/O DEFINITIONS
LANGUAGE: C

DESCRIPTION:
---------

VIRTUAL TERMINAL Include File Description

FILE NAME: TRMRTN
PURPOSE: TERMINAL (DEVICE DRIVER) ROUTINES
LANGUAGE: C

DESCRIPTION:

DECLARATIONS FOR ALL TRM* DEVICE SPECIFIC DEVICE DRIVER ROUTINES.
3.10.10 Hierarchy Chart

The following hierarchy charts show the relationships between all of the modules mentioned in the above documentation. A module may call a subroutine several times within its code, but the call will only be shown once as a single relationship on this hierarchy chart. All modules shown at the top of the first page are considered Main Programs as described in section 3.10.1 above.

There is an internal paging scheme as marked by the numbers in the upper right corner of each page. An index after the last page of the chart shows where a routine and its calls are first defined. If a routine has no page reference, it either makes no calls or is an external routine. A continuation box on the end of a tree limb shows where that the tree continues on the page numbered mentioned. A number in a box with a routine name points to the page where the routine is further defined within the hierarchy tree. If there is no number in a box, the routine either makes no calls or is an external routine.
VIRTUAL TERMINAL Include File Description

FILE NAME: SCREEN
PURPOSE: INTERNAL SCREEN DEFINITIONS
LANGUAGE: C

DESCRIPTION:

DESCRIPTION
DEFINES SYMBOLS, EXTERNALS, ETC. FOR THE INTERNAL SCREEN BUFFER.
VIRTUAL TERMINAL Include File Description

FILE NAME: STDTYP
PURPOSE: STANDARD TYPE DEFINITIONS
LANGUAGE: C

DESCRIPTION:

THIS FILE ENSURES THAT THE FOLLOWING STANDARD TYPES ARE AVAILABLE:

- FLOAT - SINGLE PRECISION FLOAT
- DOUBLE - DOUBLE PRECISION FLOAT
- LONG - 32 BIT (OR LARGER) SIGNED INTEGER
- LBITS - 32 BITS (OR MORE) FOR BIT MANIPULATION
- INT - NATURAL SIZE SIGNED INTEGER
- UNSIGNED - NATURAL SIZE UNSIGNED INTEGER
- BOOL - NATURAL SIZE LOGICAL (ZERO / NON-ZERO ONLY)
- SHORT - 16 BIT (OR LARGER) SIGNED INTEGER
- USHORT - 16 BIT (OR LARGER) UNSIGNED INTEGER
- BITS - 16 BITS (OR MORE) FOR BIT MANIPULATION
- CHAR - SINGLE MACHINE CHARACTER (REAL CHARACTERS ALWAYS POSITIVE)
- TINY - 8 BIT (OR LARGER) SIGNED INTEGER
- UTINY - 8 BIT (OR LARGER) UNSIGNED INTEGER
- TBITS - 8 BITS (OR MORE) FOR BIT MANIPULATION
- TBOOL - 8 BIT (OR LARGER) LOGICAL (ZERO / NON-ZERO ONLY)
- METACHAR - 16 BIT (OR LARGER) AUGMENTED CHARACTER (SIGNED)
- VOID - FUNCTION THAT RETURNS NO VALUE
- FORTRAN - STORAGE CLASS FOR FOREIGN (NON-C) ROUTINES OR C ROUTINES WHICH ARE CALLABLE FROM FOREIGN ROUTINES

SINCE NOT ALL COMPILERS SUPPORT USHORT, TINY, AND UTINY, THE FUNCTIONS USHORT(), TINY(), AND UTINY() SHOULD BE USED WHENEVER REFERENCING THEM.

IN ADDITION, THE FOLLOWING UTILITY MACROS ARE DEFINED:

- LURSHIFT(N, B) - UNSIGNED LONG RIGHT SHIFT
- MAX(A, B) - MAXIMUM OF A AND B
- MIN(A, B) - MINIMUM OF A AND B
PS 620344300
30 September 1990

3

[Diagram of a tree-like structure with nodes labeled DRIVER/MAIN, PRINTF, TRMNAT, EXIT, (CONT), RMVWND, DEFWND, SWNPC, DEFFLD, (CONT), FNDWND, ERAWND, FREE, STFMTF, (CONT), INTVIT, CALLOC, MALLOC, FATAL, (CONT), TVTPRC, PRINTF.]

4

[Diagram of a tree-like structure with nodes labeled PUTVT, (CONT), RMVWND, DEFWND, SWNPC, DEFFLD, (CONT), FNDWND, ERAWND, FREE, STFMTF.]

5

[Diagram of a tree-like structure with nodes labeled DRIVER/MAIN, (CONT), INTVIT, CALLOC, MALLOC, FATAL, (CONT), TVTPRC, PRINTF.]

3-136
PS 620344300
30 September 1990

DEFWNDF

FNDWND  CALLOC  MAX  MIN  STFMTF

STFMTF

DEFFLD

CALLC  MALLOC  MEMSET  ISPRINT

PUTVT

(CONT)  BVTIDS  ERAWND  PCHVTI  CLRMOD  (CONT)

ERAWND  FREE  CBIT
TPUTNUM........... 29
TPUTS.............. 10
TRMCHK............. 10
TRMEND............. 10
TRMFLS............. 2
TRMGET............. 19
TRMINI............. 9
TRMNAT
TRMPUT............. 27
TRMVT............. 10
TVTPRC............. 1
VT100/MOVCUR....... 2
VT100/SETATR..... 37
ZERO
3.11 Program Listings Comments

This information is contained in the Module Descriptions in section 3.10.
SECTION 4
QUALITY ASSURANCE PROVISIONS

4.1 Introduction and Definitions

"Testing" is a systematic process that may be preplanned and explicitly stated. Test techniques and procedures may be defined in advance, and a sequence of test steps may be specified. "Debugging" is the process of isolation and correction of the cause of an error.

"Antibugging" is defined as the philosophy of writing programs in such a way as to make bugs less likely to occur and when they do occur, to make them more noticeable to the programmer and the user. In other words, as much error checking as is practical and possible in each routine should be performed.

4.2 Computer Programming Test and Evaluation

The quality assurance provisions for test consists of the normal testing techniques that are accomplished during the construction process. They consist of design and code walk-throughs, unit testing, and integration testing. These tests are performed by the design team. Structured design, design walk-through and the incorporation of "antibugging" facilitate this testing by exposing and addressing problem areas before they become coded "bugs."