



DATREND
Systems Inc.

vPad-VTTM

venTest Plugin for vPad-Check App

Plugin Version 1.00.0

Operating Manual

vPad-VT™
venTest Plugin for vPad-Check App
Operating Manual

© 2017 Datrend Systems Inc.
Unit 130 - 4020 Viking Way
Richmond, BC • CANADA • V6V 2L4
Tel: 800.667.6557 (North America Only) or
604.291.7747 • Fax 604.294.2355
e-mail: customerservice@datrend.com



Revision	Revision History Description	Date
A	Initial release	2017-Oct-03

Copyright

Datrend Systems Inc. ("DSI") agrees to a limited copyright release that allows you to reproduce manuals and other printed materials for use in service training programs and other technical publications. If you would like other reproductions or distributions, submit a written request to Datrend Systems Inc.

Unpacking and Inspection

Follow standard receiving practices upon receipt of goods. Check the shipping carton for damage. If damage is found, stop unpacking. Notify the freight carrier and ask for an agent to be present while the goods are unpacked. There are no special unpacking instructions, but be careful not to damage the goods when unpacking. Inspect the goods for physical damage such as bent or broken parts, dents, or scratches.

Claims

Our routine method of shipment is via common carrier. Upon delivery, if physical damage is found, retain all packing materials in their original condition and contact the carrier immediately to file a claim. If the goods are delivered in good physical condition but does not operate within specifications, or if there are any other problems not caused by shipping damage, please contact your local sales representative or DSI immediately.

Standard Terms and Conditions

Refunds & Credits

Please note only serialized products (products labelled with a distinct serial number) and accessories are eligible for partial refund and/or credit. Non-serialized parts and accessory items (cables, carrying cases, auxiliary modules, etc.) are not eligible for return or refund. In order to receive a partial refund/credit, the product must not have been damaged, and must be returned complete (meaning all manuals, cables, accessories, etc.) within 90 days of original purchase and in "as new" and resalable condition. The *Return Procedure* must be followed.

Software which has been activated may not be returned for credit. Use it in Demo mode where possible to ensure suitability for purpose.

Return Procedure

Every product returned for refund/credit must be accompanied by a Return Material Authorization (RMA) number, obtained from Datrend Customer Service. All items being returned must be sent *prepaid* (freight, duty, brokerage, and taxes) to our factory location.

Restocking Charges

Product returned within 30 days of original purchase are subject to a minimum restocking fee of 15%. Product returned in excess of 30 days after purchase, but prior to 90 days, are subject to a minimum restocking fee of 20%. Additional charges for damage and/or missing parts and accessories will be applied to all returns. Product which is not in "as new" and resalable condition, are not eligible for credit return and will be returned to the customer at their expense.

Certification

This instrument was thoroughly tested and inspected and found to meet DSI's specifications when it was shipped from the factory.

Where applicable, calibration measurements are traceable to the National Research Council of Canada (NRC) and/or the National Institute of Standards and Technology (NIST). Devices for which there are no NRC/NIST calibration standards are measured against in-house performance standards using accepted test procedures.

Warranty

Warranty and Product Support

Software Applications (Apps) for use in conjunction with ES601 and vPad Safety Analyzers and other Datrend Products are supplied subject to the Terms and Conditions agreed to by the act of Activating the App. In the event no Activation is required, the Terms and Conditions published on the Datrend Systems Inc. (DSI) website shall prevail. By using the App, you agree to these terms and Conditions. IN NO EVENT SHALL DATREND SYSTEMS INC. BE LIABLE FOR CONSEQUENTIAL DAMAGES.

DSI DISCLAIMS ALL OTHER WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE OR APPLICATION.

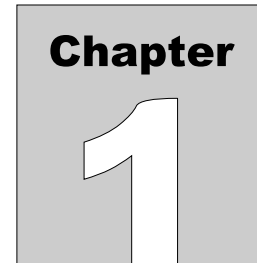
vPad™, vPad-Check™, vPad-ES™, vPad-353™, vPad-AS™, vPad-A3™, vPad-A1™, vPad-RF™, vPad-Record Manager™, vPad-RM™, vPad-EQM™, vPad-Cal™, vPad-mT™, vPad-XPORT™, Datrend Docs™ and CMX™ are trademarks of Datrend Systems Inc.

Android™ is trademark of Google Inc.

Table of Contents

TABLE OF CONTENTS.	III
1 OVERVIEW.	1
2 INSTALLATION.	3
3 OPERATION.	5
3.1 vPad-VT Setup.	5
3.2 venTest Setup.	9
3.2.1 Test Connections.	9
3.2.2 Display Setups.	9
3.2.3 Display Programming.	10
3.2.4 File Synchronization.	11
3.3 Running a Ventilator Checklist.	12
3.3.1 Checklist Startup.	12
3.3.2 Inspection Prompts and Checks.	13
3.3.3 Electrical Safety Test.	14
3.3.4 Performance Tests.	15
3.3.4.1 Initialization.	15
3.3.4.2 Loading of setups.	17
3.3.4.3 Measurements.	18
3.3.4.4 Flow-pressure waveforms.	19
3.3.5 Results Preview.	20
4 VENTEST PLUGIN PROGRAMMING.	21
4.1 plugin statement.	22
4.2 connect command.	24
4.3 disconnect command.	24
4.4 autopass command.	25
4.5 ident command.	25
4.6 search command.	26
4.7 load command.	27
4.8 read command.	28
4.9 graph command.	31
5 UPGRADES AND MAINTENANCE.	33

venTest Plugin for vPad-Check



1 Overview

vPad-Check™ is a software application (app) which may be installed on the vPad series of electrical safety analyzers, and which may be used to automate preventive maintenance (PM) inspections of ventilators and other types of medical equipment.

vPad-VT™ is an optional *software plugin* which enables vPad-Check to connect with and control the *venTest™* ventilator test system. vPad-VT, also referred to as the "venTest plugin", enables PM inspections performed with a vPad safety analyzer to include venTest measurements such as tidal and minute volumes, PEEP and plateau pressures, ventilation rate, I:E ratio and other tests of ventilator performance.

The plugin communicates with the venTest analyzer wirelessly, by means of the *venTest Bluetooth Adaptor*. The venTest Bluetooth Adaptor is an optional hardware module for the venTest which enables interface with vPad safety analyzers, and which may be connected to either one of the USB Host Ports located on the rear panel of the venTest unit.

This manual describes setup of vPad-VT and how ventilator PM checklists may be written in vPad Programming Language or "vPL" to include tests with the venTest analyzer. For more information on vPad-Check, PM checklists and vPL programming, refer to **MN-084 vPad-Check Operator's Manual**.




NOTE: For compatibility with vPad-VT, vPad-Check must be **Version 1.95** or higher, and the venTest must have these firmware versions (or later versions) installed:
GUI: **D2.0.2.3-15**
Main: **D2.0.2.3-14**
Flow sensor: **D2.0.2.3-12**

venTest Plugin for vPad-Check

2 Installation

vPad-VT or the venTest plugin is accessory software for vPad-Check which adds venTest connectivity to vPad-Check. Thus, to control the venTest analyzer from a vPad safety analyzer, you must have both the vPad-Check app and the optional venTest plugin installed.

Normally, vPad tablets are provided with the vPad-Check app and vPad-VT pre-installed. The icon for vPad-Check may be apparent on the tablet's HOME screen, however, plugin icons are usually *not* placed on the HOME screen to avoid clutter.

Enter the Apps Menu  of the Android tablet and confirm that icons for both vPad-Check and vPad-VT are present in the list of apps:



In order to use the vPad-Check app, it will need to be activated. When vPad-Check is started for the first time, the activation screen will be presented. Enter the activation code provided by your dealer to allow the app to run.

The venTest plugin does not require activation, however, vPad-Check requires installation and activation of at least one app for electrical safety testing; this may be any one or more of vPad-AS, vPad-NFPA, vPad-A3 or vPad-A6.

In the event that either vPad-Check or vPad-VT is not on the tablet, contact your dealer and an installation package for the app will be provided via download.

venTest Plugin for vPad-Check

3 Operation

This chapter describes how vPad-Check is used to perform a ventilator PM inspection in which a venTest analyzer, under the control of the venTest plugin, is used for performance testing.

This chapter does not cover all aspects of the vPad-Check app, but focusses instead on setting up the plugin to work with the venTest, and on ventilator test features added to vPad-Check by the plugin. If necessary, refer to **MN-084 vPad-Check Operator's Manual** for general instructions on use of vPad-Check for automated PM inspection and testing.

3.1 vPad-VT Setup

Use the USB cable provided with the venTest Bluetooth Adaptor (Datrend P/N 7400-116) to connect the Adaptor to a USB Host Port located on the rear panel of the venTest analyzer (*Figure 1*). The venTest provides two such USB Ports, and either may be used for connection of the Adaptor.




Figure 1 - Connection of adaptor 7400-116 to venTest USB.

Apply power to the venTest unit. After startup, verify the venTest displays a "RS1" or "RS2" indication at the top of the touchscreen (*Figure 2*). This indicates the venTest has detected the Adaptor at USB Port #1 or Port #2 respectively, and is ready to communicate through it.

venTest Plugin for vPad-Check



Figure 2 - Adaptor connected to USB port 1.

Apply power to the vPad tablet and wait until the device displays the HOME screen. Enter the Android settings menu  and verify "Bluetooth" in the list on the left is set to "ON". If it is not ON, press the toggle button to make it so.

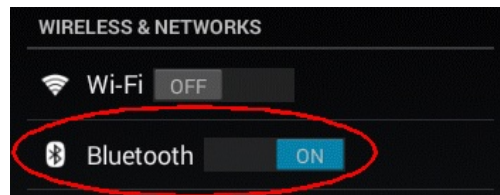


Figure 3 - vPad Bluetooth 'ON'.

Select the "Bluetooth" item on the left, then select "SEARCH FOR DEVICES" near top right on the tablet screen.



Figure 4 - Select Bluetooth search.

The vPad will perform a search for nearby Bluetooth devices which are powered up and running. When the search finishes, scroll the right side of the touchscreen down to view "AVAILABLE DEVICES". Select the venTest Bluetooth Adaptor, which has an ID that begins with "DSI-VENT- ..." (Figure 5). When prompted for a PIN, enter "1234" and press "OK".

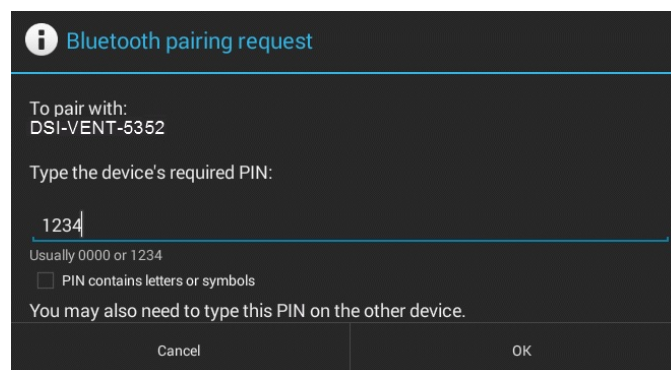


Figure 5 - Enter venTest Bluetooth PIN.

venTest Plugin for vPad-Check

Once the vPad tablet has been successfully paired with the venTest, exit Android Settings with the BACK

button



. From the HOME screen, enter the Android App



, then find and run

venTest plugin app



The plugin will display a "venTest Bluetooth setup" menu which will most likely be showing the ID of the venTest Bluetooth Adaptor.



Figure 6 - venTest plugin setup menu.

If the ID is not correct, press the **Bluetooth ID** selector near the middle of the setup menu and choose the Adaptor's ID from the pick list. Press the **Save** button to save the ID.



NOTE: On first run of the plugin, you must always press **Save** to store the ID, even if the ID initially displayed by the menu is the correct one.

After saving the ID, perform a Bluetooth connection test by pressing the **Test** button.

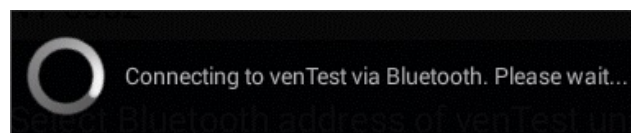


Figure 7 - Bluetooth connection test.

venTest Plugin for vPad-Check

If the plugin is able to contact the venTest, a pop-up message will appear after a few seconds which identifies the venTest analyzer by its serial number.

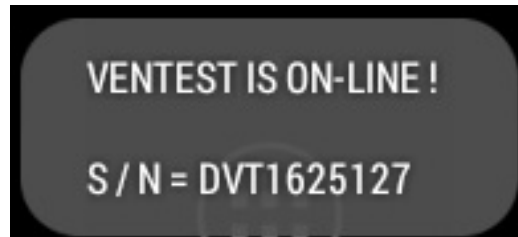


Figure 8 - Successful Bluetooth test.

If the plugin cannot contact the venTest, an error dialog is shown.

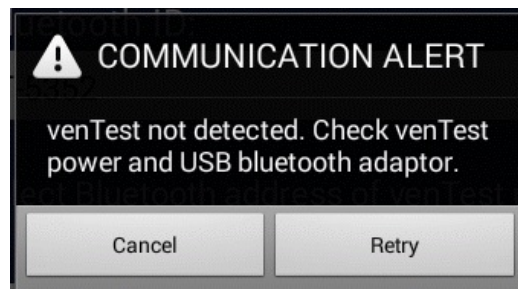


Figure 9 - Unsuccessful Bluetooth test.

Follow the instructions on the dialog then press the **Retry** button. If the plugin fails to connect, press **Cancel** and confirm the selected **Bluetooth ID** is correct, then retest. If errors persist, you may try unplugging and replugging the venTest Bluetooth Adaptor and/or cycling the power on the venTest.

After testing, press the **Exit** button to close the plugin's setup menu.

3.2 venTest Setup

3.2.1 Test Connections

For testing of adult ventilators, the venTest will typically be connected as shown in the schematic of *Figure 10*. The FSH module, which incorporates a gas flow sensor as well as a pressure sensor, is connected in-line between the ventilator output and a test lung. Additional pressure sensors and/or gas concentration sensors may be used according to test requirements. Power for the ventilator may be obtained from the test receptacle of the vPad safety analyzer as shown. For further instructions on ventilator connections and setup, refer to in **MN-108** *venTest Operator's Manual*.

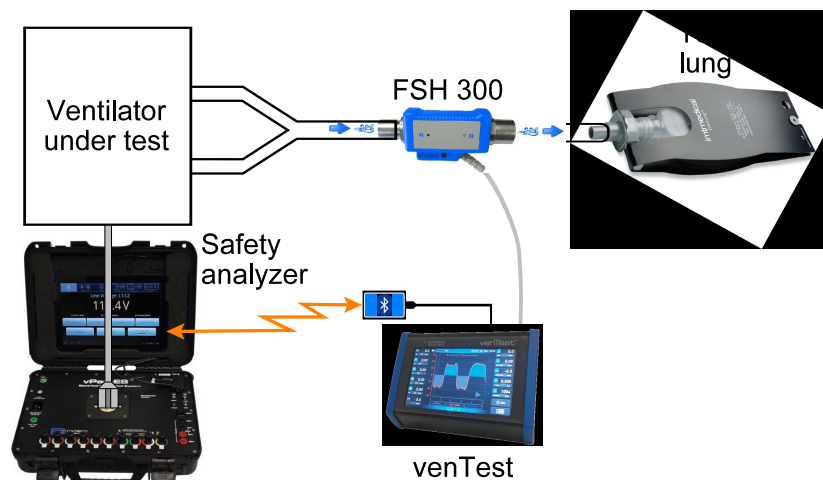


Figure 10 - Setup for ventilator inspection.

3.2.2 Display Setups

The venTest can make use of up to ten different sensors to measure 28 parameters related to ventilator performance. These parameters include flow, pressure, volume and gas concentrations. Environmental parameters such as barometric pressure and humidity can also be measured.

The venTest's touchscreen provides numeric readouts to display up to eleven of the 28 available parameters at once (*Figure 11*). A chart in the middle of the touchscreen displays two signals in real-time as they are being output from sensors connected to the venTest, or which are internal to the analyzer. The real-time curves are interpreted by the venTest to produce the numeric results shown, including tidal and minute volumes, peak flows and pressures, and timing measurements such as ventilation frequency and inspiration-expiration ratio.

venTest Plugin for vPad-Check



Figure 11 - venTest touchscreen display.

The venTest user can arbitrarily select and assign specific measurement parameters to the eleven readouts. This is done by pressing a readout and then choosing a desired parameter and its units of measure from a menu. It is also possible to set a readout to display no parameter. The user can choose which sensor signals are displayed by the real-time chart and can set ranges for the horizontal and vertical axes. This is done by pressing the chart and then choosing the graph settings from a menu. In addition to readout and chart selections, a setup may include flow and/or pressure trigger levels which are important for derivation of the ventilation volumes and timing results. After configuring the display, the user can assign a name to the settings and then save the resulting *Display Setup* in the venTest's memory.

The above is a brief summary of venTest Display Setups and how they are generally configured and saved. Additional information may be found in **MN-108** *venTest Operator's Manual*.

3.2.3 Display Programming

Before a ventilator can be tested, the user must program a suitable Display Setup for the test and then save it to venTest's memory. Readout and chart settings, and possibly trigger levels, are chosen by the user to suit the types of performance measurements which have been specified by the ventilator manufacturer in the OEM's inspection procedure and/or service manual. Such programming is beyond the scope of this manual, but further guidance may be found in manual **MN-108** as well as the "*Introduction to venTest*" training video which may be seen at the Datrend website (www.datrend.com/support/videos).

venTest Plugin for vPad-Check

3.2.4 File Synchronization

When a ventilator checklist is executed by vPad-Check, the venTest will be commanded by the vPad safety analyzer to recall one or more Display Setups from memory. If the venTest does not have a required Setup in its memory, an error message will be displayed by vPad-Check and further execution of the ventilator inspection beyond that step will be inhibited.

Accordingly, synchronization of ventilator checklists stored in the vPad tablet and Display Setups stored in venTest memory is essential. This is a data synchronization issue which is analogous to the synchronization issues around electrical safety testing, in which a checklist will launch a testing app such as vPad-NFPA and will need that app to have the required AutoSequence setup that is to be executed as part of the inspection.

To facilitate the distribution of Display Setups, the venTest provides a number of file transfer options, for example, file transfer by SD card; refer to **MN-108** for further information.

3.3 Running a Ventilator Checklist

vPad-Check and the venTest plugin work together to facilitate preventive maintenance (PM) inspections of ventilators in which performance tests are completed automatically or semi-automatically with a venTest analyzer. In such inspections, vPad-Check calls on the plugin as needed to interface with the venTest; unlike the case in which an electrical safety test is launched from a checklist and the test is then handled by a different App, when the venTest plugin is operating, vPad-Check will remain visible to the user at all times.

This section describes steps which may be seen when running a typical checklist for a ventilator. These steps include: user instructions (prompts); inspection checkpoints (checks); an electrical safety test; and performance measurements.

3.3.1 Checklist Startup

To perform a ventilator PM, an appropriate checklist is found by scrolling the list on the main menu of vPad-Check and is then selected by pressing the row (*Figure 12*). After starting the inspection, the user will be prompted to enter information about the ventilator, such as the control number or equipment ID (*Figure 13*). The **Next Step** button increments the user through the PM inspection one step at a time, displaying instructions and requesting input of information when required by the procedure.

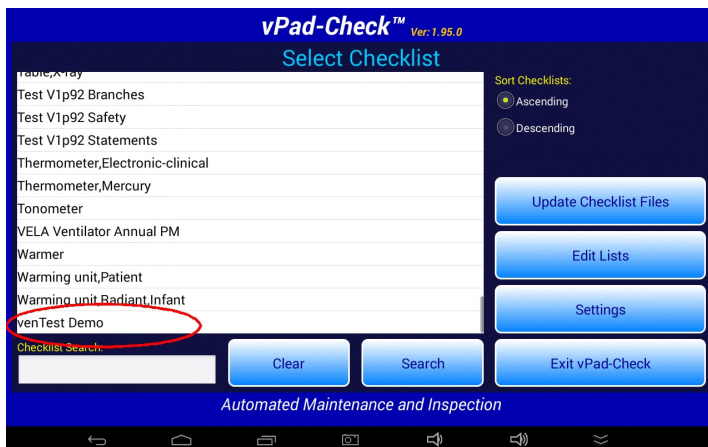


Figure 12 - vPad-Check main menu.

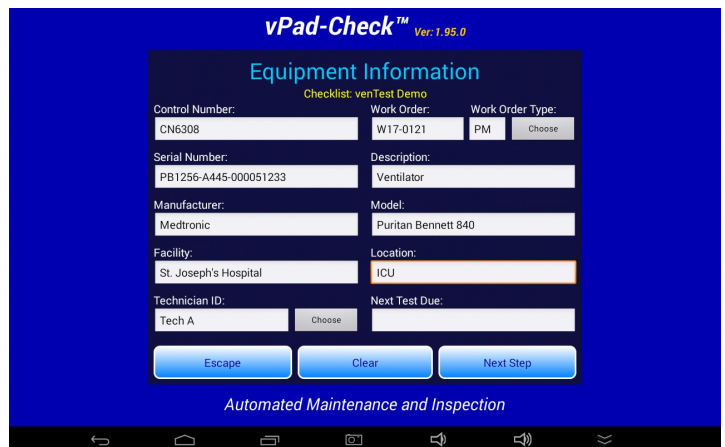


Figure 13 - Entering ventilator information.

venTest Plugin for vPad-Check

3.3.2 Inspection Prompts and Checks

Similar to other checklists which may be executed with vPad-Check, a ventilator inspection will include user instructions or "prompts" (*Figure 14*), and inspection points which will be shown on the touchscreen as "checks" with PASS/FAIL buttons (*Figure 15*).

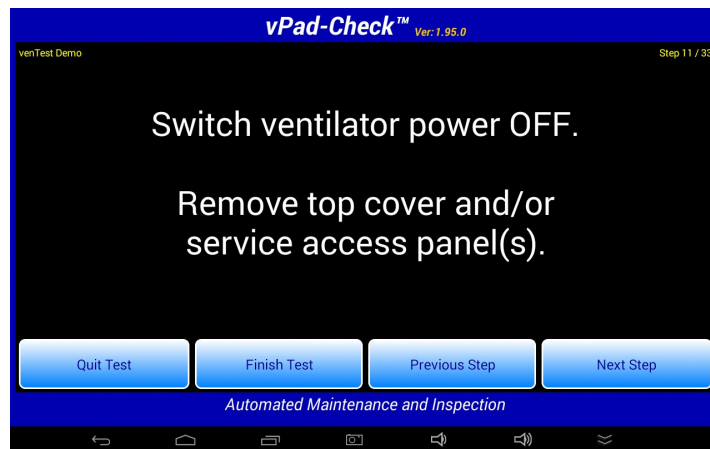


Figure 14 - Inspection prompt.

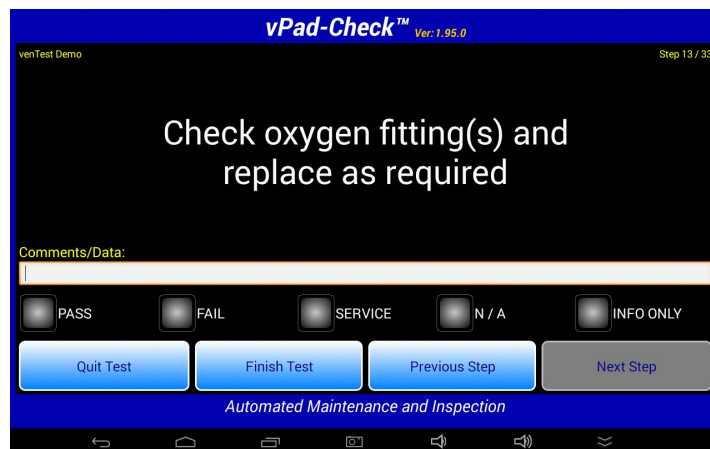


Figure 15 - Inspection checkpoint.

Prompts and check steps can be configured to closely follow a manufacturer's inspection procedure, typically including steps such as visual inspection of the ventilator's controls, indicators and components, replacement of worn parts either preventatively or as required, and tests of the internal battery and its charging system. Through checklist programming as described in Chapter 4 of **MN-084 vPad-Check Operator's Manual**, inspection checkpoints can be optionally configured to display images or video when needed.

venTest Plugin for vPad-Check

3.3.3 Electrical Safety Test

Most ventilator inspections will include an electrical safety test in which, at minimum, ground resistance and chassis leakage of the ventilator are evaluated for conformance to a standard such as NFPA 99, IEC62353 or IEC60601. A ventilator checklist will perform such tests by launching a safety testing app, for example vPad-NFPA as shown in *Figure 16*.

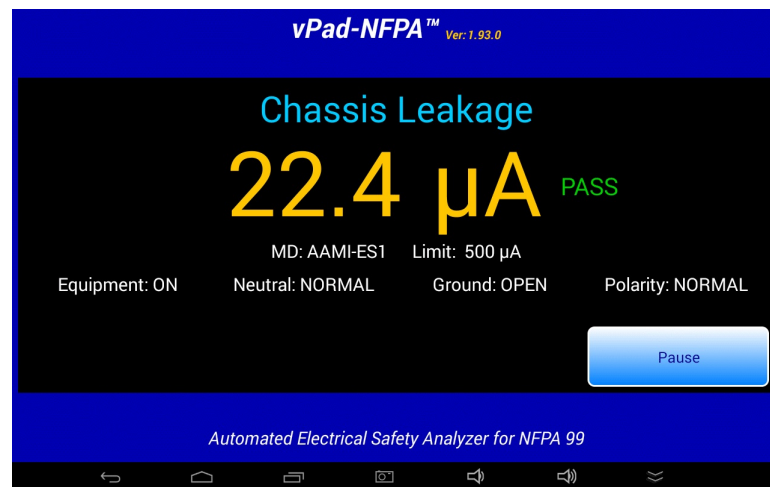


Figure 16 - Automated electrical safety test.

Electrical safety measurements are performed automatically, and the results are then incorporated into the test record for the PM inspection (*Figure 17*).



Figure 17 - Safety test results.

venTest Plugin for vPad-Check

3.3.4 Performance Tests

Following visual inspections and an electrical safety test, a ventilator PM may include tests of the ventilator's performance. Performance testing with the venTest will typically comprise a number of steps, including: venTest initialization; loading of a Display Setup (see section 3.2.2); and recording of various measurements while the ventilator is running. It is also possible to capture real-time waveforms from the venTest display and save them as PNG image files on the vPad tablet.

3.3.4.1 Initialization

In preparation for ventilator performance testing, the checklist will execute a number of steps to initialize the venTest analyzer (*Figure 18*). First, the checklist will command the vPad safety analyzer to connect with the venTest by Bluetooth. Once the venTest is on-line, the checklist will then search the venTest to confirm Display Setups required to complete the PM are currently stored in the venTest's memory. "Auto-PASS" mode may also be enabled during the initialization phase. Initialization steps are automated, and provided each step is completed successfully, the checklist will increment through the steps without user intervention.



Figure 18 - Bluetooth connection, Display Setup verification, and enabling of test automation.

venTest Plugin for vPad-Check

While establishing the Bluetooth connection to the venTest, a progress dialog is displayed by the vPad tablet. This connection normally requires a few seconds to complete.

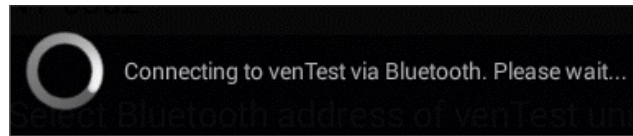


Figure 19 - Bluetooth connection dialog.

If the vPad tablet is able to connect, the checklist will advance to the next step automatically. As shown in *Figure 18*, the venTest memory will then be searched to confirm Display Setups required for the PM are currently stored in the analyzer.

If the vPad tablet cannot contact the venTest, a COMMUNICATION ALERT is displayed.

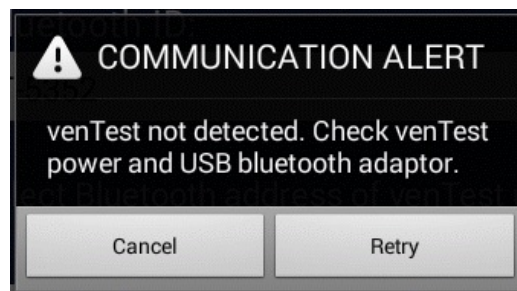


Figure 20 - Connection alert.

If the COMMUNICATION ALERT appears, follow the instructions on the dialog then press the **Retry** button. If errors persist, you may try unplugging and replugging the venTest Bluetooth Adaptor and/or cycling the power on the venTest.

Referring to *Figure 18*, "auto-PASS" mode is enabled by the checklist after Display Setups have been verified. "Auto-PASS" is a form of test automation in which a measurement result within specified limits causes the checklist to advance to the next step automatically after a short delay. Such automation can speed up performance testing significantly, particularly when there are a large number of readings to be checked. By default, "Auto-PASS" mode is disabled at startup of an inspection and is explicitly enabled by the checklist when required, for example as shown in *Figure 18*.

Any error experienced during venTest initialization will cause the PM to be stopped at the step where the problem occurs. The user will be unable to proceed until the problem is corrected, or alternatively, the user chooses to abort the inspection in progress. However, the user is not restricted from backing up to a previous step, then retrying the current step once the problem has been overcome. In the case of a Display Setup which is not found during a search, the user may

venTest Plugin for vPad-Check

choose to program and save the missing setup on the fly, or alternatively transfer the missing setup to the venTest by means of an SD card (see **MN-108** *venTest Operator's Manual* for details).

Because initialization problems may result in an aborted PM, ventilator checklists should be constructed so that these steps are completed at or near the beginning of the inspection procedure. See APPENDIX A of this manual for an example checklist in which initialization steps are performed just after the procedure title and introductory instructions are displayed to the user.

3.3.4.2 Loading of setups

At certain steps of the PM, the checklist may command the venTest to load a specified Display Setup. Per section 3.2.2, a Display Setup configures venTest readouts and trigger levels that will permit the required measurements to be executed successfully at subsequent steps of the PM. Provided the Display Setup file exists, the venTest is configured and the checklist then advances to the next step automatically (*Figure 21*).

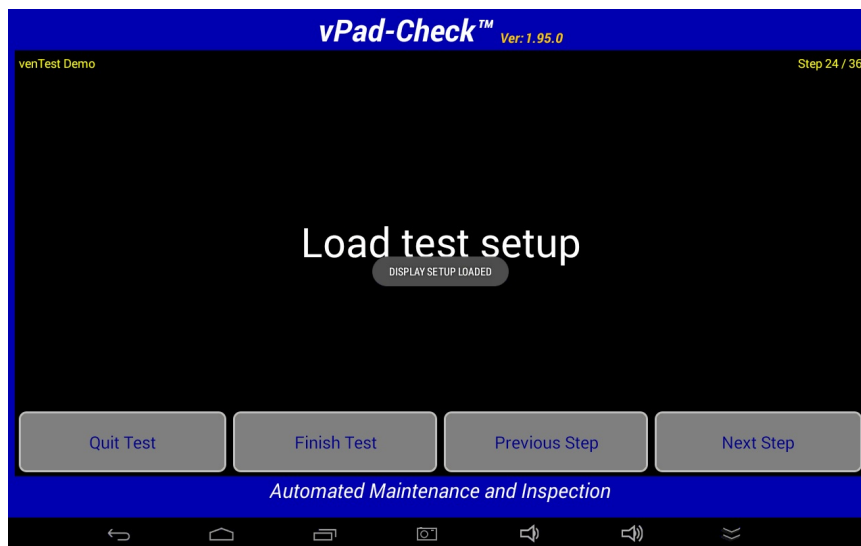


Figure 21 - Successful load of venTest Display Setup.

If the specified Display Setup cannot be found in venTest memory, an error message is displayed and further execution of the checklist is prevented. However, the user is not restricted from backing up to a previous step, then retrying the current step once the missing setup has been installed in the venTest. The user may choose to program and save the missing setup on the fly, or alternatively transfer the setup to the venTest by means of an SD card (see **MN-108** *venTest Operator's Manual* for details).

3.3.4.3 Measurements

In a performance test, the user will be instructed to set the ventilator controls and will then allow the ventilator to run for a period of time. Once the ventilator has stabilized, the checklist may then proceed to acquire measurements of ventilator flow, pressure, volume, gas concentrations and other parameters (*Figure 22*). Each acquired measurement is compared to upper and lower acceptance limits which are defined in the checklist. A test is assigned a "PASS" or "FAIL" depending on whether the value is within the limits in the checklist.

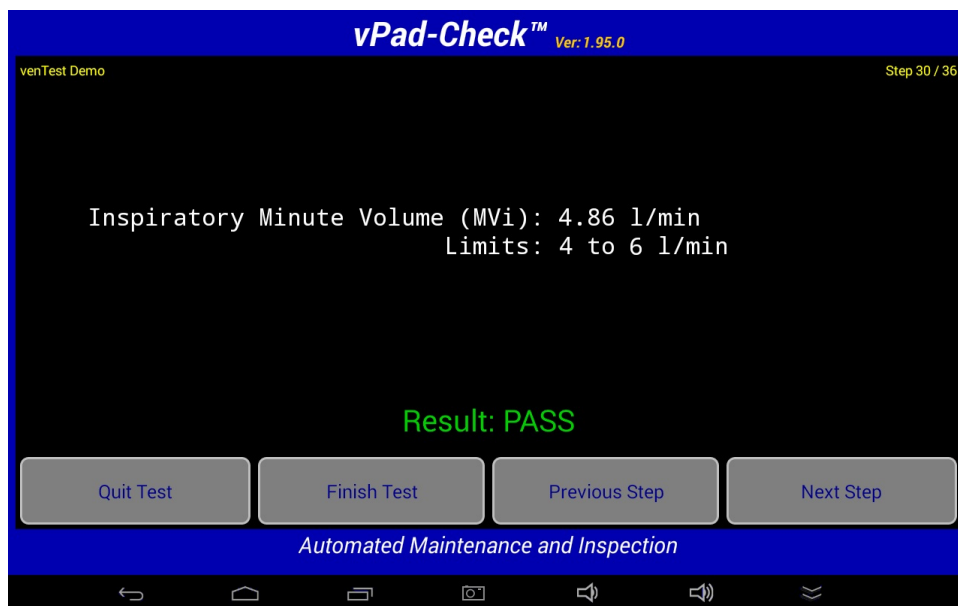


Figure 22 - Ventilator measurement within limits.

If the result is a "PASS" and the "auto-PASS" mode has been enabled as described in section 3.3.4.1, the checklist advances to the next step automatically after a short delay. It is possible for a checklist to execute a completely automated sequence of measurements in this manner, requiring no intervention from the user other than to change ventilator settings when instructed to do so by the checklist.

venTest Plugin for vPad-Check

3.3.4.4 Flow-pressure waveforms

A checklist can command the venTest to download a copy of the real-time chart being shown on the venTest's screen (*Figure 23*). The venTest graph may be then saved as a PNG image file on the vPad tablet.

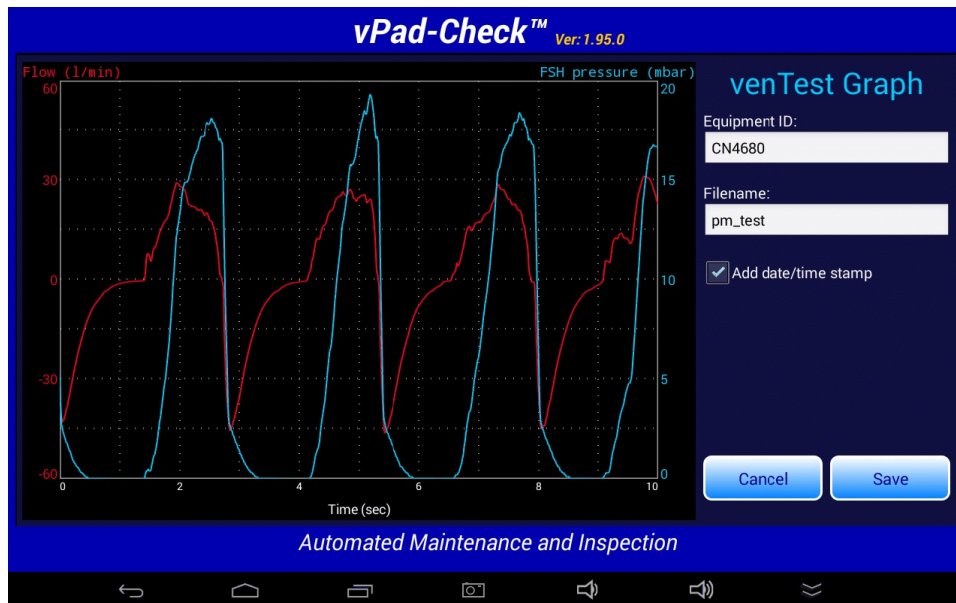


Figure 23 - Ventilator waveform capture by vPad-Check.

Edit fields to the right of the graph enable the user to optionally modify the equipment ID entered at test startup (see section 3.3.1), and a default filename which is defined in the checklist. A checkbox below the edit fields enables a date/time stamp to be appended to the filename, if desired.

The **Cancel** button closes this menu without saving the image. The **Save** button saves the PNG image to this file path:

Datrend/vPad-Check/venTest/<Equipment ID>/<Filename>

where the "Equipment ID" and "Filename" are read from their respective fields shown to the right of the graph, with a date/time stamp appended to the end of the "Filename" if enabled by the checkbox. After the file is saved and the menu closed, the folder and filename information is then copied to the test record for later reference.

venTest Plugin for vPad-Check

3.3.5 Results Preview and Test Record


When all inspections and tests of the ventilator checklist have been completed, a preview of the results may then be displayed (*Figure 24*). After the results preview is closed with the **OK** button, a test record file containing the ventilator equipment information and test results may then be saved with the **Create Record** menu (*Figure 25*).



Figure 24 - vPad-Check main menu.

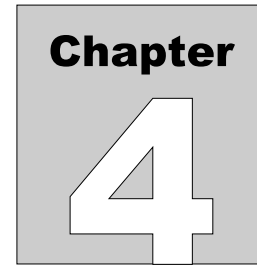


Figure 25 - Entering ventilator information.

Test record files are stored by vPad-Check in the **Datrend/vPad-Check/Records** folder on the tablet. Records may be viewed, printed, converted to PDF documents or transferred to a PC or cloud storage by means of the vPad Record Manager app .

For additional information on use of the vPad-Check , see **MN-084 vPad-Check Operator's Manual**.

For information on the vPad Record Manager, see **MN-087 vPad Record Manager Operator's Manual**.



4 venTest Plugin Programming

vPad "checklists" correspond to a set of files which are stored in the vPad-Check directory of the Android tablet. A checklist file, identified by its **vp1** file extension, contains a *computer program* which has been written in the vPad Programming Language or "*vPL*".

Each step of a checklist, for example a ventilator inspection procedure as described in section 3.3 of this manual, has a direct one-to-one relationship with a corresponding vPL *statement* in the checklist file. Statements that begin with a **prompt** keyword can display instructions to the user, for example as shown by *Figure 14* on page 13. Statements based on the **check** keyword (*Figure 15*, page 13) display PASS and FAIL radio buttons and wait for the user to choose an inspection result.

This manual does not cover all aspects of the vPad Programming Language but only those aspects which are relevant to interface with the venTest analyzer through the vPad-VT plugin. Accordingly, this chapter focusses on only one statement from the vPL language - **plugin** - and the associated **<command>** instructions that are passed by the **plugin** statement to vPad-VT to set up and control the venTest.

For more general information on programming checklists in vPL with examples, refer to Chapter 4 of **MN-084 vPad-Check Operator's Manual**.

4.1 plugin statement

Syntax: **plugin** **<plugin_name>** | **<description>** | **<command>**

UI reference: Sections 3.3.4.1 through 3.3.4.4 inclusive

Description: The **plugin** statement causes vPad-Check to launch a *software plugin* to execute a function or a test, as specified by a **<command>** word which is included in the statement.

Arguments of the **plugin** statement are:

<plugin_name> This is a required argument which specifies the software plugin to be launched by vPad-Check. The venTest plugin, which interfaces with the venTest analyzer, is referred to in **plugin** statements as **vPad-VT**.

<description> This is a required argument and it is meant to be a description of the function or test that is performed by the statement. The **<description>** is arbitrary and will be shown on the vPad screen as the statement is being executed. If the purpose of the **<command>** is to obtain a measurement value or other test result, then the **<description>** will be copied to the "Task" field of the test record when the test result is saved. Thus, the **<description>** is chosen by a checklist author to give a result in the test record some context.

<command> This is a required argument which is passed to the plugin when the plugin is launched.

Analogous to a vPL statement, the **<command>** will begin with a keyword and may include arguments, all of which will be meaningless to vPad-Check but which will be understood by the plugin. The plugin executes the **<command>** and within a few seconds or less will return information to vPad-Check. The information may be a test result, or it may be an instruction that causes vPad-Check to continue with the next step of the checklist.

While executing the **<command>**, plugins may display temporary messages to the user, or they may post dialogs that require the user to respond by pressing a button. Such messages and dialogs are

venTest Plugin for vPad-Check

displayed "over top" of vPad-Check and will appear to have been produced by the checklist app rather than by the plugin. This is irrelevant as far as a user is concerned; vPad-Check and its plugins are designed to work together to provide what appears to be seamless behaviour from a user's perspective.

Commands are specific to the plugins that execute them. The remainder of this chapter will specify the commands that are used with the vPad-VT plugin.

Examples:

```
plugin    vPad-VT | "Connect to venTest via Bluetooth" | connect

plugin    vPad-VT | "Confirm venTest memory has all\nDisplay Setups "\n+\n"required for this PM" | search:Vent_PM_Setup;Pressures_Setup

plugin    vPad-VT | "Load ventilator test setup" | load:Vent_PM_Setup

plugin    vPad-VT | "Check maximum flow" | read:PFi;20;40
plugin    vPad-VT | "Check tidal volume" | read:VTi;200;350
plugin    vPad-VT | "Check minute volume" | read:MVi;4;6
plugin    vPad-VT | "Check maximum pressure" | read:Peak;7;10
plugin    vPad-VT | "Check PEEP" | read:PEEP;0.5;1.5
plugin    vPad-VT | "Check ventilation rate" | read:Rate;18;22
plugin    vPad-VT | "Check I:E ratio" | read:I/E;0.8;1.4

plugin    vPad-VT | "Download flow/pressure graph" | graph:PM_test

plugin    vPad-VT | "Disconnect Bluetooth" | disconnect
```

4.2 connect command

Command syntax: **connect**

UI reference: Section 3.3.4.1

Example of usage: **plugin vPad-VT | "Connect to venTest via Bluetooth" | connect**

Description: This command establishes a Bluetooth connection between the vPad tablet and the venTest. Once connected to the venTest, the vPad tablet remains connected until explicitly disconnected with the **disconnect** command.

This statement should appear first in a checklist, after which other vPad-VT plugin statements may then follow. The statement should be placed at or near the beginning of the checklist, prior to completion of any significant inspection work or other tests. This will confirm the venTest will be ready when performance measurements are required.

If a plugin statement with **connect** command is omitted (usually by mistake), other **plugin** statements may attempt to make a Bluetooth connection before executing the specified command. Regardless, a properly-written checklist will always include a **connect** statement near the beginning of a procedure.

4.3 disconnect command

Command syntax: **disconnect**

UI reference: None

Example of usage: **plugin vPad-VT | "Disconnect venTest Bluetooth" | disconnect**

Description: Once a Bluetooth connection has been made with the venTest, this command will be used to disconnect once all performance tests have been completed. The statement is usually placed at or near the end of a checklist.

4.4 autopass command

Command syntax: **autopass : [on|off]**

UI reference: Section 3.3.4.1

Usage: **plugin vPad-VT | "Enable auto-PASS mode" | autopass:on**
or:
plugin vPad-VT | "Disable auto-PASS mode" | autopass:off

Description: "Auto-PASS" is a form of test automation in which a measurement result within specified limits causes the checklist to advance to the next step automatically after a short delay. Such automation can speed up performance testing significantly, particularly when there are a large number of readings to be checked. By default, "Auto-PASS" mode is disabled at startup of the checklist and must be explicitly enabled with an **autopass : on** command, as above.

If auto-PASS mode is desired, it will usually be enabled near the beginning of the checklist, immediately or shortly after the initial **connect** command (see 4.2).

4.5 ident command

Command syntax: **ident**

UI reference: None

Example of usage: **plugin vPad-VT | "Get venTest traceability info" | ident**

Description: This statement is used to read the ID (serial number) of the venTest unit. On receipt, the venTest ID is copied to the test record and is assigned an "INFO ONLY" result. This statement is useful when there is a institutional requirement to include test equipment traceability information in maintenance records.

This statement may be placed anywhere in a checklist, following the initial **connect** command (see 4.2).

4.6 search command

Command syntax: **search:<setup1>;[setup2];[setup3];[setup4];[setup5]...**

UI reference: Section 3.3.4.1

Example of usage: **plugin vPad-VT | "Check venTest memory for installed setups"\+
| search:Factory default;Vent_PM_Setup;Anesthetic_Gas_Setup**

Description: This statement is used to search the contents of the venTest memory and verify that each Display Setup specified in the **search** command exists. The list of Display Setups is delimited from the **search** keyword by a full colon (:), and the individual setups to be located are delimited by one or more semicolons (;) as required. The command must specify at least one setup.

If all setups specified in the command are found, the checklist will be advanced to the next step automatically.

If a setup cannot be found, the missing setup is reported to the user by an error message posted on the vPad screen. In this case, execution of the checklist beyond the current step is inhibited. However, the user is not restricted from backing up to a previous step, then retrying the current step once the missing setup has been installed in the venTest. The user may choose to program and save the missing setup on the fly, or alternatively transfer the setup to the venTest by means of an SD card (see **MN-108** *venTest Operator's Manual* for details).

This statement should be placed close to the beginning of the checklist, prior to completion of any significant inspection work or other tests. This will confirm the venTest will be ready when performance measurements are required. Accordingly, an appropriate place to put the statement is immediately following the initial **connect** command (see 4.2).

4.7 load command

Command syntax: **load:<display_setup>**

UI reference: Section 3.3.4.2

Example of usage: **plugin vPad-VT | "Load PM test setup" | load:Vent_PM_Setup**

Description: This statement is used to load a specified **<display_setup>** from venTest memory, thereby configuring the venTest's readouts, trigger levels and real-time chart for a given performance test.

If the **<display_setup>** is successfully loaded from venTest memory, the checklist will be advanced to the next step automatically.

If the **<display_setup>** is not found in venTest memory, the missing setup is reported to the user by an error message posted on the vPad screen. In this case, execution of the checklist beyond the current step is inhibited. However, the user is not restricted from backing up to a previous step, then retrying the current step once the missing setup has been installed in the venTest. The user may choose to program and save the missing setup on the fly, or alternatively transfer the setup to the venTest by means of an SD card. Note this type of error is preventable through use of a **search** command near the beginning of a checklist (see section 4.6).

4.8 read command

Command syntax: **read:<mnemonic>;<low_limit>;<high_limit>**

UI reference: Section 3.3.4.3

Example of usage: **plugin vPad-VT | "Check minute volume" | read:MVi;4;6**

Description: This statement retrieves a measurement value as it is currently being shown on the venTest display, and compares the value to lower and upper limits. Although there are 28 measurements that could potentially be displayed by the venTest, at any given time the number of measurements that will be accessible to the **read** command is limited to the number of venTest readouts (which is eleven), with the measurement shown by each readout depending on the Display Setup which has been loaded (see sections 3.2.2, 3.3.4.2 and 4.7 respectively).

If the retrieved measurement is within the upper and lower limits, it is assigned a "PASS". If the value is outside the limits, the result is assigned a "FAIL". The measurement and pass/fail result is saved to the test record when the **Next Step** button is pressed. If "auto-PASS" mode is enabled per 4.4, the checklist saves the result and increments to the next step automatically if the result is a "PASS".

Arguments of the **read** command are:

<lower_limit>	Limit values are decimal numbers. Units of measure are implicit and must agree with the units of measure according to the Display Setup in use.
<upper_limit>	
<mnemonic>	This is a mnemonic code which represents the measurement value to be retrieved from the venTest. Measurements which may be potentially displayed by venTest are specified in TABLE 4.8-1 on the following page, along with their corresponding read mnemonics. The mixture of upper and lower case letters in mnemonics is a venTest convention; when used in a read command, mnemonics are not case-sensitive.

TABLE 4.8-1: Ventilator measurement mnemonics

read command mnemonic	Measurement description
Flow	Instantaneous flow
Peak Flow Insp	Peak inspiratory flow (PFi)
Peak Exp Flow	Peak expiratory flow (PFe)
VTi	Inspiratory tidal volume
VTe	Expiratory tidal volume
MVi	Inspiratory minute volume
MVe	Expiratory minute volume
Frequency	Ventilation rate or frequency
PEEP	Positive end-expiratory pressure
Peak	Peak pressure
Plateau	Plateau pressure
Mean	Average pressure
I/E	Inspiration-expiration ratio (I:E)
Ti	Inspiration time
Te	Expiration time
Tip	Inspiration pause time
Tep	Expiration pause time
HFO Frequency	HFO frequency
HFO I/E	HFO inspiration-expiration ratio
HFO Pmean	HFO average pressure
HFO Pamp	HFO pressure amplitude
HFO Pmax	HFO maximum pressure
HFO Pmin	HFO minimum pressure
Humidity	Relative humidity
O2	Oxygen concentration [O2]

venTest Plugin for vPad-Check

read command mnemonic	Measurement description
CO2	Carbon dioxide concentration [CO2]
N2O	Nitrous oxide concentration [N2O]
Agas	Anesthetic gas concentration [PA]

To streamline coding of vPL checklists, abbreviated mnemonics may be used for some of the venTest parameters as shown in TABLE 4.8-2 below. As with the mnemonics of TABLE 4.8-1, these abbreviations are not case-sensitive.

TABLE 4.8-2: 'read' mnemonic abbreviations

read command mnemonic	Alternative abbreviated mnemonic(s)
Peak Flow Insp	PFi
Peak Exp Flow	PFe
Frequency	Freq <i>or</i> Rate
HFO Frequency	HFO Freq <i>or</i> HFO Rate

For real-time pressure measurements, the **read** command must identify which venTest sensor is to be used for the measurement. Pressure-related mnemonics are specified in TABLE 4.8-3 below. Like other ventilator measurements, pressure-related **read** commands are not case-sensitive.

TABLE 4.8-3: Pressure measurement mnemonics

read command mnemonic	Pressure measured by...
Pr(FSH)	FSH sensor
Pr(FSL)	FSL sensor (venTest accessory option)
Pr(20 mbar)	±20 mbar sensor
Pr(200 mbar)	±200 mbar sensor
Pr(1 bar)	±1 bar sensor
Pr(10 bar)	10 bar sensor (venTest accessory option)
Pr(Ambient)	Ambient (barometric) sensor

4.9 graph command

Command syntax: **graph:<filename>**

UI reference: Section 3.3.4.4

Example of usage: **plugin vPad-VT | "Download flow/volume chart" | graph:PM_test**

Description: This statement downloads the real-time graph as it is currently being shown on the venTest display and enables the user to save the graph as a PNG image file on the vPad tablet. This command is normally used to save ventilation flow-volume curves, although the venTest graph can also be configured via the Display Setup to show real-time signals from other pressure sensors (± 20 mbar ... 10 bar) and/or the gas concentration sensors (O₂; N₂O; CO₂; and anesthetic gas or "Agas").

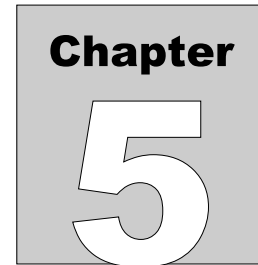
By default, the PNG file is saved to this folder path:

Datrend/vPad-Check/venTest/<control_number>/<filename>

where **<control_number>** is the ventilator ID entered as part of the equipment information at test startup (see section 3.3.1), and **<filename>** is as specified in the **graph** command, above.

If a **<control_number>** was not entered by the user at test startup, **"Ventilator"** is then substituted as a default. If a **<filename>** is not specified in the graph command, **"Test"** is used as a default. Before the image file is saved, users have the ability to change the folder path and append a date/time stamp to the **<filename>** (see section 3.3.4.4). When the image is saved, the filename and folder path are copied to the test record for later reference.

venTest Plugin for vPad-Check



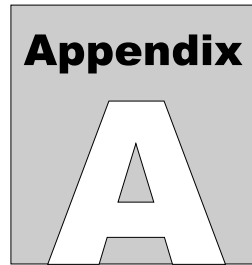
5 Upgrades and Maintenance

Software Applications (Apps) may be upgraded from time-to-time to provide enhanced features or improvements. Contact Datrend or your local dealer for information on these upgrades.

In some cases, the changes made to an App may be to correct operational issues that have come to the attention of DSI. These shall generally be referred to as maintenance upgrades. In other cases, the changes may be to improve performance or add features which would enhance the overall functionality of the App. The decision to apply a charge to any upgrade shall be solely at the discretion of DSI.

Datrend Systems Inc.
Unit 130 - 4020 Viking Way
Richmond, BC • CANADA • V6V 2L4
Tel 604.291.7747 or 800.667.6557 • Fax 604.294.2355
e-mail customerservice@datrend.com
www.datrend.com

venTest Plugin for vPad-Check



APPENDIX A. Example Checklist

Below is a listing of the "venTest Demo" which is included with the vPad-Check App. The demo illustrates how **plugin** statements may be used to interface the vPad tablet to the venTest analyzer via the vPad-VT plugin software ("venTest plugin"). To show how venTest measurements might be included in a preventive maintenance inspection, the demo includes some example steps related to visual inspection, routine maintenance, and electrical safety testing of the ventilator. This is an example for illustrative purposes only, and is not intended for use as an actual PM inspection for a ventilator.

```
// File:          venTest Demo.vpl
// Author:         ---
// Version:        1.0
// Created:        2017-May-31 1:40:08 PM
// Updated:        2017-Jun-07 4:32:46 PM
// Approved By:    ---
// Approved Date:  ---
// Description:    Demo checklist for venTest and vPad-VT plugin.
//
// Introduction
prompt  "Demo of venTest\nRemote Control via\nvPad-VT Plugin" | bold
prompt  "A venTest Bluetooth Adaptor and the vPad-VT software plugin for\n"+
        "vPad-Check are required to run this demo.\n\n"+
        "Confirm the venTest is powered up and the Bluetooth Adaptor "+
        "is connected to a USB port on the venTest.\n\n"+
        "Press 'Next Step' when ready to proceed with the demo..."
        | medium

// venTest initialization
plugin  vPad-VT | "Connect to venTest" | connect
plugin  vPad-VT | "Check venTest display setups" | search:Factory default
plugin  vPad-VT | "Enable auto-PASS mode" | autopass:on

// Visual inspection and servicing
header  "VISUAL INSPECTION\nand MAINTENANCE" | bold
check   "Inspect exterior of ventilator\nfor physical damage"
check   "Inspect casters and brakes"
check   "Inspect power cord and plug"
check   "Check circuit breaker or fuse"
power   on
check   "Apply power to ventilator and run self-test"
check   "Check indicators, displays and controls"
prompt  "AC supply will be switched off"
power   off
check   "Check operation on battery power"
```

venTest Plugin for vPad-Check

```
prompt    "Switch ventilator power OFF.\n\n"\ncheck    "Remove top cover and/or\nservice access panel(s)."\ncheck    "Inspect/replace air filter(s)"\ncheck    "Check oxygen fitting(s) and\nreplace as required"\ncheck    "Inspect/replace muffler(s)"\ncheck    "Inspect/replace inspiratory check valve"\nprompt    "Re-assemble ventilator and prepare to test"\n\n// Check ground resistance and chassis leakage\nheader    "ELECTRICAL SAFETY TEST" | red\nrun        vPad-NFPA | 4\n\n// Test ventilator using the venTest\nheader    "PERFORMANCE TESTS" | bold\npower     on\nprompt    "VENTILATOR TEST SETUP:\n\n"\n        "1. Connect ventilator to medical gas supply (air/O2/N2O).\n"\n        "2. Connect ventilator patient circuit to input of venTest "\n        "flow sensor.\n"\n        "3. Connect output of venTest flow sensor to test lung.\n"\n        "4. Set ventilator controls per manufacturer's instructions.\n"\n        "5. Press 'Next Step' when ready to test..." | medium\nplugin    vPad-VT | "Load test setup" | load:Factory default\n\nprompt    "Start the ventilator"\nprompt    "Testing ventilator. Please wait..." | normal | 10\nplugin    vPad-VT | "Check maximum flow" | read:PFi;10;100\nplugin    vPad-VT | "Check maximum pressure" | read:Peak;2;20\nplugin    vPad-VT | "Check minute volume" | read:MVi;1;10\nplugin    vPad-VT | "Check ventilation rate" | read:Rate;10;30\nprompt    "Tests complete.\nStop the ventilator"\nplugin    vPad-VT | "Disconnect from venTest" | disconnect\n\nprompt    "Ventilator will be powered off"\npower     off\n\nprompt    "End of ventilator inspection"\n// End of script
```