

VrOne - SOCET SET

Introduction

SOCET SET® is a registered trademark of BAE SYSTEMS Mission Solutions Inc.
Geospatial eXploitation Products™ is a registered trademark of BAE SYSTEMS Mission Solutions Inc.
Copyright © 2006
All rights reserved.

BAE Systems' products are subject to the export control laws and associated regulations of the United States, including but not necessarily limited to the International Traffic in Arms Regulations and Export Administration Regulations. Thus parts of the functionality described herein may be available to certain customers only.

VrOne is a registered trademark of Cardinal Systems, LLC
Copyright © 1995-2006
All rights reserved.

All other trademarks are the property of their respective owners.

Last revised May 10, 2006

The information in this manual is furnished for informational use only, is subject to change without notice, and should not be construed as a commitment by BAE SYSTEMS or Cardinal Systems. Neither BAE SYSTEMS nor Cardinal Systems assume any responsibility or liability for any errors or inaccuracies that may appear in this document.



BAE SYSTEMS

Table of Contents

Introduction.....	1
Overview.....	1
VrOne Integration with SOCET SET	2
Download VrOne.....	2
Install VrOne.....	2
SOCET SET DLL Search Path.....	2
VrOne License Request.....	3
General VrOne-SOCET Configuration	4
VrOne Configuration	4
XYZ Digitizer	4
Keypoint.....	4
Start with Mouse or XYZ Stereoplotter	4
Custom Files	4
SOCET SET Configuration.....	5
VrOne-SOCET Button Definitions	6
Starting VrOne.....	7
Exiting VrOne.....	7
VrOne Help	7
Open an Existing VrOne File.....	7
Create a New VrOne File	7
Close an Existing VrOne File.....	8
Starting an Application (Command).....	8
Custom Symbol and Line Font Creation	9
Define a New VrOne Symbol File	9
Create a New VrOne Symbol or Line Font	9
Using a New VrOne Symbol or Line Font.....	9
Zoom and Window Display Functions.....	10
Zoom and Display Suggestions.....	11
Data Collection.....	12
Insert Line	12
Insert Fly Line.....	13
Insert Square	13
Insert Symbol.....	13
Insert Text	13
Insert Annotated Symbol.....	14
Insert Line Text	14
Changing the Graphical Attributes.....	14
Entity Search Radius	14
Data Edit.....	15
Edit Line	15
Edit Symbol.....	15
Edit Text	15
Trim Lines	15
Break Line	15
Reverse Line	15
Join Lines	16
Square Existing.....	16
Deleting Objects.....	17
Fast Delete.....	17
Delete Window.....	17
Cut Polygon.....	17
Selecting the Active Workspace.....	17

Batch Editing.....	18
Batch Join Lines.....	18
Batch Trim.....	18
Batch Node Lines	18
Batch Filter Lines.....	18
Batch Offset.....	18
Line Close.....	19
Global Change.....	20
Undo Last	20
Function Keys	20
Touch Pad and Key-in Input	21
Macros.....	21
Application Overlay	21
Sample Files.....	22
Troubleshooting Notes	22
Application Window Isn't Visible	22
Lines Aren't Displayed with Graphical Attributes.....	22
Appendix B – Line Fonts and Symbols.....	24
Appendix C – Types of Annotated Symbols.....	26
Index.....	33

Introduction

This purpose of this document is to provide an introduction for the use of VrOne with SOCET SET as an advanced map compilation and editing tool.

Although this is a generalized guide for VrOne-SOCET use, it's not expected that these notes will replace VrOne product training or even a detailed VrOne product demonstration. But rather, this document is primarily provided to help those users who wish to proceed with integration and product use on their own.

Overview

The historical roots of VrOne go back to the early 1980's. Since that time, VrOne and its related parent products have been used in numerous commercial photogrammetric companies. Consequently, VrOne has realized a significant evolution as a result of production-generated feedback. And as programming methods have also improved, VrOne has been able to provide functionality that is truly unique, such as Application Overlay.

VrOne stores all vector data in its own binary file format, which makes for very efficient data access and manipulation. VrOne can open and display up to 2000 vector files and can also open and display up to 2000 TIFF images (with world files) as a backdrop to the vector display.

The VrOne vector database has the following characteristics:

- 3D double precision
- 10,001 data layers
- Unlimited file size
- Unlimited number of points on a line
- 48 character feature code available for each entity
- 32 bit non-graphic pointer
- Text labels up to 512 characters
- Support for 256 vector colors

Integration of VrOne with SOCET SET

Download VrOne

Users wanting to download the latest version of VrOne-SOCET need to log on to the Cardinal Systems website and request an access code for FTP retrieval:

<http://www.cardinalsystems.net/UploadDownload.htm>

After obtaining the FTP access code from Cardinal Systems and logging in, go to the **VrSocet** directory and download the **vrssAAA-BBB.exe** (~55 MB) self-extracting executable to a setup or temp directory on your system.

The VrOne-SOCET installation file name includes both the VrOne and SOCET SET version numbers. For example Vrss301-530.exe is VrOne version 3.01 for SOCET SET version 5.3.

Install VrOne

IMPORTANT NOTICE!

Before the VrOne setup program or the VrOne HostID Utility program can be executed, a password MUST be entered when prompted. This password is only available via email. Please send your request to mike@vrone.com and to curt.lima@baesystems.com.

Using Windows Explorer, go the setup or temp directory into which the VrOne installation file was downloaded and double-click on **vrssAAA-BBB.exe**. Since this self-extracting executable is password-protected, enter the password obtained as per the above **IMPORTANT NOTICE**.

During the installation process, the default installation directory of **C:\vr** can be changed if necessary. And if desired, short-cut icons on the Desktop can be created for **vrone_ss.exe** and **vrconfig_ss.exe**.

SOCET DLL Search Path

Note: If **vrone_ss** or **vrconfig_ss** is executed from a Desktop icon or by double-clicking on the program from within Windows Explorer, an “Unable to Locate Component” message window may appear that says: **“This application has failed to start because librti_5*.dll was not found. Re-installing the application may fix this problem.”**

Although it shouldn't be necessary to re-install the application, this message does indicate that the required SOCET SET DLL files can't be found in the system's search path. As a result, this error can be addressed by either of the following.

- Select **VrOne SOCET** or **VrOne Config** from the SOCET SET pull-down menu. This method causes VrOne to explicitly inherit the environment variables normally defined during the execution of SOCET SET, which does include the SOCET **.\lib** directory.
- Manually include the SOCET **.\lib** directory in your system's search path by:
 - o Right-mouse clicking on the My Computer desktop icon, then selecting Properties > Advanced > Environment Variables.
 - o In the System Variables window, select Path, press the Edit button, and add the SOCET **.\lib** directory to the Variable Value field. Then press the OK buttons to save and exit.

Of the two methods, *manually* adding the SOCET **.\lib** directory to your system's search path will provide more flexibility insofar that VrOne can then be executed without having SOCET SET running.

VrOne License Request

The information required for the VrOne license can be obtained by executed the `\vr\bin\vrone_ss.exe` program. If the VrOne license file hasn't been installed, a **Vr Security – VrOne** window will appear with the Computer Name, HostID, UserName and Password.

To facilitate the delivery of the VrOne license file to you, an email request can be made with the following:

To: vipsupport@baesystems.com
Cc: curt.lima@baesystems.com
Cc: <Your_BAE_Sales/Support_Representative>
Subject: VrOne-SOCET License Request

Your Name:
Contact Email Address:
Company Name:
Company Address:
Company Phone Number:

License Request Type: Demo/Evaluation or Purchased/Permanent, as indicated in the **Vr Security – VrOne** window.

Computer Name:
HostID:
UserName:

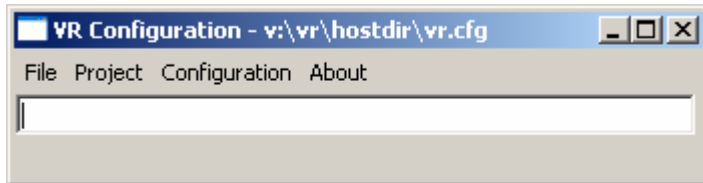
Upon verification of the license request, an email containing the VrOne password will be sent to you. It is that license code string that will need to be entered into the "Password" field of the **Vr Security – VrOne** window.

If the VrOne license has been extended or upgraded, the `\vr\data\vr.sec` file can be renamed to allow for the entry of the updated license code key.

General VrOne-SOCET Configuration

VrOne Configuration

The VrOne Configuration can be accessed by either double-clicking on the **Vr SOCET Configuration** icon, or by using Windows Explorer to navigate to the `\vr\bin\` directory and double-clicking on `vrcfig_ss.exe`.



In the **VR Configuration** window:

```
Project > Project Parameters
Use default directories?      No
```

```
Configuration > XYZ Digitizer
XYZ Digitizer                BAE SYSTEMS SOCET SET
```

```
Configuration > Keypoint
Keypoint Type                Keypoint 252 USB or Serial (if using a Keypoint)
                             None (if not using a Keypoint)
```

If the user wishes to always start with either the Mouse or XYZ Input:

- **Vr SOCET Config: General > Start input device.** Select **Mouse** or **XYZ Stereoplotter** and press **OK**.

But from within VrOne, either input mode can be selected by entering `digmou` or `digxyz` in the **VrOne key-in** window.

If the user wants to use customized data files:

- **Vr SOCET Config > Configuration > General > Function Key File**
Enter or browse to the custom file and press **OK**.
- **Vr SOCET Config > Configuration > General > Keypad Command File**
Enter or browse to the custom file and press **OK**.
- **Vr SOCET Config > Configuration > General > Macro File**
Enter or browse to the custom file and press **OK**.
- **Vr SOCET Config > Configuration > Vector Graphics > Symbol File Name**
Enter or browse to the custom file and press **OK**.
- **Vr SOCET Config > Configuration > Vector Graphics > Pen Table File Name**
Enter or browse to the custom file and press **OK**.

The colors used with VrOne (**VrSOCET Config > Configuration > Vector Graphics > Use pen table?**) can be defined in one of two ways:

If this is set to **Yes**, then VrOne will use the specified pen table (which is one line above the “Use pen table?” line) and will map out the colors by VrOne layers. If the default VrOne pen table (`\vr\data\default.pen`) is opened with a text editor, the Layer / Entity Type / Pen Number (color) is shown. Each VrOne layer (1-10001) can have a separate pen number for line entities, splines, symbols and text. This means that the colors will always be fixed based on the entity type for a specific layer. (For example, all lines found in layer 35 would

have the same color.) The default is to have one pen number (color) for all of the entities within a layer, but this isn't a requirement. If desired, one VrOne layer could have different colors for each of the four entity types.

If "Use pen table?" is set to **Yes**, it's possible to cycle through the available colors with the following:

1. Insert a line into a VrOne file.
2. Pick the line with **Pen > Change Pen**.
3. Press **Button 2** to increment or **Button 3** to decrement the pen number (color). The **Change to** pen number and color are shown in the main **VrOne** menu window.

If "Use pen table?" is set to **No**, then the user will define colors to be used and won't be forced to use the same color for all of the elements within a particular Layer-Entity type. This would then allow for the placement of utility valve, water valves, gas valves, ... into the same VrOne layer, but each could have a different color.

Lastly, VrOne doesn't display all color combinations to the screen, but only goes up to pen number 70. The screen display colors are determined by `\vr\data\defaultscreen.ct`, and this file also provides a cross-reference of pen number to color name.

Please refer to Appendix A – Pen Table Colors for a listing of the default screen display colors.

SOCET SET Configuration

Once SOCET SET is running, VrOne can be loaded by selecting the **VrOne SOCET** icon from the desktop or by selecting **VrOne > VrOne SOCET** from the **SOCET SET Extraction** pull down menu.

The current SOCET SET versions should include **VrOne SOCET** and **VrOne Config** in the **SOCET SET Extraction** pull down menu. If VrOne is not present, it is possible to manually add those items to the menu as follows:

1. Open `\usr\geoset\internal_dbs\CONFIG\standard_NT.cnf` with a text editor.
2. Add the last three highlighted VrOne lines, below, to the **Extraction** section.

Please note that tabs must be used to indent the lines, spaces will not be recognized.

```
Extraction|0|0|0
  Terrain|0|0|0
    Automatic Extraction|0|2|0|$(EXEDIR)/ate PREF
    Interactive Edit|2|1|0|$(EXEDIR)/ite PREF
  Feature|0|0|0
    Feature Specification Editor|0|0|0|$(EXEDIR)/spec_ed
    Feature Extraction|2|1|0|$(EXEDIR)/fei PREF RT
    Feature Database Merge|2|1|0|$(EXEDIR)/fdbm PREF
  ClearFlite|2|1|0|$(EXEDIR)/clearflite
  Merge|0|2|0|$(EXEDIR)/dtm_ftr PREF RT
  Annotation & Counting|2|1|0|$(EXEDIR)/annot PREF
  VrOne|0|0|0
    VrOne SOCET|0|1|0|$(VRHOME)/bin/vrone_ss
    VrOne Config|0|0|0|$(VRHOME)/bin/vrcfig_ss
```

VrOne-SOCET Button Definitions

As with other SOCET SET applications, it is possible to cross-reference the 12 VrOne buttons to the SOCET SET 3D input device with the following:

1. Copy `\vr\bin\vrone.acc` to the SOCET SET events directory (e.g., `\usr\geoset\internal_dbs\EVENTS` or `\SOCET_SET_5.3.0\internal_dbs\EVENTS`).
2. Rename `...\internal_dbs\EVENTS\vrone.acc` to `...\internal_dbs\EVENTS\VRONE`.
Please note that VRONE must be upper-case.
3. Save a copy of the current SOCET key definitions:
`copy $HOME\socet_keys.localhost $HOME\original_keys`
4. Start up SOCET SET and activate the **Keyboard/Trackball Accelerator Mapping** window (SOCET: **Preference > Keyboard/Trackball Accelerator Mapping**).
5. From within the **Keyboard/Trackball Accelerator Mapping** window, press the **Restore All Defaults** button to remove the current button definitions.
Before doing this, please ensure that a copy of those button definitions has been made as suggested above.
6. Select **VrOne** from the **Application** option menu. The events for the twelve VrOne buttons will appear in the list box.
7. Select the VrOne button whose hot key you want to change in the box. Use the **Change Mapping**, **Add New Mapping**, or **Remove Mapping** buttons as desired. When you select **Change Mapping** or **Add New Mapping**, a dialog box will appear instructing you to press the key you wish to assign to the event.
8. Repeat the above step for each VrOne button that you wish to cross-reference.
9. Click **OK** to save the mappings or **Cancel** to exit without saving.
10. It is also possible to define unused device buttons to events in the **Main Image Display**. For example, it might be useful to assign the **Zoom In** and **Zoom Out** events to two unused buttons.
11. After defining the VrOne-SOCET button events, exit **SOCET SET** and save a copy of the current VrOne-SOCET key definitions:
`copy $HOME\socet_keys.localhost $HOME\vr_keys`
12. Then, if required, two batch files can be created with a text editor (and sent to the Desktop as short-cuts) to easily set the button definitions for the current SOCET SET session:
 - o **Create \$HOME\vrkeys.bat** (to re-initialize the VrOne SOCET button definitions)
 - o **Copy \$HOME\vr_keys \$HOME\socet_keys.localhost**
 - o **Create \$HOME\socet_keys.bat** (to re-initialize the original SOCET button definitions)
 - o **Copy \$HOME\original_keys \$HOME\socet_keys.localhost**

Starting VrOne

With SOCET SET running (and an absolute oriented model displayed), VrOne can be started by either:

- Selecting the **VrOne SOCET** icon from the desktop, or by
- Selecting **VrOne SOCET** from the main **SOCET SET** menu (**SOCET: Extraction > VrOne > VrOne SOCET**).

Exiting VrOne

The normal procedure for exiting VrOne can be done with any of the following:

- **VrOne: File > Exit**
- Enter **exi <Return>** in the **VrOne Key-In** dialog box.
- Clicking the **Close** button (X) in the upper right-hand corner of the **VrOne Main Window**.
- Double-clicking on the **VrOne icon** in the upper left-hand corner of the **VrOne Main Window**.

VrOne Help

VrOne has extensive local Web-based Help pages that can be accessed from the **About** pull down in the **VrOne Main Window**.

- **VrOne: About > Help** – Presents the top-level help and general usage selections.
- **VrOne: About > Global Help (key-in: ghelp)** – Lists and describes the global key-ins that are available.
- **VrOne: About > Local Help (key-in: help)** – Calls up the help section for the currently active VrOne command. For example, if the Reverse Line command is active, selecting **About > Local Help** or entering **help** in the **Key-In** dialog box will load the Reverse Line help page in your specified Web browser.

Opening an Existing VrOne File

While VrOne is running, one or more VR files can be opened with:

1. **VrOne: File > Open VR Files.**
2. Navigate to the directory containing the VR file(s) to be opened.
3. Using the mouse to select and the arrow buttons, move the files to the **Files to open** section in the right portion of the window
4. Press **OK** in the **Open VR File** window

Creating a New VrOne File

While VrOne is running, a new VR file can be created with:

1. **VrOne: File > Open VR Files**
2. Navigate to the directory into which the new file will be created.
3. Select the **New File** button in the lower-right portion of the window.
4. Enter the **New File Name**, when prompted
5. Press **OK** in the **Open VR File** window
6. Define the **VR file parameters** in the **Workspace** window.
 - o **Target Scale** (1:n): whether for feet or meters, enter the ratio. For example, 1:1200 (for 1"=100') or 1:2000 or ...
 - o **Units:** select US Foot or Meter or International Foot
 - o **Coordinate Format:** select State Plane or UTM
 - o **State plane zone**
 - o **UTM zone**
 - o **Description**

Closing an Existing VrOne File

VrOne files can be closed with:

1. **VrOne: File > Close VR Files.**
2. Using the mouse to select and the arrow buttons, move the files to the **VR Files to close** section button in the right portion of the window.
3. Press **OK** in the **Close VR File** window

Starting an Application (Command)

VrOne applications can be started through a number of different methods.

- Pulling down a menu and selecting an item may start commands.
- Commands may also be started from a key-in using the command name. Each command name is the first three letters of the first two words on the pull down. For example, the key-in command name for **Insert Line** is **InsLin**.
- Selecting the command from a pre-programmed input device, such as a Keypoint.

Custom Symbol and Line Font Creation

Define a New VrOne Symbol File

The default symbol and line font file is `\vr\data\default.sym`, but it is recommended that a custom file with a different name (e.g., `\vr\data\SomeOtherName.sym` or `D:\safe\baesystems.sym`) be created to prevent the inadvertent overwriting of the file during VrOne upgrading. Since the default VrOne symbol file contains numerous examples of symbols and line fonts, it may be more practical to copy the default file to a new name and location and then modify that file.

See Appendix B for a listing of the default VrOne line fonts and symbols.

The Symbol file to be used by VrOne is specified with

VR Configuration > Configuration > Vector Graphics > Symbol file name.

Create a New Symbol or Line Font

Custom symbol and line fonts are easily and quickly created with VrOne. A normal procedure would be as follows:

1. Start with a new (empty) VrOne file or go to a blank area in an existing file.
2. Turn the **Grid Display ON (Graphics > Grid On or `grion`)**. This allows for the symmetrical placement of the font components.
3. If necessary, set the **Grid Display Resolution (Graphics > Set Grid Resolution or `setgri`)** to a reasonable size.
4. Draw the components that will be included in the new symbol or line font entity with **Insert Line (Insert > Insert Line or `inslin`)**. A line font will contain the components needed to complete one complete unit of the font. (For example, the creation of a new fence line font only needs to include a single unit length of line with a single X.) The scale is not important, since symbols and line fonts are scaled at plot time.
5. Call up the **Create Symbol** application (**Insert > Create Symbol or `cresym`**).
6. Place a window box that includes the entities to be used for the new symbol or line font, and **Accept** or **Reject** the selection.
7. Digitize the origin of the new symbol or line font that corresponds to the center of a symbol or the beginning of the line font.
8. Specify whether it is a **new symbol or line font**.
9. Enter the **symbol or line font number** (1-1000).
10. Enter the **symbol or line font name**.
11. Enter the **symbol scale factor** (usually 1.0) or the repeating line font length (which defines the plotting length for one line font segment when plotting the line font).

Using a New Symbol or Line Font

A newly created symbol or line font can be placed into the current VrOne file by selecting **Insert Line (`inslin`)** or **Insert Symbol (`inssym`)** and manually entering the number of the new symbol or line font:

`grp=83` (or 27 or 473).

Or, as for other feature types, a function key containing the newly created symbol or line font number can be defined and used.

Zoom and Window Display Functions

Zooming into and out of the VrOne graphics window can be done from the VrOne menu pull-downs (**Graphics > Zoom**) or from key-in commands:

Zoom Center	<code>zooC</code>	Zoom In	<code>zoo.5</code>	Zoom Right	<code>zoor</code>
Zoom All	<code>zooa</code>	2x Zoom	<code>zool</code>	Zoom Down	<code>zood</code>
Zoom Left	<code>zool</code>	Last Zoom Window	<code>zoow</code>	Zoom Out 2x	<code>zoo2</code>
Zoom Up	<code>zoou</code>	Zoom Previous	<code>zoop</code>		

The **Zoom Left/Right/Up/Down** commands are also keyed to the four keyboard arrow keys. And the **Zoom In/Out** commands are keyed to the two keyboard **Page Down/Up** keys.

The graphics can be **replotted** to all VrOne windows with

Graphics > Replot `rep`

The VrOne **window rotation** can be defined with

Graphics > Set Window Rotation `winrot`

This is useful if the operator wants to have the VrOne cursor move and track in the same direction as the SOCET SET floating mark.

The **window rotation** can be set to a specific value (such as 0 or 30 or -45 degrees) with:

Environment > Set Parameters Rotation about the Z axis → 0 (or 30 or -45)

The `rotz` key-in command can also be used to define the **window rotation angle**:

`rotz 0` (or `rotz 30` or `rotz -45`)

Please note that the rotation defined by a `rotz` key-in command might also require that one of the above zoom commands be applied **before** the new rotation value will become apparent.

The VrOne window can be configured to **automatically re-center** whenever the cursor touches the edge of the graphics window with:

Graphics > Toggle Window Shift `togwin`

Additionally, this parameter can be set globally with:

Vr SOCET Config: Configuration > General Graphics > Window Shift > select Off or On

Zoom and Display Suggestions

Displaying Line Fonts

A normal behavior when VrOne entities are displayed to the screen is that lines appear to “lose” their fonts. This is “normal” because as the display scale gets smaller than a certain size, line fonts, symbols, and text will be displayed by VrOne in a simple format as unfonted lines (lines), single dots (symbols), and as lines (text).

The solution is to zoom in and out with the **Page Up** or **Page Down** keys, or to use **zoom window (zoow)** to zoom in to a smaller area, or to manually change the display scale (**Environment > Set Parameters > Scale (1:n): 2000**). As the display scale gets larger, the line fonts will re-appear.

VrOne Config (Global Default): Configuration > Vector Graphics > Minimum Text Size and Minimum Symbol Size

VrOne SOCET (Session Default): Environment > Edit Configuration > Minimum Text Size and Minimum Symbol Size

Also, issuing a **Zoom All (zooa)** command will display unfonted lines. Provided that the display scale is not too small, issuing a **Replot (rep)** command will display the line fonts.

Display Data Disappears with Zoom All

The **zooa** command (**Zoom All**) displays and fits the entire contents of the VR file to the graphics screen. If there's an outlier at say $x=0$ and $y=0$, the screen scale might be extremely small and no data would be visible. If this happens, one of the following might help:

1. Do a **Pack VR File (File > Pack VR File)** and then a **Zoom All**. This will permanently remove any deleted VR entities and will resize the XY file dimensions.
2. If step 1 doesn't work, and a small dot of data is visible on the screen, try to zoom into that area. If this is successful, use **Insert Line** and draw a box around the outside of the “good” area. Then use **Cut Polygon (Edit > Cut Polygon or cutpol)**, set the parameters to **Cut from outside** and **Throw data away**, pick the newly drawn box, and accept the action.
3. If it is not possible to see even a small dot of data, get the **Range XY (File > Range XY or ranxy)**, make note of the **Maximum X and Y**, and then use **Set Graphics Parameters (Environment > Set Parameters or setpar)** to define the lower left east and north and scale to “sane” values.

Data Collection

VrOne has numerous routines for the interactive insertion of line, symbol, and text entities. For example: **Insert Line**, **Insert Fly Line**, **Insert Parallel Line**, **Insert Symbol**, **Insert Annotated Symbol**, **Insert Text**, **Insert Line Text**, etc. These applications are typically found under the **Insert** pull down in the **VrOne Main Window**.

When an insertion application is started, a **Menu Keys** dialog window is displayed. At the top of the window is the **command name**. The next area displays the **entity parameters**. This information is command dependent and normally shows where the data is to be stored on insert commands or where the data came from on edit commands.

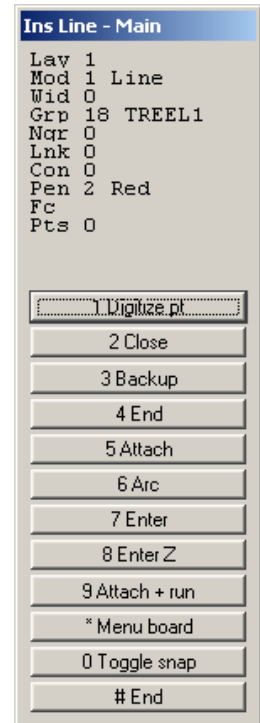
The third area is the **menu keys** area. These show what functions are assigned to the 12 menu key buttons. These keys are normally mapped onto an input device such as the keypad. They are also mapped onto the 12 function keys on a standard keyboard.

Following are a few examples for inserting various types of data into a VrOne file. Specific help can be called up for each of the insertion routines with:

VrOne: About > Local Help or key-in **help**

For these examples:

- Open a new VrOne file.
- Initialize the **Target Scale** to 1:1200 if using English units, or 1:1500 if using metric units.
- Using the **Page Down** and **Page Up** keyboard keys, set the **VrOne display scale** to approximately 1:1200 or 1:1500.
- Use the system mouse as the input device (**Input > Digitizer Mouse** or **digmou**).



Insert Line

The **Insert Line** function would be used to collect linear features where each data point is digitized manually, such as roads, sidewalks, fences, railroads, etc.

1. **Insert > Insert Line (inslin)**
2. Move the **system mouse cursor** into the **VrOne graphics** window. (While in this window, the cursor should change to a cross.)
3. While moving the mouse, press and release the left mouse button (or press the F1 key) to digitize points.
4. To **back-up** along the active line, press **Button 3** (Backup) or **F3**.
5. To **end** the line, press **Button 4** (End) or **F4**.
6. To **end** the **Insert Line** function, press the **# Button** or **F12**.
7. During the insertion of a line, **B2 (F2)** will *close*, but not end, the line and **B6 (F6)** will place an **Arc** through the last three digitized points.
8. The mode can be changed from **line** mode (mod=1) to **spline** mode (mod=2).
9. The **line font** (graphic pointer) can also be changed by entering **grp=18** (which is a treeline if using the default VrOne symbol file).
10. The active **layer** can be changed by entering **lay=4**

Insert Fly Line

The Insert Fly Line function would be used to collect linear features where the data points are automatically collected in a streamline mode, such as treelines, water features, contours, etc.

1. **Insert > Insert Fly Line** (`insfly`)
2. The **Fly Line filtering parameters** can be defined by pressing **B7 (F7)**, which will load a **Parameters** window with: **Tube Width**, **Maximum Distance**, **Dither Distance**, and **Z Tube Width**. As described in the local help for **Insert Fly Line**, these parameters affect and determine the frequency with which data points are sampled and stored.
3. Move the **system mouse cursor** into the **VrOne graphics** window. (While in this window, the cursor should change to a cross.)
4. While moving the mouse, press and release the **left mouse button** (or press the **F1** key) to automatically digitize points. The points will be sampled, filtered and stored based on the filtering parameters.
5. To **back-up** along the active line, press **Button 3** (Backup) or **F3**.
6. To **end** the line, press **Button 1** (End line) or **F1**.
7. During the insertion of a fly line, **B2 (F2)** will close and end the line. **Button 4 (F4)** will **Force** points that won't be removed by the filtering parameters. **B5 (F5)** allows for a **manually digitizing mode**, and **B6 (F6)** lets the operator go back along the active line and **splice** in a new section.
8. The mode can be changed from **line** mode (mod=1) to **spline** mode (mod=2).
9. The **line font** (graphic pointer) can also be changed by entering `grp=16` (which is a water line if using the default VrOne symbol file).
10. The active **layer** can be changed by entering `lay=6`

Insert Square

The **Insert Square** function is used to collect linear features where the internal segments are placed at right angles to each other.

- **Insert > Insert Square** (`inssqu`)
- The squaring parameters include options to square to the **Longest Side**, **First Side**, **Mean of All Sides** and **Azimuth**. A **Squaring Tolerance** can also be defined that tells **Insert Square** to not square segments that fall outside the specified angle.

Insert Symbol

Symbols can be placed into the VrOne file with this function.

- **Insert > Insert Symbol** (`inssym`)
- Symbols can be inserted with a fixed or pre-defined radius and rotation. Or the size and rotation can be interactively defined as part of the digitizing process.

Insert Text

Text can be placed into the VrOne file with this function.

- **Insert > Insert Text** (`instex`)
- Text can be inserted with a fixed or pre-defined size and rotation. Or the size and rotation can be interactively defined as part of the digitizing process.
- Ten different justifications can be used when placing text as well as numerous graphical attributes.

Insert Annotated Symbol

The Insert Annotated Symbol function is used to collect symbols with the inclusion of a text string, such as spot elevations, annotated fire hydrants, annotated manholes, etc.

- **Insert > Insert Annotated Symbol** (**insann**)
- The local help for this function lists the various key-ins commands that can be used when defining entity to be inserted. This is particularly useful when accessing this application from a Function Key definition.

See Appendix B for examples of the different types of Annotated Symbols.

Insert Line Text

The **Insert Line Text** function is used to insert text onto an existing line. This routine would typically be used to place annotations on index contours.

- **Insert > Insert Annotated Symbol** (**inslte**)

Changing the Graphical Attributes

Whenever a VrOne insertion command is started, the graphical attributes from the last usage are active. One of the ways in which these attributes can be changed is by manually keying in the new attribute(s). For example:

- To change the current **layer** (1-10001), enter **lay=4** (or **lay=125** or **lay=2065**, etc.)
- To change the current **line mode** (1-2), enter **mod=1** (line) or **mod=2** (spline)
- To change the current **line width** (1-256), enter **wid=1** (or **wid=2** or **wid=7**, etc.)
- To change the current **graphic pointer** (1-1000), enter **grp=11** (or **grp=24** or **grp=38**, etc.)
- To change the current **pen code** (1-256), enter **pen=3** (or **pen=4** or **pen=14**, etc.)

The better solution to the manual changing of VrOne graphical attributes is the use of pre-defined Function Keys, which is addressed later in this document.

Entity Search Radius

The VrOne entity search radius can be defined in two different ways:

1. As the default (global) value from within the VR Configuration program
VR Config: Configuration > Vector Graphics > Entity search radius (in ground units)
2. As a session value from within the VR SOCET program:
VrOne: Environment > Edit Configuration > Entity search radius (in ground units)

Data Edit

As with the insertion applications, VrOne also has numerous edit functions. For example: **Edit Line**, **Edit Symbol**, **Edit Text**, **Reverse Line**, **Join Lines**, **Square Existing**, **Trim Lines**, **Cut Polygon**, **Delete Layers**, and many more. These applications are typically found under the **Edit** pull down in the **VrOne Main Window**.

Edit Line

- **Edit > Edit Line** (**edilin**)
- **Edit Line** lets the operator move, delete, or insert new points along the selected line, and the line itself can be moved, copied, rotated, mirrored, or closed. Additionally, the graphical attributes can be modified with this routine.

Edit Symbol

- **Edit > Edit Symbol** (**edisym**)
- **Edit Symbol** lets the operator move, copy, delete, rotate, expand, or flip the selected symbol. Additionally, the graphical attributes can be modified with this routine.

Edit Text

- **Edit > Edit Text** (**editex**)
- **Edit Text** lets the operator move, copy, delete, rotate, expand, change justification, or flip the selected text. Additionally, the graphical attributes can be modified with this routine.

Trim Lines

- **Edit > Trim Lines** (**trilin**)
- **Trim Lines** will extend and cut-back one line to another. Options included **nodding**, **extension distance** and **polygon clearing**.

Break Line

- **Edit > Break Line** (**brelin**)
- **Break Line** will break a line into two entities at two user-identified points.

Reverse Line

- **Edit > Reverse Line** (**revlin**)
- **Reverse Line** will reverse the direction of lines. At times the order of the line points must be in a certain direction. This is true when drawing asymmetrically-fonted lines such as tree lines or depression contours. **Reverse Line** can reverse single lines by identifying individual lines, and it can batch reverse lines of user defined layers and modes.

Join Lines

- **Edit > Join Lines** (**joilin**)
- **Join Lines** will join or match two lines together. The lines to be joined are identified by the user and may reside the same workspace or any open workspace. Lines that are short of each other or overlap each other may be joined. If lines are joined, the second line is added to the first. The line parameters of the first line are used when joining. Lines that are matched are graphically joined but remain in two entities and maintain their graphic parameters. Lines identified in multiple workspaces will be matched regardless of the **Join Mode**.

Square Existing

- **Edit > Square Exist** (**squexi**)
- **Square Existing** squares existing line(s) in the current workspace or all open workspaces. Lines to square may be identified by pointing at them, or they may be squared by specifying layers in a batch mode. Line segments are rotated about their midpoints to the **Square to Azimuth** at 0, 90, 180 or 270 degrees and are re-intersected with the previous and next segments to square a polygon. Line segments that are greater than the **Square to Tolerance** are not squared (rotated).

Deleting Objects

Fast Delete

- **Edit > Fast Delete** (**fasdel**)
- With **Fast Delete**, the operator is able to delete entities by pointing at them.
- The **Search for Entity Type** can include lines and/or symbols and/or text.
- There are several procedures within VrOne that will delete entities: **Edit Line, Edit Symbol, Edit Text, Global Change** and **Delete Window**. But there are times when entities should be picked one by one for deletion. **Fast Delete** is a quick way to touch an entity and delete it.

Delete Window

- **Edit > Delete Window** (**delwin**)
- This function will delete the contents of a user-defined window. Only those entities that fall completely within the window will be deleted. Entities that are turned off will not be considered.

Cut Polygon

- **Edit > Cut Polygon** (**cutpol**)
- This function will cut data against a user defined polygon. Data can be cut from the inside or outside of the polygon, and the data can be discarded, copied to a file, moved to a file, or have the graphical pointer changed.

Selecting the Active Workspace

Up to 256 VR files can be open and displayed within VrOne, but only one file is “active” and it is into this file that data will be stored. The **Active Workspace** (VR file) can be specified with:

- **VrOne: Environment > Set Active Workspace**. Enter the number of the **active workspace (AWS)** and press **OK**.
- Another way to change the **Active Workspace** is to turn **Snap On** (**snaon**) and then select an entity in one of the non-active workspaces. The **Snap** window appears and, by pressing **button #7** (Change to Ws x), the user will be able to select the **AWS** in which the snapped-to object was found.

Batch Editing

VrOne also has numerous batch edit functions. For example: **Batch Attach Line, Batch Join Lines, Batch Node Lines, Batch Offset, Batch Trim, Line Cross, Line Close, Global Change**, and many more. These applications are typically found under the **Batch** pull down in the **VrOne Main Window**.

Batch Join Lines

- **Batch > Batch Join Lines (batjoi)**
- This batch routine will attempt to join lines within the layers specified. If more than one workspace is open and **Workspaces to consider** is set to **All open** then lines between workspaces will be matched. **Data layers, line modes, and graphic pointers** can also be considered. Care should be taken when setting the **Search Distance**. Undesirable joins may occur if this parameter is too large. Layers that are turned off will not be considered when searching.

Batch Trim

- **Batch > Batch Trim (battri)**
- This batch routine will make up to 60 different trimming passes on the specified VrOne workspaces and will extend and cut-back one set of lines to another, change graphics pointers, and will delete or relayer the trimmed segments.

Batch Node Lines

- **Batch > Batch Node Lines (batnod)**
- This batch routine will make up to 60 different nodding passes on the specified VrOne workspaces. Options include the ability to node lines only (which places a common point on each line at the intersection) or node and break lines at the intersection points. Under-shoots and over-shoots will be considered.
- The **Z rule** for computing the elevation at the intersection of the two lines can be specified. During nodding, an elevation for each line is computed at the break point. These elevations are based on the slope of the line segment as determined by the point before and after the break point. The **Z Rule** determines the action to be taken on these two elevations as follows:
 - o **Original elevations** – The elevations of the break points is not modified
 - o **Mean node elevations** – The elevations are averaged and stored for both node points.
 - o **Elevation from line 1** – The elevation from the node point on line 1 is used on both lines.
 - o **Elevation from line 2** – The elevation from the node point on line 2 is used on both lines.

Batch Filter Lines

- **Batch > Batch Filter Lines (batfil)**
- This batch routine will search for lines within the specified layers and will reduce the amount of points on those lines with a tube width and maximum distance filter. Layers that are turned off will not be considered when searching.

Batch Offset

- **Batch > Batch Offset (batoff)**
- This batch routine will search for lines found within the specified layers and will place up to 10 additional lines that are parallel to the original. Each new line can have a layer width, graphic point, horizontal offset and vertical offset that are different from the source line.

Line Close

- **Batch > Line Close (linclo)**
- This batch routine will close or will check for closure of lines in user defined layers. Lines that are not closed may be closed by one of the following methods:
 - o Move the endpoint onto start point
 - o Mean the endpoints
 - o Add line segment between the endpoint and the start point
 - o Intersect the end segments
 - o Report unclosed polygons in drive file only (do not modify line)
- A drive file is always created for unclosed lines even if this application closes them.

Global Change

- **Batch > Global Change** (**g1ob**)
- This extremely powerful batch routine will make up to 60 passes on the specified VrOne workspaces and will globally change line, symbol and text attribute information. Each pass has **Search For** and **Change To** criteria.

Undo Last

- **File > Undo Last** (**undo**)

Function Keys

The use of Function Keys provides one of the most important aspects of VrOne custom data collection. Taking the time to properly set up function keys before starting a project has the potential to save hundreds of hours of production time.

Function keys reduce the amount of knowledge the mapping professional needs to know about VrOne. They simply choose the feature they want to collect and go. Entering its name or function key number may start a function key. These names or numbers may be keyed in, placed in macros or even placed in other function keys.

For example, if the operator wanted to collect an asphalt road, the **Insert Line** command would be started, and the various attributes specific to that feature would need to be defined, such as Layer, Mode, Line Width, Line Font, and Pen Code. But VrOne function keys would allow for the pre-defining of this feature, so that the operator would simply enter a numeric code or a descriptive term for the asphalt road, which would automatically assign all of the desired parameters and behavior to **Insert Line** and start the command.

A more detailed description of VrOne Function Keys can be found at

VrOne: About > Help > Getting Started with VrOne > Function Keys.

The default function key file is `\vr\data\default.fk`, but it is recommended that a custom file with a different name (e.g., `\vr\data\SomeOtherName.fk` or `D:\safe\socet.fk`) be created to prevent the inadvertent overwriting of this file during VrOne upgrading. The Function Key file to be used by VrOne is specified with

VR Configuration: Configuration > General > Function key file

Touch Pad and Key-in Input

Just as the use of VrOne Function Keys facilitates the collection of specific project-related features by automatically assigning the desired parameters and behavior to one of the **Insert** commands, incorporating a touch pad input device with a feature extraction system adds even more efficiency to the collection capabilities.

The **Keypoint 252** is a touch-sensitive programmable keypad that can be used with VrOne. The 252 programmable keys can have the standard VrOne commands and VrOne Function Keys assigned to individual keys, which allows an operator to press a single key to activate a particular feature to be collected.

The default keypad file is `\vr\data\default.kp`, but it is recommended that a custom file with a different name (e.g., `\vr\data\SomeOtherName.kp` or `D:\safe\socet.kp`) be created to prevent the inadvertent overwriting of this file during VrOne upgrading. The Keypad file to be used by VrOne is specified with:

VR Configuration: Configuration > General > Keypad command file

The syntax of the keypad file (default.kp or socet.kp) is:

```
@xxx VrOne_KeyIn_Command
```

where **xxx** is the Keypad Number defined for a particular key (see “kpvr252_numbers.doc” for the key numbering reference) and **VrOne_KeyIn_Command** is the key-in command, FKey Name or FKey Number that VrOne recognizes for its various applications.

For example: **inslin** (for **Insert Line**), **instex** (for **Insert Text**), **rep** (for **Replot**), **zooa** (for **Zoom All**), **zoow** (for **Zoom window**), **exi** (for **Exit**), **bldg** (for **FKey Building**).

Macros

Macros give the ability to execute a group of VrOne commands as a single word. Macros can execute VrOne commands, function keys and other macros. One common use of macros is to have it set specific snap parameters for a certain situation. Also, with 10,001 layers that are available with VrOne, a macro can also be created to easily turn groups of layers on and off.

Application Overlay

Application Overlaying lets the operator place the current command into a “pause” mode while other commands are executed.

Multiple applications may be overlaid and stacked within VrOne which allows the operator to collect data in the order it presents itself instead of feature type by feature type. This unique VrOne feature allows for the collection of data in a flexible and realistic way.

Sample Files

To help with the initial use of VrOne, the following documents and files are available:

kp252vr_numbers.doc	Lists the assigned Keypad 252 numbers, which will be used to cross-reference VrOne commands to the keypad keys.
kp252.doc	Sample Keypad overlay.
keyin_xref.doc	Sample listing of VrOne key-in commands.
socet.fk	Sample VrOne SOCET Function Key file.
socet.kp	Sample VrOne SOCET Keypad file.
socet.pen	Sample VrOne SOCET Pen Table file.
socet.sym	Sample VrOne SOCET Symbol and Line Font file.

If desired, the above four **socet.*** files can be copied to a separate directory and used as customized data files. Please refer to the **General VrOne-SOCET Configuration - VrOne Configuration section** of this document for a more detailed explanation of the required configuration steps.

Troubleshooting Notes

VrOne Application Window Is Not Visible

If a VrOne application, such as **Insert Flyline** or **Batch Trim**, is executed and the corresponding application window isn't visible, the window might be placed *outside* the current graphics screen. The problem can be corrected by exiting VrOne and deleting the particular menu key file in the `\vr\hostdir` directory. The ***.mk** keys contain the last position of that window on the desktop.

For example, if the window for **Insert Flyline** doesn't appear, use **Windows Explorer** and delete the `\vr\hostdir\insflyline.mk` file.

Line Entities Aren't Displayed with Graphical Attributes

If the graphical attributes of one or all VrOne line entities are not displayed (i.e., one or more lines are displayed as solid white lines), the following can be used:

If **ALL** lines appear without graphical attributes, check and correctly define the following:

- **Environment > Set Display Parameters** (`setdis`)
- **Display Lines?** → Yes
- **Display Splined Lines?** → Yes
- **Display Line Fonts?** → Yes

If only **ONE** line (or just a few) appears without graphical attributes, enter the following:

- **Graphics > Set Display All** (`setall`)

Appendix A – Pen Table Colors

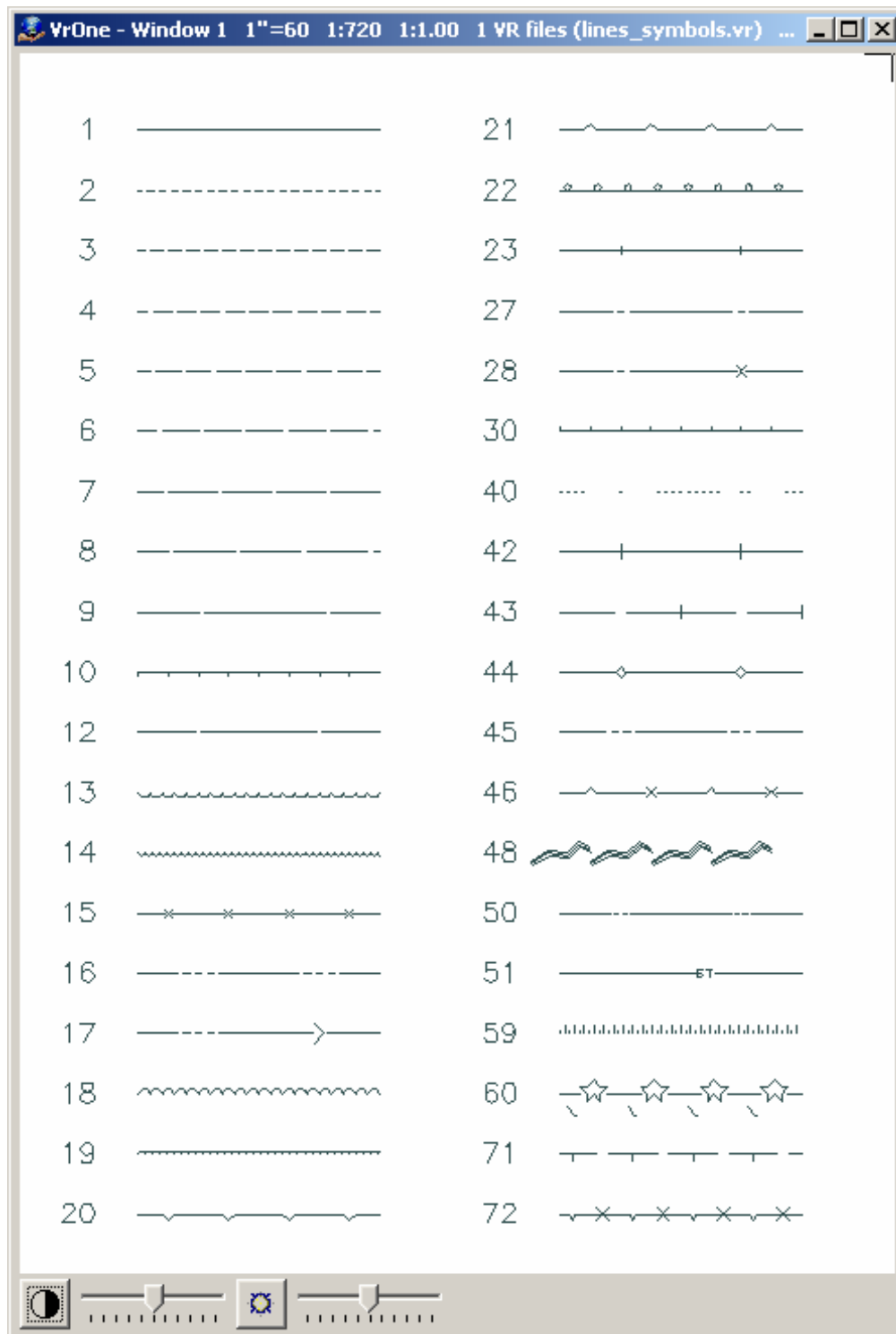
VrOne Pen Table Colors

Filename: \vr\data\defaultscreen.ct

Pen #	RGB	Color Name	Pen #	RGB	Color Name
1	255 255 255	White	36	0 250 154	MediumSpringGreen
2	255 0 0	Red	37	152 251 152	PaleGreen
3	255 255 0	Yellow	38	46 139 87	SeaGreen
4	0 255 0	Green	39	0 255 127	SpringGreen
5	0 255 255	Cyan	40	154 205 50	YellowGreen
6	0 0 255	Blue	41	47 79 79	DarkSlateGray
7	255 0 255	Magenta	42	47 79 79	DarkSlateGrey
8	192 192 192	Grey	43	105 105 105	DimGray
9	127 255 212	Aquamarine	44	211 211 211	LightGray
10	102 205 170	MediumAquamarine	45	192 192 192	Grey
11	95 158 160	CadetBlue	46	240 230 140	Khaki
12	100 149 237	CornflowerBlue	47	176 48 96	Maroon
13	72 61 139	DarkSlateBlue	48	255 165 0	Orange
14	173 216 230	LightBlue	49	218 112 214	Orchid
15	176 196 222	LightSteelBlue	50	153 50 204	DarkOrchid
16	0 0 205	MediumBlue	51	186 85 211	MediumOrchid
17	123 104 238	MediumSlateBlue	52	255 192 203	Pink
18	25 25 112	MidnightBlue	53	221 160 221	Plum
19	0 0 128	NavyBlue	54	205 92 92	IndianRed
20	0 0 128	Navy	55	199 21 133	MediumVioletRed
21	135 206 235	SkyBlue	56	255 69 0	OrangeRed
22	106 90 205	SlateBlue	57	208 32 144	VioletRed
23	70 130 180	SteelBlue	58	250 128 114	Salmon
24	255 127 80	Coral	59	160 82 45	Sienna
25	178 34 34	Firebrick	60	210 180 140	Tan
26	165 42 42	Brown	61	216 191 216	Thistle
27	255 215 0	Gold	62	255 255 240	Ivory
28	218 165 32	Goldenrod	63	64 224 208	Turquoise
29	238 221 130	MediumGoldenrod	64	0 206 209	DarkTurquoise
30	0 100 0	DarkGreen	65	72 209 204	MediumTurquoise
31	85 107 47	DarkOliveGreen	66	238 130 238	Violet
32	34 139 34	ForestGreen	67	138 43 226	BlueViolet
33	50 205 50	LimeGreen	68	245 222 179	Wheat
34	34 139 34	MediumForestGreen	69	255 255 255	White
35	60 179 113	MediumSeaGreen	70	173 255 47	GreenYellow

Appendix B

Default Line Fonts and Symbols (socet.sym)





Appendix C – Types of Annotated Symbols

Spot with Fixed Position Text

The image shows a screenshot of the 'VrOne Function Key Edit' dialog box. The title bar indicates the file path 'V:\docs\VrOne\For_Delivery\socet.fk'. The dialog is titled 'Function Keys' and shows a list of keys on the left, with 'spot_fixed' selected. The main area contains fields for 'Key name' (spot_fixed), 'Key num' (-1), and 'Vr function' (Insert Asym). Below this are tabs for 'Line', 'Symbol', and 'Text', with 'Text' selected. The 'Text' tab contains fields for 'Layer' (521), 'Font' (SR Simplex Ron), 'Just X' (Left), 'Just Y' (Lower), 'Non-graphic' (0), 'Height' (0.100), 'Pen number' (1), 'Width' (0.100), 'Link' (0), 'Rotation' (0.000), 'Construction' (Off), and 'Slant' (0.000). There is also a 'Text' field containing 'NULL'. At the bottom, there are fields for 'Local args' (INSSYM=1, IACTIVE=0, DEC=2, OFSX=-0.10, OFSY=0.1), 'Global args', 'Assoc cmd1', 'Assoc cmd2', 'On end cmd', and 'On quit cmd'. The dialog has 'Add', 'Change', 'Delete', 'Ok', and 'Cancel' buttons.

Function Keys

Key name: spot_fixed

Key num: -1

Vr function: Insert Asym

Description:

Line | Symbol | Text

Layer: 521

Font: SR Simplex Ron

Non-graphic: 0

Pen number: 1

Link: 0

Construction: Off

Feature code:

Text: NULL

Just X: Left

Just Y: Lower

Height: 0.100

Width: 0.100

Rotation: 0.000

Slant: 0.000

Local args: INSSYM=1, IACTIVE=0, DEC=2, OFSX=-0.10, OFSY=0.1

Global args:

Assoc cmd1:

Assoc cmd2:

On end cmd:

On quit cmd:

Buttons: Add, Change, Delete, Ok, Cancel

Spot Elevation with Manually Placed Text (freely positioned by the operator)

Function Keys

Key name: Vr function:

Key num: Description:

Line	Symbol	Text
Layer	<input type="text" value="522"/>	Just X: <input type="text" value="Left"/>
Font	<input type="text" value="SR Simplex Ron"/>	Just Y: <input type="text" value="Lower"/>
Non-graphic	<input type="text" value="0"/>	Height: <input type="text" value="0.100"/>
Pen number	<input type="text" value="1"/>	Width: <input type="text" value="0.100"/>
Link	<input type="text" value="0"/>	Rotation: <input type="text" value="0.000"/>
Construction	<input type="text" value="Off"/>	Slant: <input type="text" value="0.000"/>
Feature code	<input type="text"/>	
Text	<input type="text" value="NULL"/>	

Local args:

Global args:

Assoc cmd1:

Assoc cmd2:

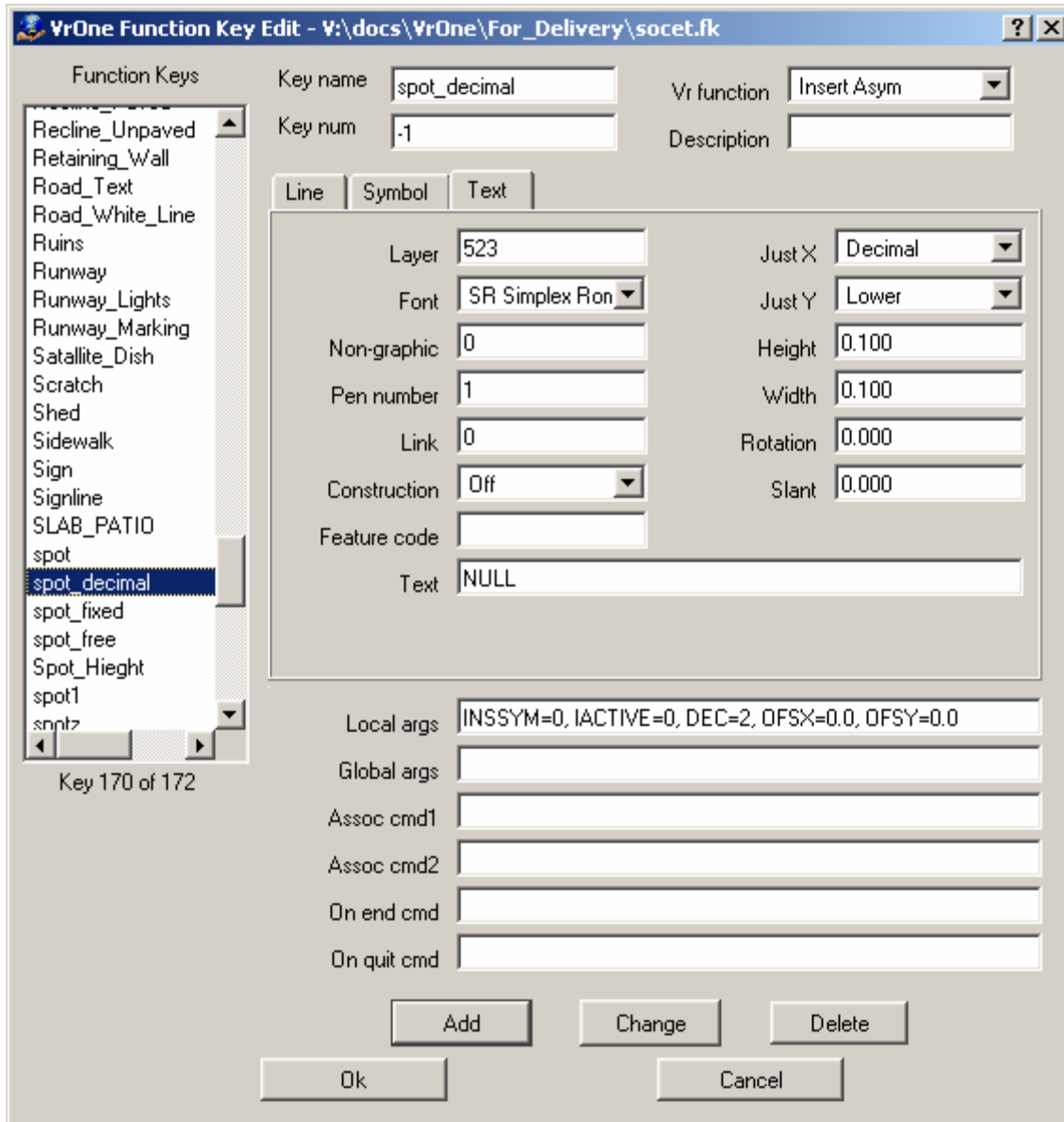
On end cmd:

On quit cmd:

Buttons: Add, Change, Delete, Ok, Cancel

Key 172 of 172

Spot Elevation where the Decimal Point Indicates the Location



Manhole with an “MH” String

Manhole: Symbol definition

The screenshot shows the 'VrOne Function Key Edit' dialog box. The title bar indicates the file path is 'V:\docs\VrOne\For_Delivery\socet.fk'. The 'Function Keys' list on the left includes 'mh-annotated', which is currently selected. The main configuration area is divided into several sections:

- Key name:** mh-annotated
- Key num:** -1
- Vr function:** Insert Asym
- Description:** (empty)

Below these fields is a table with three columns: 'Line', 'Symbol', and 'Text'. The table is currently empty.

The main configuration area contains the following fields:

- Layer:** 675
- Graphic:** 1
- Non-graphic:** 0
- Pen number:** 3
- Link:** 0
- Construction:** Off
- Radius:** 0.050
- Rotation:** 0.000
- Feature code:** (empty)

Below the main configuration area are several text input fields:

- Local args:** INSSYM=1, LABSOR=1, IACTIVE=0, OFSX=.12, OFSY=.12
- Global args:** (empty)
- Assoc cmd1:** (empty)
- Assoc cmd2:** (empty)
- On end cmd:** (empty)
- On quit cmd:** (empty)

At the bottom of the dialog are five buttons: 'Add', 'Change', 'Delete', 'Ok', and 'Cancel'.

Manhole: Text definition

Function Keys

Key name: Vr function:

Key num: Description:

Line Symbol Text

Layer: Just X:

Font: Just Y:

Non-graphic: Height:

Pen number: Width:

Link: Rotation:

Construction: Slant:

Feature code:

Text:

Local args:

Global args:

Assoc cmd1:

Assoc cmd2:

On end cmd:

On quit cmd:

Key 173 of 173

Buttons: Add, Change, Delete, Ok, Cancel

Fixed Text String

Function Keys

Key name: Pond_Text

Key num: -1

Vr function: Insert Text

Description:

Line	Symbol	Text
	Layer	635
	Font	SR Simplex Ron
	Just X	Center
	Just Y	Center
	Non-graphic	0
	Height	0.100
	Pen number	1
	Width	0.100
	Link	0
	Rotation	0.000
	Construction	Off
	Slant	0.000
	Feature code	
	Text	Pond

Local args:

Global args:

Assoc cmd1:

Assoc cmd2:

On end cmd:

On quit cmd:

Key 174 of 174

Add Change Delete

Ok Cancel

Appendix D – Key-In Cross Reference

Insert Line	inslin	Copy Window	copwin	Backup VR File	bacvr
Insert Symbol	inssym	Set DTM	setdtm	Open VR File	opevr
Insert Text	instex	Run DTM	rundtm	Close VR File	clovr
Insert Square	inssqu	Drive Grid	drigri	Replot	rep
Insert Parallel Line	inspar	Z DTM	z dtm	Zoom Left	zool
Insert Line Text	inslte	Label Bear/Distance	labbea	Zoom Center	zooc
Insert File	insfil	Grid Lines	grilin	Zoom Right	zoor
		Tick Lines	iclin	Zoom Previous	zoop
Edit Line	edilin	Inverse	inv	Line Font ON	lfon
Edit Symbol	edisym	Drive To	dri	Line Font OFF	lfoff
Edit Text	editex	Redrive	redri	Spline ON	splon
Trim Line	trilin	Reset Drive	resdri	Spline OFF	sploff
Break Line	brelin	Delete Duplicates	deldup	Line Points ON	lpon
Hack Line	haclin	Batch Filter	batfil	Line Points OFF	lpoff
Point on Line	pol	Batch Trim	battri	Line Elevations ON	leon
Reverse Line	revlin	Batch Offset	batoff	Line Elevations OFF	leoff
Join Lines	joilin	Sheet Grid	shegri	Range XY	ranxy
Insert Flyline	insfly	Plot Control	plocon	Edit VR Header	edivr
		Create Symbol	cresym	UNDO	undo
Interpolate Contours	intcon	Zoom All	zooa	Zoom In 4X	zoo .25
Square Existing	squexi	Zoom In 2X	zoo .5	Zoom Down	zood
Hatch Line	hatlin	Zoom Up	zoo	Zoom Out 4X	zoo 4
Label Point	labpoi	Zoom Out 2X	zoo 2	Pop	pop
Symbol Fill	symfil	Zoom Windows	zoow	Pop ALL	popall
Edit Elevation	ediele	Toggle Grid	toggri	Swap Applications	swa
Fast Delete	fasdel	Set Grid Resolution	setgri	Associate Command 1	ac1
Line Cross	lincro	Toggle Window Shift	togwin	Associate Command 2	ac2
Change Layer	chalay	Set Display Parameters	setpar	Z Source	zsou
Change Graphic Pointer	chagra	Set Display Parameters	setdis	Plot	plo
Change Pen Code	chapen	Set Layer	setlay	Help	help
Global Change	glob	Reset Layer	layres	Global Help	ghelp
Cut Polygon	cutpol	List Layers	lislav	Edit Macros	edimac
Delete Window	delwin	Edit Config Params	edicon	Edit Function Keys	edifun
Move Window	movwin	Pack VR File	pacvr	EXIT	exi

Index

- Active workspace, 17
- Annotate contours, 14
- Annotated symbol, insert, 14
- Annotated symbols
 - types, 26
- Application
 - overlay, 1, 21
 - starting an, 8
- Attributes, graphical, 14
- Automatic window
 - recenter, 10
- Available colors, 5
- Batch edit
 - close line, 19
 - filter lines, 18
 - join line, 18
 - node line, 18
 - offset, 18
 - trim, 18
- Break line, 15
- Button definitions, 6
- Change
 - active workspace, 17
 - global, 20
 - graphical attributes, 14
- Close
 - line, 15, 19
 - VrOne file, 8
- Collect. See Insert
- Colors, 4
- Configuration
 - general, 4
 - keyport, 4, 21
 - SOCET SET, 5
 - VrOne, 4
 - XYZ digitizer, 4
- Contour annotation, 14
- Create
 - new VrOne file, 7
 - symbol file, 9
 - symbol or line font, 9
- Cross reference
 - button definitions, 6
- Custom files, 4, 9, 22
- Cut inside, 17
- Cut outside, 17
- Cut polygon, 11, 17
- Data collection. See Insert
- Data edit. See Edit
- Database, 1
- Delete
 - cut polygon, 17
 - fast, 17
 - inside, 17
 - outside, 17
 - window, 17
- Digitizer, 4
- Digmou, 4
- Digxyz, 4
- Display
 - data disappears, 11
 - line fonts, 11
 - suggestions, 11
- Download VrOne, 2
- Edit
 - batch. See Batch edit
 - break line, 15
 - close line, 15, 19
 - global change, 20
 - join lines, 16
 - line, 15
 - reverse line, 15
 - symbol, 15
 - text, 15
 - trim lines, 15, 18
- Entity search radius, 14
- Error messages, 2
- Events, 6
- Exiting VrOne, 7
- Failed to start, 2
- Fast delete, 17
- Fence delete, 17
- File format, 1
- Filtering, 13, 18
- Fit all, 10
- Fly line
 - filtering, 13
 - insert, 13
- Function keys, 14, 20, 26
- Global change, 20
- Graphical attributes, 14
 - not displayed, 22
- Graphics replot, 10
- Help, 7, 12
- History, 1
- Input device, 4
- Insert
 - annotated symbol, 14
 - fly line, 13
 - line, 12
 - line text, 14
 - square, 13
 - symbol, 13
 - text, 13
- Install VrOne, 2
- Integration, 2
- Introduction, 1
- Join line, 16, 18
- Kappa rotation, 10
- Key definitions, 6
- Keyboard keys, 10, 12
- Keyboard/Accelerator
 - mapping, 6
- Keyport, 4, 21
- Label contours, 14
- License request, 3
- Line
 - batch close, 19
 - batch join, 18
 - break, 15
 - close, 15
 - edit, 15
 - insert, 12
 - join, 16
 - node, 15, 18
 - parallel, 18
 - reverse, 15
 - trim, 15, 18
- Line fonts
 - default, 24
 - display, 11
- Line text, insert, 14
- Macros, 21
- Menu key files, 22
- Missing DLL, 2
- New
 - symbol file. See Create symbol file
 - symbol or line font. See Create symbol or line font
- Node line, 15, 18
- Open VrOne file, 7
- Over-shoots, 15, 18
- Overview, 1
- Pack VR file, 11
- Parallel lines, 18
- Pen tables, 4
- Peuker filter, 13, 18
- Place. See Insert
- Polygon delete, 17
- Range XY, 11
- Recenter window, 10
- Replot graphics, 10
- Reverse line, 15
- Rotate window, 10
- Sample files, 22
- Screen display colors, 5
- Screen.ct, 23
- Search path, 2
- Search radius, 14
- Selecting the active workspace, 17
- Set window rotation, 10
- Small display scale, 11
- Snap distance, 14
- SOCET SET
 - button definitions, 6
 - configuration, 5
 - DLL search path, 2
- Socet.fk, 22
- Socet.kp, 22
- Socet.sym, 24
- Square
 - existing, 16
 - insert, 13
- Start with XYZ
 - stereoplotter, 4
- Starting an application, 8
- Starting VrOne, 7
- Stream digitizing, 13
- Symbol
 - edit, 15
 - insert, 13
- Symbol file. See Create symbol file
- Symbols, default, 24
- Table of contents, 2
- Text
 - edit, 15
 - insert, 13
- Touchpad, 21
- Trim lines, 15, 18
- Troubleshooting
 - no graphical attributes, 22
 - window isn't visible, 22
- Tube filtering, 13, 18
- Types of annotated symbols, 26
- Unable to locate component, 2
- Under-shoots, 15, 18
- Undo, 20
- Use pen table, 4
- View rotation, 10
- VrOne
 - button definitions, 6
 - configuration, 4
 - database, 1
 - download, 2
 - exit, 7
 - help, 7, 12
 - install, 2
 - integration, 2
 - license request, 3
 - start, 7
- VrOne file
 - close existing, 8
 - create new, 7
 - format, 1
 - open existing, 7
- Vrone.acc, 6
- Window
 - automatic recentering, 10
 - delete, 17
 - not visible, 22
 - rotation, 10
- XYZ digitizer, 4
- Zoom
 - commands, 10
 - suggestions, 11