



BurnInTest

BurnInTest Linux

© 2023 PassMark Software

BurnInTest Introduction

by PassMark Software

PassMark's BurnInTest is a software tool that allows all the major sub-systems of a computer to be simultaneously tested for reliability and stability.

Anyone who has used computers for any period of time knows the importance of having a stable system. The cost of a hardware failure can be enormous. If you are lucky your data will only become inaccessible while faulty components are replaced.

If you're unlucky you can lose your files completely or have them permanently corrupted. Even if you have good backups, you can still lose days of work returning the PC to its pre-failure state. In many cases the lost data can never be replaced and businesses can be brought to their knees, with the loss of documentation, customer details and financial records.

BurnInTest Linux

© 2023 PassMark Software

All rights reserved. No parts of this work may be reproduced in any form or by any means - graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems - without the written permission of the publisher.

Products that are referred to in this document may be either trademarks and/or registered trademarks of the respective owners. The publisher and the author make no claim to these trademarks.

While every precaution has been taken in the preparation of this document, the publisher and the author assume no responsibility for errors or omissions, or for damages resulting from the use of information contained in this document or from the use of programs and source code that may accompany it. In no event shall the publisher and the author be liable for any loss of profit or any other commercial damage caused or alleged to have been caused directly or indirectly by this document.

Printed: June 2023

Table of Contents

Foreword	0
Part 1 Introduction and Overview	7
Part 2 What's New	8
Part 3 Quick Start Guide	8
Part 4 How to Purchase BurnInTest	8
Part 5 Entering A Licence Key	8
Part 6 Test Descriptions	9
1 Advanced Network Test.....	10
2 2D Graphics Test.....	12
3 3D Graphics Test.....	13
4 Optical Drive Test.....	13
5 CPU Maths Test.....	15
6 Hard Disk and Floppy Disk Test.....	16
7 Memory (RAM) test.....	18
8 Network Test.....	18
9 Parallel Port Test.....	20
10 Serial Port Test.....	21
11 USB Test.....	23
12 Sound Test.....	24
13 Plugin.....	25
14 PCIe slot and PSU test.....	26
15 Battery.....	27
16 Temperature.....	27
Part 7 Test Preferences	27
1 Advanced Network Preferences.....	28
2 2D Graphics Test.....	29
3 Disk Preferences.....	30
4 CPU Math Preferences.....	32
5 Memory Preferences.....	32
6 Network Preferences.....	32
7 CD-ROM/DVD Preferences.....	34
8 Parallel Port Preferences.....	35

9	Serial Port Preferences.....	35
10	USB Test Preferences.....	36
11	Sound Preferences.....	36
12	Pre-test Preferences.....	36
13	Post-Test Preferences.....	37
14	Error Handling Preferences.....	38
15	Logging Preferences.....	38
16	Plugin preferences.....	40
17	PCIe Slots.....	40
18	Management Console.....	40
19	Battery Preferences.....	41
20	Temperature Preferences.....	41
Part 8 Test Selection and Duty Cycles		42
Part 9 Interpreting the results		45
1	Test Window display.....	47
2	Detailed Error log history.....	49
Part 10 Menu Options		50
Part 11 Scripting		51
Part 12 Advanced options		60
1	Maintaining Multiple Test Configurations.....	60
2	Report Information (Machine ID and Notes).....	60
3	Command Line Parameters.....	61
4	Preparing a Customer Test Certificate.....	62
5	Management Console.....	63
Part 13 Command Line Version		64
Part 14 Other Useful Information		70
1	Contacting PassMark Software.....	70
2	System Requirements.....	70
3	Example ASCII Log file.....	71
4	Precautions for thorough and careful testing.....	72
5	Fault finding.....	74
6	Common Error Messages.....	75
7	USB Loopback Plugs.....	80
8	Acknowledgements.....	81

9 Making Loopback plugs.....	82
Serial port loopback plug	82
Parallel port loop back plug	83
 Index	 84

1 Introduction and Overview



BurnInTest is a software tool that allows all the major sub-systems of a computer to be simultaneously tested for reliability and stability. Anyone who has used computers for any period of time knows the importance of having a stable system. The cost of a hardware failure can be enormous.

If you are lucky your data will only become inaccessible while faulty components are replaced. If you're unlucky you can lose your files completely or have them permanently corrupted. Even if you have good backups, you can still lose days of work returning the PC to its pre-failure state.

In many cases the lost data can never be replaced and businesses can be brought to their knees, with the loss of documentation, customer details and financial records.

BurnInTest thoroughly exercises the hardware in a PC in a short period of time, in the same way as normal applications use a PC over a long period of time. This takes advantage of the fact that computer hardware tends to fail either very early in its life or after many thousands of hours of use.

Typically BurnInTest should be used before a PC is put into service, as a failure of a computer that isn't being used for a critical application and doesn't have any critical data is much less damaging, (especially if it is still under warranty).

BurnInTest will bring to the surface intermittent or hidden problems so that after a successfully run the computer can be used with a much higher level of confidence.

BurnInTest tests,

- The CPU via mathematical operations
 - Hard drives / Floppy drive
 - CD ROM and DVD
 - 2D graphics
 - 3D graphics
 - Memory & Network connection
 - Serial ports
 - USB ports
 - Parallel ports with loop back connectors (x86 only)
- All these tests can be done simultaneously as the application is multithreaded.

You can use BurnInTest to,

- Avoid delivering D.O.A. (Dead on Arrival) hardware to your customers.
- Dramatically reduce your burn in times with multithreaded simultaneous testing of components.
- Build your image as a supplier of quality systems.
- Avoid costly downtime, system rebuilds and lost data.
- Test the stability of a system after configuration changes or hardware upgrades (critical for over clocking).

For more information.

To get started quickly see the Quick start page

See Interpreting the results for information of BurnInTest controls and different views

See Test Descriptions for information about each test

Have a look at what each menu option does

Find out how to purchase the software

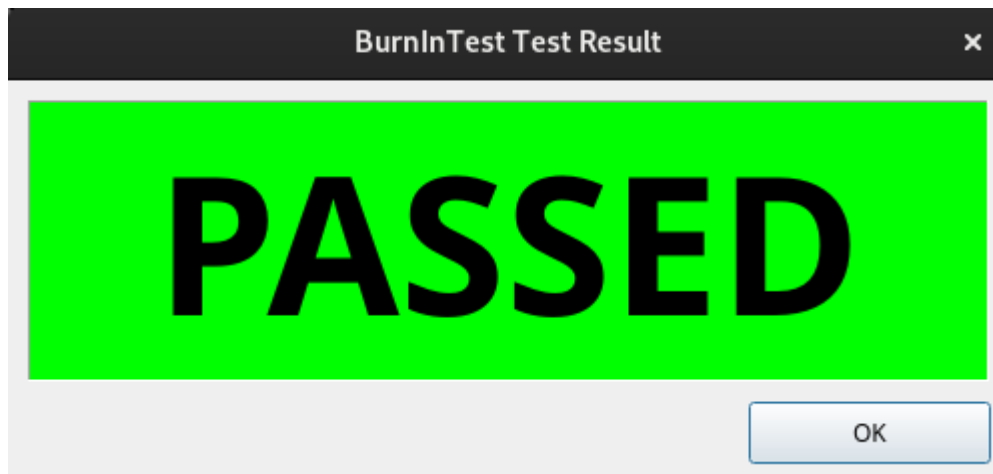
2 What's New

Please see our version history online at: https://www.passmark.com/products/burnintest/bitlinux_history.php

3 Quick Start Guide

Here's a summary of the steps to go through to start testing;

1. Use the Test Duty Cycles window to select the type of tests you wish to perform (see Test selection & Duty cycles).
2. Use the Configuration, Test Preferences window to set any parameters that you wish to use, e.g which hard drive to use (see Test preferences).
3. Put paper in the printer, a data CD in the CD-drive and a disk in the disk drive (if you selected these tests).
4. Click on the Start Tests button



Test finished window showing tests passed

4 How to Purchase BurnInTest

See the PassMark website for current pricing: <https://www.passmark.com/sales/>

Please note that the Linux and Windows versions of BurnInTest are separate products.

5 Entering A Licence Key

Copy your user name and key from the E-mail

- Select the entire key, including the -----START_OF_KEY----- and -----END_OF_KEY----- lines


```

-----START OF KEY-----
Test User
K82AKA9ZODKA91KA0DFLQ19DKSA91KD9FDAKDAC
ASD9KQ29CXKZB1AAAKA19839KFKALDDKA57ABBW
LA9289FXKMSDI3248FKS934KFSKSSOFS2KN2
-----END OF KEY-----

```

- Copy your user name to the clipboard. This can be done by using the Edit / Copy menu item in most E-Mail programs. Alternatively you can use the CTRL-C key combination on the keyboard.

Paste your user name and key into the software

Complete after ordering to license software

Username	-----START_OF_KEY-----
and Key	Test User
	K82AKA9ZODKA91KA0DFLQ19DKSA91KD9FDAKDAC
	ASD9KQ29CXKZB1AAAKA19839KFKALDDKA57ABBW
	LA9289FXKMSDI3248FKS934KFSKSSOFS2KN2
	-----END_OF_KEY-----

- Click on continue. If the user name and key was accepted, the program will start and the welcome screen will not be displayed again.

BurnInTest Linux Command Line

Installing a key for the command line version uses a command line argument and a file created called key.dat that contains your key;

```
./bit_cmd_line_x64 -k
```

Registers your key stored in the key.dat. When using this flag BurnInTest will expect your username/key, from (and including) the "-----START_OF_KEY-----" to (and including) the "-----END_OF_KEY-----" flags, to be stored in a file called key.dat in the same directory as BurnInTest. You will need to create a new file called key.dat and paste the key as is. It should be 6 separate lines, 1 each for the start and end flags, 1 for the username and 3 for the key, and then save the file. You will need to restart BurnInTest after registration and you can delete the key.dat file you have created.

Do not alter the savedkey.dat file, this is a file maintained by BurnInTest and altering this file will corrupt it.

6 Test Descriptions

Several tests are built into the software. Each of these tests contains one or more sub-tests designed to exercise different aspects of your computer system.

These tests can be run concurrently, with each test having it's own window and using it's own CPU thread or as individual tests. It is also possible to vary the load mix by increasing or decreasing the duty cycle for a particular test.

Each test keeps an error counter, a cycle counter and an operations counter. The significance of each of these counters is detailed with each test.

The following tests are currently available;

- 2D Graphics Test
- 3D Graphics Test
- CD and DVD Test
- CPU Maths Test
- Hard Disk and Floppy Disk Tests
- Memory (RAM) Test
- Network Test
- Parallel Port Test (x86 only)
- Serial Port Test
- USB Port Test
- Sound Test
- Plugin
- PCIe slot and PSU test (x86 only)
- Battery
- Temperature

6.1 Advanced Network Test

Overview

The Advanced Network test allows stability and load testing of up to 12 Network Interfaces (NICs) on a system running BurnInTest.

The Advanced Network test is a peer to peer TCP/IP test of NICs using various parameters and load settings. BurnInTest will detect the NIC's on the local system. BurnInTest will detect remote systems on the same network segment that are running EndPoint (EP) software (i.e. all EndPoints need to be visible to all the BurnInTest system NICs) and the NICs on each Endpoint. BurnInTest will need to run on one system and the remote machines will need to run the EP software. It is recommended that all local and remote NICs that are not to be tested be disabled, including Wireless cards.

Each network card should only have a single IP assigned to it, eg multiple IPv4 addresses for a single card are not supported.

IPv4 and IPv6 can be tested. Note: BurnInTest and the Endpoints must support IPv4.

One instance of the BurnInTest Advanced Network test can only be run on a Network. That is, you can not run 2 (or more) BurnInTest Advanced Network tests at the same time on the same network. Up to 12 NIC's on the BurnInTest system may be selected for testing.

The EP application (advnet_endpoint) may be run on up to 12 other systems on the network. Only 1 EP can be run per system. An EP can not be run on the BurnInTest system.

To set the Advanced network configuration options, select Configuration -> Test Preferences -> the Network tab, select "Advanced network test" and select the "Advanced network options" button. Please see Advanced Network Test for descriptions of the options available.

EP (End point) overview

The EP purpose is to detect requests from BurnInTest to start network testing and to send, receive and verify data from BurnInTest.

EP is a standalone application that is started on a remote machine and will be able to run whether there is an instance of BurnInTest running or not. It will need to be able to access all available NICs of the machine that it is running on.

Each network card should be on the same network segment as the BurnInTest system. Each network card should only have a single IP assigned to it, eg multiple IPv4 addresses for a single card are not supported.

The "advnet_endpoint" software file is included in the BurnInTest installation directory and should be copied to the remote PC's and run from these systems.

To use the advanced network test there must be at least 1 EP running on another system on the network. The higher the configured load (required %) and the larger the number of NIC's under test (on the BurnInTest system), the more Endpoints you should use.

It is recommended that the EP NIC's be the same or higher speed as the fastest BurnInTest system NIC. ie. it is probably not a good test if the EP NIC is 10Mb/s, and the BurnInTest system NIC is 100Mb/s or 1Gb/s. Only the 'first' EP NIC will be used for testing, hence it is suggested that the slowest NIC's are disconnected, leaving a single NIC (the fastest) connected on the EP system.

As a guide, it is suggested that with low load (required %), up to a maximum of 3 NICs can be tested per EP. With high load (required %), up to a maximum of 1 NIC can be tested per EP. For example:

1. If you were testing 2 x Gb NICs (BurnInTest side) with a configured load of 5%, then 1 x EP with a Gb NIC would be suitable. 2 x EPs could be used. 3 x or more EP's would provide no value.
2. If you were testing 2 x Gb NICs (BurnInTest side) with a configured load of 90%, then 2 x EPs with a Gb NIC would be suitable. 1 x EP would most likely provide lower throughput. 3 x or more EP's would provide no value.
3. If you were testing 6 x 100Mb NICs (BurnInTest side) with a configured load of 5%, then 2 EPs with a 100Mb NICs would be suitable. A single EP would most likely provide lower throughput. 3 to 6 would provide greater throughput. 7 x or more EP's would provide no value (and is not supported - a maximum of 6 EP's are supported).

It should be noted that the system may not be able to achieve the configured load (required %) due to a number of factors, including CPU load, and network configuration. It is suggested that other BurnInTest tests (e.g. CPU test) are not set such that they push the CPU beyond 'hard on 90%' when running the Advanced Network test.

To successfully run the test, BurnInTest uses broadcasts to find all the available EP applications on the network, so anything that may block the broadcasts, such as a router, needs to be taken into account. Firewalls could also interfere with the test and may need to be configured to allow BurnInTest and the EP applications to use the network.

How the test works

At the start of testing, BurnInTest broadcasts a request to the EPs on the local LAN segment. The broadcast will use UDP and be directed at the universal 255.255.255.255 broadcast address. Each EP will respond to this request. This provides BurnInTest with a list of all available EPs.

BurnInTest will then start a network test for each BurnInTest NIC selected for testing. Each of these BurnInTest system NICs will be assigned one Endpoint NIC to test with. For an IPv4 test NIC, the first EP on the same subnet will be used. For an IPv6 test NIC, a round robin of EPs will be used. This allocation will remain the same for the duration of the test.

A test will be started for each NIC and BurnInTest will send a control message to it's designated EP NIC to request it to start testing, on receiving a 'Ready to test' response from the EP, BurnInTest will setup test TCP sockets (WinSock) on the user specified port (+ the test number) and start sending test data to the EP. If full duplex is configured for that NIC, then the EP will also send test data back to it's allocated BurnInTest system NIC.

The test data is pseudo randomly generated at both the BurnInTest and EP ends (ie. the data is not just 'looped' by the EP). A checksum will be added to the data to allow BurnInTest and each EP to verify the data integrity (if the user has selected this option).

Both BurnInTest and the EP monitor Network load in order to estimate how much data should be sent from BurnInTest to an already connected EP (and from an EP to BurnInTest) in order to achieve the required % load.

Errors will be logged in the standard BurnInTest log files as specified in Configuration -> Test Preferences -> Logging.

Notes

In some cases, it is possible that a failing NIC (such as reported send or receive errors) can lead to subsequent false positive errors being reported for other NICs. When analyzing Advanced Network errors, this should be considered and the first error (within a period of 1 minute) taken to be the problem NIC.

Target load

BurnInTest will attempt to maintain the target speed load of the network card by inserting sleep periods into the send and receive threads, this is not the exact speed the test will run at but a rough average of the speed BurnInTest will attempt to maintain. The test has been designed with 1Gbps speed card in mind, slower cards such as 10Mbps will be harder to throttle sufficiently so their test speeds will jump around more.

There is no attempt to keep the total amount of data sent and data received the same so a discrepancy between the amount of data sent and received is not indication of an error. They run in separate threads and the endpoint attempts to throttle the data sent back to BurnInTest as well so there is no easy way to keep the data sent vs data received the same and they can vary significantly due to thread prioritization and speed throttling.

6.2 2D Graphics Test

Lines and Bitmaps

This suite contains a number of tests that exercise Qt's QPainter graphics functions and the video card. The numbers displayed at the top of the window for this test represent how many millions of graphical operations have been performed. Each test runs for half a second. After all 3 tests have been ran the cycle count is incremented. The duty cycle, the CPU speed and the video card speed, determine how many operations can be processed during the half-second period.

The number of shapes, lines and squares (bitmaps) drawn is displayed at the top of the test window followed by K (indicating thousands), M (millions) or B (billions).

The total of these three values is displayed in the main window.

Line Drawing

Lines are drawn into window. The color is changed every 500 lines to enable the lines to be seen in the test window.

Painting Bitmaps (Squares)

A square bitmap is painted into a window as quickly as possible.

Outline shapes

Two different shapes are drawn into a window:

- An ellipse
- A square with rounded corners

The color is changed every 64 shapes to enable the shapes to be seen in the test window.

Local Video Memory (GPU, RAM)

This test directly writes, reads and verifies data to the GPU memory using OpenCL.

The test will attempt to allocate 85% of the on-board memory of the GPU. If this amount can't be allocated (the system is already using some, eg for 4K displays) then it will decrease the amount requested until successful or the amount requested is less than 10%.

The test fills this memory with a series of test patterns. Each test pattern is written to video memory, read back from video memory and verified. Any errors are reported.

The test patterns used are similar to the memory test;

- Ones (11111111)
- Zeros (00000000)
- Binary 1 (10101010)
- Binary 2 (01010101)
- Sequence (1, 2, 3, ...)

1 test cycle is defined to be 1 complete pass (write, read and verify) of the test video memory.

Requires that OpenCL drivers are installed for the system (installing "clinfo" should install the required drivers and information tool to verify OpenCL can run).

6.3 3D Graphics Test

Tests the OpenGL's graphics system.

This test creates and animates a rotating cube with a different color on each side to test the OpenGL functionality of the video card and it's OpenGL capabilities.

For the 3D test a cycle is defined to be 200 frames of 3D animation. The 'operations' count represents the number of polygons displayed.

The test makes use of Qt's QOpenGLFunctions graphics class to run the test on systems that use OpenGL or OpenGL-ES (eg ARM embedded systems). If OpenGL or OpenGL-ES is not installed on your system, you will not be able to run this test (in fact you may not even be able to start BurnInTest). OpenGL can be downloaded from www.opengl.org and should come standard with all new versions of Linux distributions.

Warning: A number of manufacturers produce video cards, which have very poor software support for 3D graphics (OpenGL). This is especially the case for the cheaper, bottom of the range video cards. Poor drivers can result in strange visual artefacts appearing on the screen, poor 3D performance and system crashes.

Helpful command line tool

```
[your_command_line_prompt]# glxinfo
```

Displays your system's OpenGL's capabilities (if any).

6.4 Optical Drive Test

Tests the CD-ROM, DVD and BluRay drives.

This test can be used with many different types of CD / DVD media, including the following types.

- Music CD
- Data CD-ROM
- Specialized test CD (see below for details)
- Data DVD-ROM
- Specialized test DVD (see below for details)
- Specialized test BluRay (see below for details)

The type of disc being used must be suitable for the test mode set in the CD / DVD preferences window however. If multiple CD / DVD players are installed in your system, you can select all or some of them from the Preferences window (see section 0, Disk preferences). Up to twenty drives can be selected for simultaneous testing.

Test Modes

Music CD playback

For a music CD, the test verifies the operation of the CD ROM and the PC's sound system by repeatedly playing a music CD. The number of cycles corresponds to the total number of times the entire CD has been played. The number of 'ops' corresponds to the number of bytes read from the CD in order to generate the sound. The duty cycle affects the time spent waiting between tracks. Choose a band you like and turn the volume up loud.

Data disk read and verify

For optical data disks, the disk is scanned with each file being read in turn. A 32bit checksum is created for each file as it is read from the disk. The disk is then re-read and the checksums verified to complete the cycle. The best data disks's to use are those, which are full and contain a good variety of files. Some large ones and some small ones. If the CD has more than 10,000 files only the 1st 10,000 will be scanned.

PassMark Test CD / DVD

This test method is the most advanced CD-ROM test mode and provides a level of in depth testing not available with the other two methods. When possible this method should be selected in preference to the other two. It allows a complete read and verify of the data on the CD, and optionally random seek testing (see Additional seeking and Seek count below).

There is the possibility to create CDs and DVDs especially for use with BurnInTest. These specialized CDs and DVDs contain a set of files that have been specifically created to thoroughly check your drive. With these CDs and DVDs, BurnInTest is able to know the layout and exact content for each file on the CD/DVD. This allows BurnInTest to read back each file and test every bit for accuracy.

PassMark Software sells these specialized CDs and DVDs but you can also make your own file set using your own CD making tool. You will still need 3rd party CD or DVD burner software to actually create the disc however.

No CD in Drive test option

This test method only attempts to detect the presence of a CD/DVD drive. No attempt is made to read from the drive. This can be a useful option when there are no discs available for testing but a basic check is still required to ensure that the drive was detected by the system. This test option is not as thorough as the other options and, if possible, one of the other options should be used.

Additional seeking and Seek count

When Additional seeking is selected (for PassMark CD/DVD's and Burn CD-RW only), seeking to different positions on the CD/DVD or CD-RW and verifying the data at this position will occur Seek Count number of times for each cycle of a file read and verify.

Display

Depending on the test the following information is displayed in the test window.

Progress bar

Displays the progress graphically for the longer CD burn phases.

Phase

Data CD and specialized CDs :

- Starting
- Checksum creation
- Searching for files
- Opening file

- Reading Test CD
- Verifying checksums
- Seeking

Files scanned

This is the number of files that have been scanned during the current cycle.

Current file

This is the name of the file that is currently being scanned. If this is in the seek phase, this will display the file and block within the file that the drive has seeked to.

Bytes read

This is the total number of bytes read from the CD.

Throughput (Current and Average)

This is read speed from the CD in Megabytes per second and a conversion to standard CD Drive speed (eg. 4x). It should be noted that this measurement includes the seek and rotational latency time, so that the speed reported will always be lower than the maximum drive speed. This is especially the case for a CD with many small files. Also most drive manufacturers quote the maximum speed for their drives, the actual transfer rate is usually significantly lower than the quoted maximum. E.g. It is not unusual for a 32x speed drive to only reach a speed of 2MB/Sec, 13x.

Errors

Data CD and specialized CDs :

The number of errors detected. See Common Errors (Appendix C) for a description of the errors that may be encountered.

6.5 CPU Maths Test

The CPU test exercises and verifies a wide range of CPU instructions, at user specified loads. The user selectable sub-categories of the CPU test are:

- General purpose instructions
- Floating Point Unit (FPU) instructions
- CPU extension instructions (x86 extensions). The specific extension instructions may be selected: SSE, SSE2.
- CPU extension instructions (ARM extensions). The specific extension instructions may be selected: NEON.
- Prime number test

The numbers displayed in the test window for this test represent how many millions of CPU operations have been performed and verified. Each different math's test is run for half a second. After all tests have been run the cycle count is incremented. The duty cycle and the CPU speed determine how many operations can be processed during the half second period. By default, a CPU test is run on each CPU core.

General purpose instructions

This test exercises and verifies correct operation of CPU instructions from the following groups:

- Integer mathematics (e.g. add, subtract, multiply and divide)
- Data transfer instructions (e.g. pushing/popping data to/from the stack)
- Bitwise logical instructions (e.g. bitwise and/or/xor)
- Shift and rotate instructions (e.g. shift data left x number of bits)
- Logical instructions (e.g. equals)
- Control transfer instructions (e.g. jump on x greater than y)
- String instructions (eg. copy a text string)

Known and random data sets are used to exercise and verify correct operation. The random numbers are generated regularly to provide a larger data set, as well as ensure that the CPU caches overflow and that this mechanism is tested.

Floating Point Unit (FPU) instructions

This test exercises and verifies correct operation of CPU instructions from the following groups:

- Floating Point math (e.g. add, subtract, multiply and divide)
- Transcendentals (e.g. sin, cos)

Extension instructions (x86 extensions)

This test exercises extensions to the x86 CPU instruction set for a variety of applications, such as multimedia.

SSE: As above, but for SSE instructions. Instructions include moving SSE registers, adding, subtracting, and multiplying. Data set: groups of 4 x 32-bit floats (e.g. a,b,c,d) where a,b,c and d are 32-bit floats.

SSE2: As for the SSE test, but for SSE2 instructions. Data set: pairs of 64-bit floats (e.g. x,y) where x,y are 64-bit floats.

Extension instructions (ARM extensions)

NEON: Test the Advanced SIMD instructions. Instructions include moving NEON registers, adding, subtracting, and multiplying. Data set: groups of 4 x 32-bit floats (e.g. a,b,c,d) where a,b,c and d are 32-bit floats.

Prime number

This test exercises and verifies correct operation of CPU through the use of a prime number generation algorithm.

Helpful command line tool

```
[your_command_line_prompt]# cat /proc/cpuinfo
```

This command will show your CPU info, including speed, cache size and features.

6.6 Hard Disk and Floppy Disk Test

This suite contains a number of tests that exercise the mass storage units (Hard disk or otherwise) connected to the computer. By default, the start-up volume (with path represented by "/") is used for the hard disk and the floppy disk is represented by the mount directory in the "/etc/fstab" file, i.e. the "mnt_dir" field of the "mntent struct" as returned by a `getmntent()` system call. You can opt to test more disks/volumes by changing the Preferences Dialog (See section, 0, Disk preferences). Up to 120 drives can be selected for simultaneous testing. From this twenty it is possible to select any combination of available floppy disk and hard disks.

Partitioned drives they must be mounted before BurnInTest can test them, they also need to be of partition type ext2, ext3, ext4, reiserfs, xfs, zfs, vfat or ntfs. Due to limited NTFS support in some Linux distributions you may need to use a non-default type (eg ntfs-3g) to mount NTFS drives with read + write permissions before you can test them.

The user can select one of the following patterns or choose to automatically cycle between the patterns. This selection is done in the preferences dialog (see section, 0, Disk preferences). Possible test mode patterns are,

- Default (Cyclic)
- Sequential data pattern (0,1,2...255)
- Random data with random seeking
- High Low freq data overwrite (10101 then 00001)
- Binary data pattern 1 (10101010)
- Binary data pattern 2 (01010101)
- Zeros data pattern (00000000)
- Ones data pattern (11111111)
- Random data pattern

During each test cycle, a file is created and verified on the disk. The size of the file is equal to a certain percentage of the disks capacity and all files are created in the root directory of the drives selected. The default file size is 1.0% of the disk size. For small capacity drives, like floppy disks, there is a minimum file size of 32KB. It is possible to change the default file size percentage from the preferences window. The default setting is to cycle between the patterns. In this case a new pattern will be selected each time the disk is full

Each test file is filled with a coded number sequence (a pattern) that is used to verify the correct operation of the disk when the file is verified (read). Files are created in the free space on the disk until the disk is 94% full. When the disk has reached this limit, all the test files are deleted and the test starts again. A limit of 94% is used to ensure that some space is kept for use by other applications (including the swap files).

A number of test modes perform additional testing to that described above. These are described below:

Random data with Random seeking

Generates 7 blocks of random data for each test disk and sequentially writes a file using the first block of random data, followed by the second, third, fourth, fifth, sixth and seventh and then the first etc, until the file size specified is reached. This file is then verified with sequential reads of data blocks from the file just written. This is followed by seeking to a block within the files created, selected at random. At random either a read and verify or a write will occur at this position on the disk. This random seek and random read/verify or write will be repeated the number of times specified by the user in the disk preferences "seek count" field. New files are written and verified with this process until the disk full level is reached. At this point, all the test files written to disk will be deleted and the cycle will be repeated.

High Low frequency data overwrite

Works by first writing a file with a high frequency pattern (10101010101010010101...), then overwriting this with a low frequency pattern (00001000010000100001...) and then verifying that the low frequency pattern has been fully and correctly written. Note: This test pattern will write more data than it reads/verifies, this is the reason the MB Written and MB verified shown on the disk test window may not be equal.

If an error is detected in the coded pattern then the error count is incremented. The numbers of bytes written and read from the disk are displayed in the test window. The addition of these two values is displayed in the main window.

Graphical progress bars indicate if the test is currently writing or verifying (reading) information, the percentage complete for the particular file and the space remaining on the disk.

The speed of the hard disk and the duty cycle determine how quickly test files are created.

There are a few issues to aware of when interpreting the results of the disk test. These are covered in the precautions section (see section 0, Precautions for thorough and careful testing).

Note (1) To test Firewire ports using BurnInTest it is recommended that an external hard disk via the Firewire port is used in conjunction with the BurnInTest disk test. (2) Similarly, to test external memory card readers/writers or other removal drives, it is recommended that the BurnInTest disk test be used.

RAW (un-partitioned) Disk Testing

When a Raw (un-partitioned) disk is detected it will be displayed in the drive columns as "Raw Disk [device name]". This will allow the device to be selected for testing and write directly to the disk, bypassing the file system. Be aware that this test is designed to be run on un-partitioned devices and running the test on a partitioned device can result in corruption of the partition and data stored on the device.

Helpful command line tool

```
[your_command_line_prompt]# cat /etc/mtab
```

This command will show all the mounted disks on your system.

```
[your_command_line_prompt]# df
```

Shows the disk usage including used space and available disk space.

6.7 Memory (RAM) test

Tests the Memory in the computer.

The Memory test, tests the reliability of the RAM installed in the computer. Some of the available RAM is being used by the operating system and any other applications running. Any memory that is not already in active use will be grabbed by the Memory test.

The memory test works by writing a pattern numbers in the RAM, then verifying the numbers read from the RAM match this sequence. The pattern used will change automatically from one cycle to the next. Possible patterns are:

Sequence (0,1,2...255)

Binary 1 (10101010)

Binary 2 (01010101)

Zeros (00000000)

Ones (11111111)

There are 3 sequences that the test goes through:

1. Memory allocation. The test will dynamically grab memory depending on how much is currently available. The amount that has been grabbed and is under test is displayed in the Test Ram Field.

2. Writing the number sequence.

3. Verifying the number sequence.

The total amount of free RAM is displayed in the Memory Test Window. This number should always be bigger than the Test Ram number. Some memory is always left available to avoid disk thrashing caused by the operating system swapping to disk.

For the Memory test a cycle is defined to be the number of times the above 3-step sequence is completed. The 'operations' count represents the number of bytes read or written.

It should be noted that not all RAM faults will be detected by this test. This is especially the case if the operating system and it's cache is using a large proportion of the available RAM. RAM faults may show up as system crashes or disk errors however.

Compatibility issues

You need to have administrator privileges to run this test.

6.8 Network Test

The Network test tests your networking hardware and software. This is done by sending special ICMP_ECHO message using the Internet Control Message Protocol (ICMP). The ICMP_ECHO message is more commonly known as a 'ping'. This message is echoed but to your computer by a remote host. This allows the reliability of the network connection to be determined. Up to 6 remote hosts can be selected by setting the Destination Network address in the Test Preferences window (Network preferences).

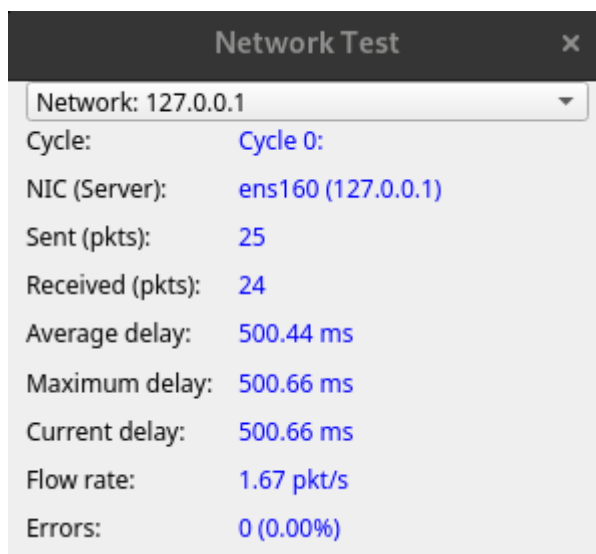
These values are used for the Network test. Each one must be a URL or an IP address (or left blank).

- A URL is the name of a network host, eg. www.hostname.com
- An IP address is a sequence of 4 numbers that correspond to a network host. eg. 169.192.0.1

The host selected must be accessible from the computer and capable of responding to the 'ping' command. PassMark recommends the selection of a local host to minimize data link problems, which are fairly common on the Internet. The IP address 127.0.0.1 can also be used for local loop back testing. It should be noted however that the loop back is done in the TCP/IP software and not on the network card, this address cannot be used when the "Test All NICs" option is checked.

The packet sent to the remote host contains a data payload and a checksum. Every time a packet is echoed from the remote host the checksum is verified and the data payload is compared byte by byte with the data that was sent. Any differences in the payload or an incorrect checksum will result in an error. The data payload is 64 bytes in length.

The amount of time BurnInTest will spend waiting for a packet can also be set in the Test Preferences window (see section 0, Network preferences).



Network test window

The meaning of the information displayed in the network test window is given below.

NIC (Server)

The NIC being tested and the address it is sending to and receiving from.

Packets Sent

The number of packets sent to the remote host.

Packets Received

The number of packets received back from the remote host. This should remain at the same level as the Packets Sent counter. If after 2 second a sent packet is not echoed, this causes a timeout error and Packets Sent will be greater than Packets received.

Average Delay

The average round trip time in milliseconds for a packet.

Max Delay

The maximum round trip time in milliseconds for a packet. The maximum values often happen at the very start of the test session. This is because the operating system is still loading and caching the required networking software.

Current delay

The round trip time in milliseconds for the last packet sent.

Bytes sent

The total number of bytes transmitted to the network.

Flow rate

The number of packets being sent per second. The duty cycle set for the Network Test determines how many packets are being sent per second.

Errors

The number of errors that have been detected and the percentage of packets that had an error are displayed. The definition of what an “error” is can be changed from the Test Preferences window (see section 0, Network preferences). Depending on the settings in the preferences window, the error count can be incremented for every detected error or alternatively it can be set to only increment when the specified error ratio is exceeded. When the specified error ratio is exceeded, the ratio is reset to zero to avoid triggering a continuous stream of errors. Thus the ratio value displayed is the ratio of bad errors to good packets since the last time the threshold was crossed.

For the purposes of detecting a crossing of the threshold and signaling an error, the ratio is ignored until a sufficient number of packets have been sent to make the ratio valid. For example, if the ratio is set to 2%, at least 50 packets must be sent before the ratio is deemed to be valid. If the ratio is set to 0.1%, at least 1000 packets must be sent before the ratio is deemed to be valid.

Compatibility issues

You need to have administrator privileges to run this test.

6.9 Parallel Port Test

Tests the parallel communications port connected to the PC. The parallel port to be tested can be selected from the Test Preferences window (see section 0, Parallel port preferences).

A parallel port loop back plug is required to run this test. These can be purchased from the PassMark web site (www.passmark.com) or you can make them yourself (see Appendix D).

Each test cycle corresponds to 500,000 bytes of data transmission. The number of ‘ops’ corresponds to the number of bytes sent and received. The duty cycle affects the time spent waiting between cycles.

The parallel port selected must not already be in use by the operating system (for example by the printer or other external device), for the test to be carried out. The default on-board Parallel port settings are that Port0 is named /dev/lp0 and corresponds to the physical IO memory address 0378.



Parallel loopback plug

Parallel Port

This is the port name for the parallel port being tested. The port can be selected from the Test Preferences window.

Bytes Sent

This is the number of bytes that have been sent to the parallel port.

Bytes Received

This is the number of bytes that have received from the parallel port.

Errors

This is the number of errors detected (see Appendix C, p.48).

Throughput

This is the measured throughput for the port.

Compatibility issues

You need to have administrator privileges to run this test.

6.10 Serial Port Test

Tests the serial communications ports connected to the PC. Up to 64 serial ports may be tested simultaneously. The serial ports and test speed can be selected from the Test Preferences window (see Serial port preferences).

A serial port loop back plug per port is required to run this test. These can be purchased from the PassMark web site (www.passmark.com) or you can make them yourself (see Making Loopback plugs).



Serial loopback plug

Each test cycle corresponds to about 10 seconds of data transmission followed by a signal pin test phase. The signal pin test phase checks that the following pins on the serial port are functioning correctly.

RTS – Request to Send

CTS - Clear to Send

DTR – Data terminal ready

DSR – Data set ready

The number of 'ops' corresponds to the number of bytes sent and received. The duty cycle affects the time spent waiting between cycles.

The serial port selected must not already be in use by the operating system (for example by the mouse or an active modem), for the test to be carried out.

The speed that the serial port operates at is independent from the modem speeds. Even if you have a 56Kbit/s modem your serial port may operate at a higher speed. The maximum serial port speed depends on the type of chip installed on your motherboard. Most PCs will only do up to 115Kbit/s, so don't be alarmed if the test fails at 128Kbit/s or above.

If the "detect only" option was selected in the preferences window then the loopback test will not be performed. The presence of the serial port in the system will still be checked for however.

The following information is displayed for each port being tested.

Serial Port

This is the name for the serial port being tested. The port can be selected from the Test Preferences window. Any port between /dev/ttyS0 and /dev/ttyS63 is supported.

Speed

This is speed that the serial port is configured for. The speed can be selected from the Test Preferences window.

Bytes Sent

This is the number of bytes that have been sent to the serial port.

Bytes Received

This is the number of bytes that have received from the serial port.

Errors

This is the number of errors detected (see Appendix C, p.48).

Throughput

This is the real measured throughput for the port. This will generally be less than the Speed (see above) as there is some overhead in the code and in the data transmission itself (e.g. Stop bits).

Compatibility issues

You need to have administrator privileges to run this test.

6.11 USB Test

Tests the USB (Universal Serial Bus) communications ports connected to the PC. The number of USB devices to test can be set in the USB test preferences window.

A USB port loop back plug is required to run this test. The USB 3.0 loopback plugs can test USB3 ports at Super-speed (5Gb/s) and USB2.0 ports up to High-speed (480Mb/s). The USB 2.0 loopback plug provides a way to test USB 1.x and 2.0 (USB 2.0 supports rates up to 480Mb/sec). These can be purchased from the PassMark web site (www.passmark.com).

Using a USB3 or USB2 loopback plug and BurnInTest, it is possible to,

- Quickly check if a USB port is powered (the red LED)
- Check that data can be sent and received from the port
- Check USB errors rates and transmission speed.
- Check that the system remains stable under long periods of load
- Check USB cabling
- Concurrently check multiple USB ports at the same time.
- Check if your PC ports are super speed 5Gbits/s (USB3 only), high speed 480Mbits/sec (USB 2.0), or full speed 12Mbits/sec (USB 1.x) via a LED

Using these plugs on a system that is functioning correctly, you can expect error free loopback transfer speeds of around:

- USB 2.0 loopback plug: 5 – 7 Mbits/sec per port.
- USB 3.0 loopback plug: 80 - 100 Mbits/sec per port.

To reach this level you need to have the USB test duty cycle set to 100%. Up to 20 of these USB test plugs can be simultaneously connected to a PC (providing that free USB ports are available on the PC or on a down stream hub). Note: If required, both USB2 and USB3 loopback plugs can be used to test different USB ports simultaneously.

Each test cycle corresponds to;

- USB 2.0 loopback plug: 8000 data blocks (4000KB)
- USB 3.0 loopback plug: 1,000,000 data blocks (1 MB)

The number of Operations ('ops') corresponds to the number of bytes sent and received. The duty cycle affects the time spent waiting between cycles.

Each plug also has its own serial number stored in EPROM, on the plug, so it is possible to identify each plug when multiple plugs are connected. USB2 loopback plugs are firmware controlled and upgradeable.

The USB test sends data to the USB loopback plug in 0.5KB blocks for USB2 loopback plugs and 1KB blocks for USB3 loopback plugs. The USB loopback plug receives this data, copies it to a new buffer and transmits it back to the PC. The PC compares the data in the block for an exact match, and then builds a new packet of random data bytes before sending it back to the PC. Any differences between the data send and received is flagged as an error.

As the low level USB protocol has some ability to detect errors and re-transmit bad data it is important to pay attention to the transfer speed. A low transfer speed may be an indicator of an intermittent or partial hardware fault. With the USB2 loopback plug an error LED will be lit when more than a defined number of recoverable bus errors occur (currently set to 4). (Note: The USB2Test software will report the number of recoverable bus errors per block). A lack of high quality cabling within or external to the PC is a typical cause of retransmission due to recoverable bus errors.

The USB2 loopback plug Error LED is reset at the start of each test if results are configured to be cleared:

- at the start of each test (Preferences, Logs, Log Clearing, Automatically clear logs at the start of each run),
- on selecting "Edit, Clear All Results" or
- on a power reset (e.g. PC sleep mode).

USB test window

The following information is displayed in the USB test window:

USB Identity

The device number, unique device serial number and the bus and port number the plug is connected to are displayed. The device number is determined by the order in which plugs are connected and disconnected and is not linked to a physical USB socket on the PC. (Note that this is different from Serial and Parallel ports). The device serial number is collected during the USB device enumeration process and comes from the memory (EEPROM) in the plug itself.

Bytes Sent

The number of Kilobytes sent to the USB device.

Bytes Rec

The number of Kilobytes received from the USB device.

Throughput

The number of Megabits of data received and transmitted in the last second.

Ave. Throughput

The average number of Megabits of data received and transmitted per second since the start of the test.

Max Read

The maximum read speed measured during the benchmark test.

Max Write

The maximum write speed measured during the benchmark test.

Errors

The number of errors that have occurred (i.e. Transmitted data does not match received data).

6.12 Sound Test

Sound loopback testing measures the distortion between audio input and audio output. It involves connecting the audio out and line in plugs of your computers sound card together with an audio loopback cable (available from <https://www.passmark.com/products/hardware.php>).

The test uses a single channel (mono), 8 bit sound format with telephone quality sample rate (11025Hz). The test generates a sin wave sample with a frequency of 300Hz and a duration of approximately 20 seconds. This sample is output, looped back over an audio loop back cable and the input recorded. Once the sample is complete, the test then analyzes the waveform for distortion. It is possible to set the maximum level of distortion before an error is logged in the Sound preferences window. An operation is defined to be an 8-bit sample of a waveform that is looped and is within the user distortion setting. A cycle is defined to be playing a waveform sample of approximately 20 seconds, recording this sample and analyzing it for distortion.

BurnInTest measures the level of distortion between the output signal and the input signal on a wave-by-wave basis. If the distortion measured exceeds the value set in the Sound Preferences window and an error is logged and the output and input waves are displayed in the Sound Test window.

The volume control for the system is a very important setting for the loopback test. If the volume setting is too low, the input signals will be correspondingly vague and therefore distortion will be high. If the volume setting is too high, the input signal may become “capped” as the sound waves become truncated. This will also lead to high distortion and errors being logged. You may need to play with this control while the test is in progress in order to find the “sweet spot”

Note 1: If a system is under high load, the audio device driver may not be able to output a continuous Sine waveform. If this occurs, you may see the "Corrupt Audio input" warnings. To determine if this is load related, reduce system load and re-test.

Note 2: If your system produces audio on certain events, such as clicking a mouse button, then these sounds should be turned off before using the sound Loopback test.

6.13 Plugin

If you have specialized hardware that BurnInTest does not test, you can write your own test and integrate it with BurnInTest.

Up to 5 Plugins can be specified

The plug-in test will appear as another test, much like all of the tests. It may be switched on or off, and duty cycle set. The Plug-in application can define the following, which may be displayed, logged and included in the BurnInTest results:

- The test window title text.
- The number of test cycles completed.
- The number of plug-in test errors in the current test run. The plug-in may flag BurnInTest when a new error occurs, and pass a plug-in defined error message (this will be displayed in the BurnInTest windows and may be logged in the log file).
- The plug-in may flag BurnInTest when it wants to display new status text (defined by the plug-in), such as "Waiting for event".
- The Plug-in may define up to 3 "number of operations counts". Typically this would be for something like, the number of writes, reads and verifies. All three operation counters may be given plug-in defined labels, which will be displayed in the BurnInTest test window. Any operation counter may be flagged as not in use, hence you could use a single operations counter, such as just "Write". BurnInTest will display each label and number of operations counters separately in the test window. They will be summed for the main window operation count. These values are 64 bit integers.
- 6 Plug-in defined string values and labels are provided for the display of other interesting data to the user, e.g. the plug-in could provide throughput in Mb/s to BurnInTest. BurnInTest displays this information in the test window. Both of these fields may be flagged as not in use.

Software may be written to interface an external plug-in test module with BurnInTest. An interface is defined to pass test parameters between an external test application and BurnInTest. BurnInTest will attempt to start this application when the Plug-in test is run and this application is specified as the Plug-in file. BurnInTest will pass a flag that indicates that the test run is still underway (e.g. has/ has not been stopped by the user) and BurnInTest will pass in the Duty cycle. Before starting the test, the plug-in application must initialize the test parameters (such as display labels) and flag that they have been set. Once the test is started, the test result parameters and most test window labels may be updated based on the test results. BurnInTest will pick up any changes. Specific cases where the Plugin must flag to BurnInTest that new parameters are available are for a new labels (as a group flag), error, a new status message and the user defined strings 1 & 2. You may need to check that BIT has read the old values before writing new values. See the Interface definition, PLUGININTERFACE, and the sample source code for details.

The PLUGININTERFACE struct (in PluginTestInterface.h) is used for communication between the plug-in and BurnInTest.

For each of the user defined fields (OUT_bUserDef1 to OUT_bUserDef6) the field is enabled by setting OUT_bUserDefX to true, setting OUT_szUserDefX to a string name/description and the OUT_szUserDefValX to the value to be displayed.

The OUT_iStatus field is used for pre-test plug-ins to signal that they have finished by setting the value to PRE_TEST_PLUGIN_COMPLETED.

The OUT_szStatus is used to display the current status of the plugin in BurnInTest.

A test failure is defined to be one or more CRITICAL, SERIOUS or WARNING errors. INFORMATION level events are logged for information only and are not considered to constitute a test failure.

6.14 PCIe slot and PSU test

When a PCIe test is started, BurnInTest will send a packet of data (using an incrementing byte pattern) to the PCIe test card, which will loop this data back in firmware and send the same data back to the PC. One PCIe test card is required for each PCIe slot to be tested. The PassMark PCIe test card can be purchased from the PassMark web site and you will need to download, build and load the PCIe test card device driver for Linux before BurnInTest can access the card.

Temperature and PCIe voltage statistics are retrieved from the PCIe card once per test cycle. The Voltage values are compared with the range values of +- 10% and if the voltage value is out of this range then an error is raised. The temperature values are compared with the range values of 0C to 90C and if the temperature value is out of this range then an error is raised.

PSU voltages can be monitored during the test by connecting a PSU connector to the PCIe loopback card. The Voltage values are compared with the range values of +- 10% and if the voltage value is out of this range then an error is raised.

PCIe test window

The following information is displayed in the PCIe test window.

PCIe Identity

The PCIe card number, the negotiated speed (x1 or x4) and the firmware version is displayed in the drop down menu at the top of the test window.

The test details for each can be displayed by selecting them from the drop down list.

Bytes Sent/ Received

The number of Kilobytes sent to and received from the PCIe card.

Average Throughput

The average number of Megabytes of data received and transmitted per second since the start of the test.

Temperature [minimum/average/maximum]

The minimum, average and maximum temperature reported by the PCIe test card (in C) since the start of the test.

12V [minimum/average/maximum]

The minimum, average and maximum PCIe 12V reported by the PCIe test card since the start of the test.

3.3V [minimum/average/maximum]

The minimum, average and maximum PCIe 3.3V reported by the PCIe test card since the start of the test.

12V PSU [minimum/average/maximum]

The minimum, average and maximum PSU 12V reported by the PCIe test card since the start of the test. Note the PSU connector must be connected to the PCIe test card to allow the monitoring of PSU voltages.

5V PSU [minimum/average/maximum]

The minimum, average and maximum PSU 5V reported by the PCIe test card since the start of the test. Note the PSU connector must be connected to the PCIe test card to allow the monitoring of PSU voltages.

3.3V aux [minimum/average/maximum]

The minimum, average and maximum PCIe 3.3V aux reported by the PCIe test card since the start of the test.

6.15 Battery

This test checks the battery's level (mWh) and compares this to a threshold value. The threshold value can be specified in absolute, or percentage terms relative to the batteries designed capacity. Threshold exceeded errors will be raised if the battery level drops below the threshold specified.

These checks can be carried out before the main tests as a pre-test (checks once), and/or during the main tests (checks once per minute).

The Battery information will be logged in an interim result summary (once per minute) and in the final result summary.

See the Battery Preferences window for test settings.

6.16 Temperature

If the temperature test is selected then an error will be logged if one of the selected temperature sensors exceeds the threshold value that has been selected.

Requires lm-sensors be installed to retrieve CPU temperatures.

Requires hddtemp or smartctl to be installed to retrieve HDD temps.

See the Temperature Preferences window for test settings.

7 Test Preferences

- Disk Preferences
- CPU Math Preferences
- Memory Preferences
- Network Preferences
- CD-ROM/DVD Preferences
- Parallel Port Preferences
- Serial Port Preferences
- USB Test Preferences
- Sound Preferences
- Pre-test Preferences
- Error Handling Preferences
- Logging Preferences
- Plugin Preferences
- PCIe Preferences
- Management Console Preferences
- Battery Preferences
- Temperature Preferences

7.1 Advanced Network Preferences

Advanced Network Test Options

Select network card(s) to test

Test mode: Select network card IP address

Network card IP address: NIC0 IPv4, ens160 : VMware VMXNET3 Ethernet Controller 192.168.2.236

☐ Test on

Test settings

☒ Half duplex ☐ Full duplex

Data port number: 10001

Block size: 16384 Bytes

Target NIC load: 5 %

Low NIC threshold: 0.0 Mbits/s (0 = unused)

☒ Validate data

Display test endpoints

The advanced network test will test IPv4/IPv6 network cards. It is a TCP/IP test to systems on the local network that are running PassMark endpoint software.

Select "Display Endpoints" to see the Endpoints current running and visible to BurnInTest.

Display endpoints

Help OK Cancel

Select network card(s) to test

Test mode

This drop down selection sets the test mode to be one of the following options:

- Select Network card IP addresses: Individually specify each Network card to be tested. The test settings can be specified per Network card.
- Test available IPv4 network ports: At the start of the test, BurnInTest will determine all of the available IPv4 network ports and set up a test for each Network port. The test settings are the same for all Network cards that are tested.
- Test available IPv6 network ports: As above, but test available IPv6 enabled network ports only.

Note: selecting a Test mode to automatically detect the Network cards has the advantage that when testing across many systems with different network cards or IP addresses, the same BurnInTest configuration can be used. The downside is that if BurnInTest is unable to detect a network card (for example, the network card device driver failed to load), then there won't be an attempt to test the faulty card, and hence no error will be raised (unless there are no cards detected).

Network card IP address

This list will show the IP of up to 20 network cards that are installed on the system and plugged in. The values in the fields below are saved for each network card and can be different for each card.

IPv4 and IPv6 NICs will be displayed in the drop down. Note: BurnInTest and the Endpoints must support IPv4.

Test On

Check this box to enable testing for the selected IP of a network card.

Test settings**Half duplex / Full duplex**

Full duplex – will send and receive data during the test.

Half duplex – will only send data during the test.

Data port Number

This is the starting port number that is used when sending data during the test, as the testing progresses more than one port may be in use by each network card. Unless it conflicts with other services in use this should be left at the default value.

Block Size (bytes)

This value is buffer size used when breaking the file into packets and sending it across the network.

Target NIC load (%)

This is the required load placed on the network card during testing, the test will be throttled to maintain the necessary amount of data being sent through the network card. For example, a load of 15% for a 1000 megabit (1 gigabit) network card will be a transfer rate around 17 MB/s. Note: If this setting is high, the system may not be able to generate sufficient traffic to meet this load requirement.

Low NIC threshold

A low throughput threshold value can be specified to raise a throughput error when the throughput falls below the specified level.

Validate data

If this is checked then the data sent during the test will be validated, this may slow down the testing due to the CPU overhead of validation.

Display test endpoints**Display endpoints**

This will scan and display the IPv4 and IPv6 address from each endpoint that is currently available for testing.

7.2 2D Graphics Test

Local video memory (GPU RAM)

Tests the video RAM by writing and reading data to and from the RAM on the video card.

Monitor Test (Lines and Bitmaps)

Tests standard two dimensional graphical functions by painting 2D shapes to the test window.

7.3 Disk Preferences

BurnInTest Preferences

CPU-Math Memory 2D **Disk** Optical Disk Parallel Port Serial Port Network USB PCIe Temp / Battery

Test disk	Test Mode	File size (%)	Duty cycle override	Seek count	Block size
Hard Disk (/home) [/dev/nvme0n1p3]	Not ...				
Hard Disk (/boot) [/dev/nvme0n1p2]	Not ...				

Edit details for drive: Hard Disk (/home) [/dev/nvme0n1p3]

☐ Test this disk

Test mode: Default (Cyclic)

File size: 1 (% of disk space)

Duty cycle override: 0 Block size: 4

Seek count: 100 Max test cycles: 1

Automatically select all hard drives

☐ Select all mounted hard drives (selects all formatted _mounted drives at run t

☐ Select all RAW hard drives

Test mode: Default (Cyclic)

File size: 1.0 (% of disk space)

Max test cycles: 1 Block size: 4096

Exclude these drives (seperate with ',')

Help OK Cancel

Partitioned drives must be mounted before BurnInTest can test them, they also need to be of partition type ext2, ext3, reiserfs, xfs, vfat or ntfs. Due to limited NTFS support in some Linux distributions you may need to use a non-default type (eg ntfs-3g) to mount NTFS drives with read + write permissions before you can test them.

Adding a disk

To add a disk for testing, select it in the list of available disks and then click the “Test this drive” checkbox. To test a shared network drive, the drives needs to have a drive letter mapped to it. Only those drives detected by the operating system as displayed in the drop down list.

The following settings can be configured differently for each drive: Test mode, File size. Duty cycle override, Block size, Seek count and Max test cycles.

Removing a disk

To remove a disk from the list, select it in the list of available disks and uncheck the "Test this drive" checkbox. The "Test Mode" field in the list will change to "Not Testing".

Up to twenty drives can be selected for simultaneous testing. Note: The simultaneous testing of two partitions on the same physical hard drive will result in a lot of seeking between partitions and slow down the test significantly.

Editing a disk

To edit the values for a disk, select it in the list of disks and if the "Test this drive" checkbox has been checked then the other options will become available. Any changes made to the current values will be automatically reflected in the list.

Edit details for drive

Test mode

The test mode patterns that can be selected are explained in the section Hard Disk and Floppy Disk Test.

File size

The user can select the test file size and test pattern that is used with the disk test. The size of the file is equal to a certain percentage of the disks capacity. The default file size is 1.0% of the disk size. So if a disk had a total capacity of 40GB then the default size of each file would be 400MB. Setting a smaller percentage results in more files being created on the disk and the read / verify cycle occurring more quickly.

Duty cycle override

To use the general disk drive Duty cycle for each disk just set the Duty Cycle override value to blank (no value), otherwise set the required value per disk.

Seek count

For test modes that perform seeking to different positions on the disk drive (eg. Random data with random seeking), the seek count specifies the number of seeks for a particular iteration (eg. After Random data with random seeking has created 100 test files, the Seek count specifies the number of times a seek should occur between blocks within these 100 files).

Automatically select all hard drives

If you would like to test all the detected disks over multiple systems (using the same configuration file) you can use the "Select all mounted hard drives" option to test all the detected mounted hard drives and "Select all RAW disks" to test all the detected RAW drives. The same test mode and file pattern will be used for all the disk tests in this case.

Exclude these drives

Any mount point (eg /) or device name for RAW devices (/dev/sda) will be excluded from the test when the "Automatically select all" option is used. Each entry in this list should be separated by a ',' character.

RAW (un-partitioned) Disk Testing

When a Raw (un-partitioned) disk is detected it will be displayed in the drive columns as "Raw Disk [device name]". This will allow the device to be selected for testing and write directly to the disk, bypassing the file system. Be aware that this test is designed to be run on un-partitioned devices and running the test on a partitioned device can result in corruption of the partition and data stored on the device.

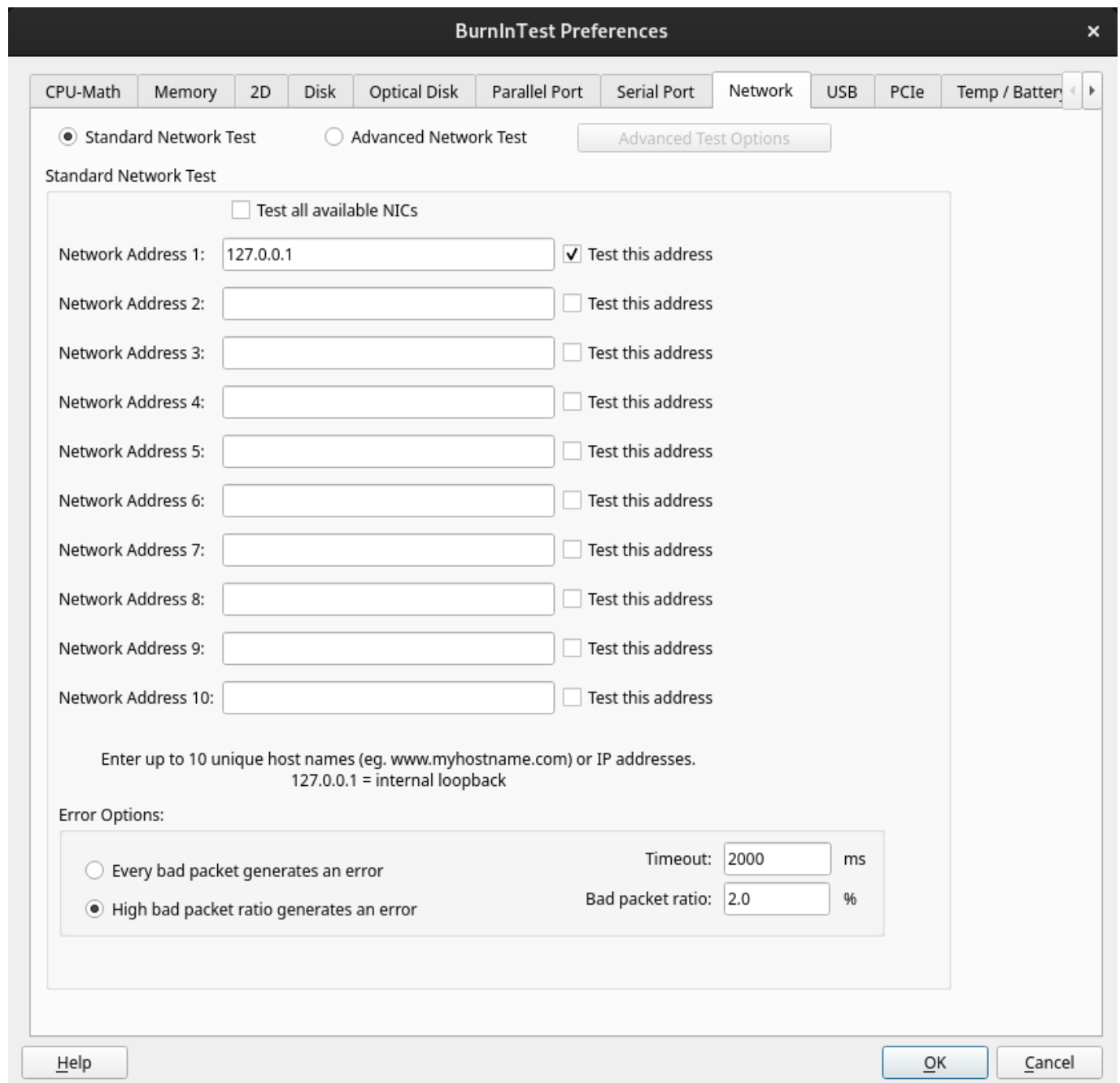
7.4 CPU Math Preferences

Using the check boxes it is possible to select which maths tests will be performed and which ones will be skipped. See CPU maths test for more details.

7.5 Memory Preferences

One of the test patterns can be selected for the test or the default option Cyclic can be used, which will cycle through each test pattern.

7.6 Network Preferences



The image shows the 'BurnInTest Preferences' dialog box with the 'Network' tab selected. The dialog has a title bar with a close button. Below the title bar is a tabbed interface with tabs for CPU-Math, Memory, 2D, Disk, Optical Disk, Parallel Port, Serial Port, Network (selected), USB, PCIe, and Temp / Battery. The 'Standard Network Test' radio button is selected, and the 'Advanced Network Test' radio button is unselected. A button labeled 'Advanced Test Options' is visible. The 'Standard Network Test' section contains a checkbox for 'Test all available NICs' which is unchecked. Below this are ten rows, each with a text input field for a network address and a checkbox for 'Test this address'. The first row has '127.0.0.1' in the input field and the checkbox is checked. The other nine rows have empty input fields and unchecked checkboxes. Below the input fields is a note: 'Enter up to 10 unique host names (eg. www.myhostname.com) or IP addresses. 127.0.0.1 = internal loopback'. The 'Error Options' section contains two radio buttons: 'Every bad packet generates an error' (unchecked) and 'High bad packet ratio generates an error' (checked). To the right of these are two input fields: 'Timeout: 2000 ms' and 'Bad packet ratio: 2.0 %'. At the bottom of the dialog are three buttons: 'Help', 'OK', and 'Cancel'.

BurnInTest Preferences

CPU-Math Memory 2D Disk Optical Disk Parallel Port Serial Port Network USB PCIe Temp / Battery

☒ Standard Network Test ☐ Advanced Network Test Advanced Test Options

Standard Network Test

☐ Test all available NICs

Network Address 1: 127.0.0.1 ☒ Test this address

Network Address 2: ☐ Test this address

Network Address 3: ☐ Test this address

Network Address 4: ☐ Test this address

Network Address 5: ☐ Test this address

Network Address 6: ☐ Test this address

Network Address 7: ☐ Test this address

Network Address 8: ☐ Test this address

Network Address 9: ☐ Test this address

Network Address 10: ☐ Test this address

Enter up to 10 unique host names (eg. www.myhostname.com) or IP addresses.
127.0.0.1 = internal loopback

Error Options:

☐ Every bad packet generates an error ☒ High bad packet ratio generates an error

Timeout: 2000 ms

Bad packet ratio: 2.0 %

Help OK Cancel

Standard Network Test / Advanced Network Test

Use these radio buttons to switch between using the standard and advanced network tests.

Advanced Test Options

When the Advanced Network Test option is selected at the top of the Window, the “Advanced Test Options” button will be available. Clicking this button will open the Advanced Network Test Options Window and allow the options for the advanced test to be set.

Standard Network Test

Test all available NICs

This option will bind each entered address to a different available network card so the traffic will be correctly routed through each network card. If more network addresses are specified than there are network cards available an error will be displayed. If this option is not selected BurnInTest will not force the routing of test traffic and data routing will depend on the operating system routing table. When using this option each test IP needs to be different.

Network addresses

Up to 10 network addresses can be entered for the simultaneous testing of the connection to 10 different locations. These connections can pass via different network interface cards (NIC) or the same one. Unless the "Test All" options is selected the NIC used will depend on the TCP/IP configuration set-up of your system. Each address must be a URL or an IP address and should be located on a different machine on the network.

A URL is the name of a network host, eg. www.hostname.com

An IP address is a sequence of 4 numbers that correspond to a network host. eg. 169.192.0.1

If less than 6 addresses need to be tested uncheck the 'Test this address' check box.

The host selected must be accessible from the computer and capable of responding to the 'ping' command. See the Network Test description for more details. PassMark recommends the selection of a local host to minimize data link problems, which are fairly common on the Internet.

Error Options

Every bad packet generates an error

High bad packet ratio generates an error

BurnInTest can either log an error for every bad (or missing) packet or can be set to only log an error when the number of errors exceed a threshold set by the user. This ratio is expressed as the percentage of bad packets compared to the overall number of packets sent, the first error will be flagged only after a certain number of packets have been sent.

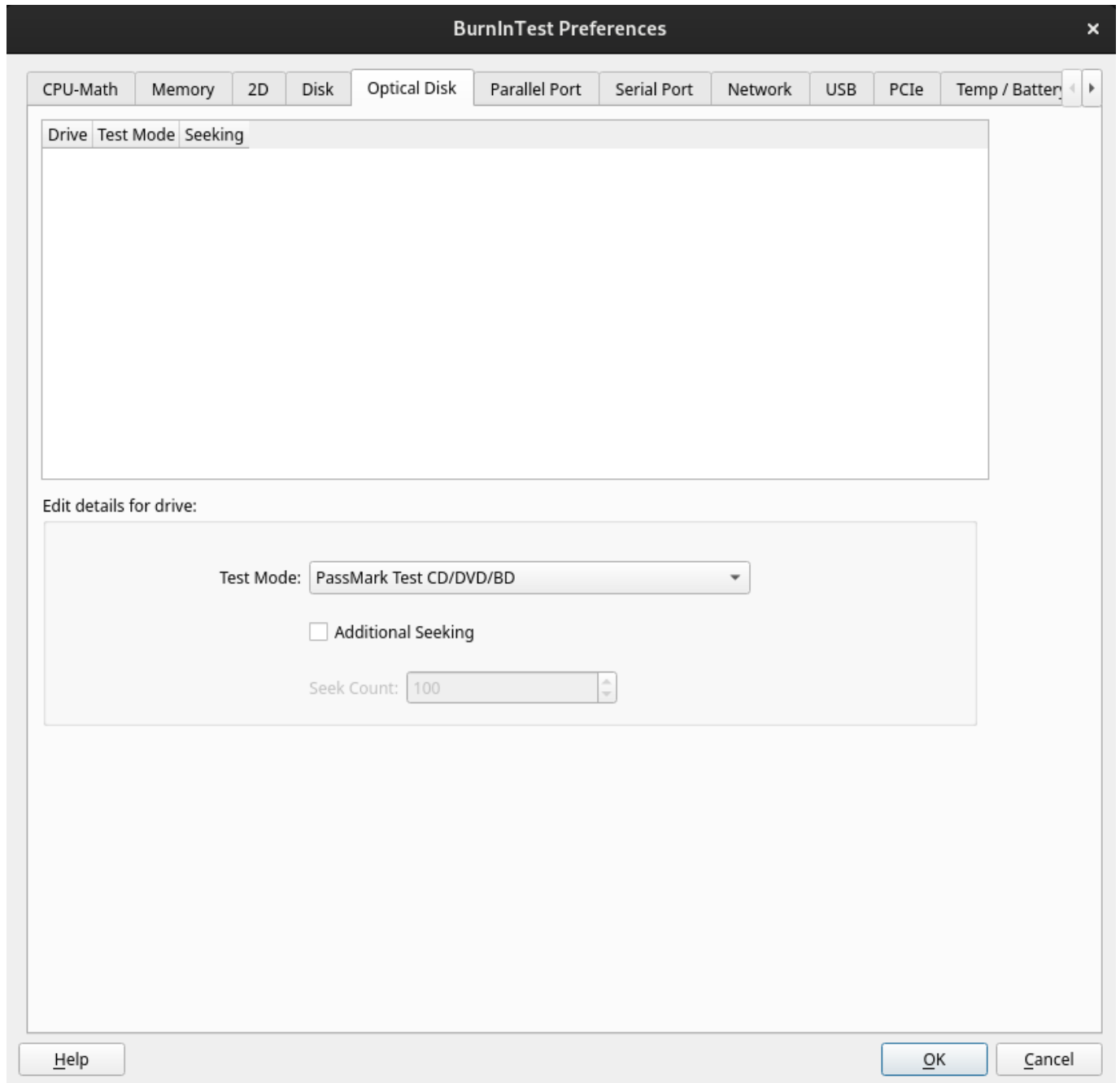
Timeout

This value determines how long BurnInTest will wait for a data packet to be sent or received before an error is reported. The value is measured in Milliseconds. 2000ms (2 seconds) is the default value.

Bad packet ratio

Percentage of bad packets compared to the overall number of packets sent

7.7 CD-ROM/DVD Preferences



CD/DVD Selection and test method

These fields allow the user to select the CD and DVD drives to be used for the CD/DVD test.

Adding a CD/DVD drive

To add a drive letter to the list of disks to be tested, select the drive letter from the drop down list, select the test mode and click on the "Add new CD/DVD to list" button. Only those CD and DVD drives detected by the operating system as displayed in the drop down list.

Up to twenty drive letters can be selected for simultaneous testing. The CD burn test supports a maximum of 1 drive simultaneously.

The type of CD used in the actual test should match the setting in this window. See the CD Test description (page, 9) for more details about test types.

Removing a CD/DVD drive

To remove a disk from the list. Selection one or more drive letters (using CTRL / ALT while clicking) and then click on the "Remove selected CD/DVDs" button.

Additional seeking and Seek count

For PassMark CD/DVD's and Burn CD-RW only. When Additional seeking is selected, seeking to different positions on the CD/DVD or CD-RW will occur Seek Count number of times for each cycle of file read/verify.

7.8 Parallel Port Preferences

Parallel port

The communications port that will be used by the parallel port test, eg /dev/lp0, /dev/lp1 or /dev/lp2 (only ports currently installed will be displayed). See the parallel port test for more details. See the parallel port test (Chapter 3, Parallel Port Test) for more details.

Only a single parallel port may be tested at one time.

Test type

It is possible to select either a "Detect and loopback test" or a "Detect only" test. The, detect and loopback test, requires the use of parallel port loopback plug (see Appendix D). The, detect only, test does not require a loopback plug. When possible use the loopback test as it is a more thorough test.

7.9 Serial Port Preferences

Adding a serial port

To select a serial port for testing, enter the linux serial device name (eg ttyS, TTXRUSB etc), enter the COM port number (0-63) from the right hand drop down list, select the test type, then click on the add button. Up to 64 ports can be selected for simultaneous testing.

Removing a serial port

To remove one or more a serial ports, select the COM port(s) from the left hand list then click on the remove button.

Serial port drop down list

The communications ports that will be used by the serial port test, eg ttyS0.

Test type

For each serial port it is possible to select either a "Detect and loopback test" or a "Detect only" test. The, Detect and loopback test, requires the use of serial port loopback plugs (see Appendix D). The detect only test does not require a loopback plug. When possible use the loopback test as it is a more thorough test.

COM port test settings

Disable RTS/ CTS and DSR / DTR

The checkbox will disable the part of the serial port that checks that the signal pins are functioning correctly.

RTS = Request to send pin

CTS = Request to send pin

DSR = Data set ready pin

DTR = Data Terminal ready pin

This can be useful to avoid errors when testing non-standard serial ports.

Send and Receive timeouts

This is the amount of time in milliseconds that BurnInTest will wait for the successful completion of a Serial I/O operation.

Port speed

The communications speed can be selected. Note that most serial ports do not support speeds above 115200bits/sec.

See the serial port test for more details.

7.10 USB Test Preferences

Automatically select all USB ports with loopback plugs

When selected, this option determines the number of USB tests at the start of each test run by detecting the number of PassMark Loopback plugs correctly connected to the system at that time.

It should be noted that while this improves test automation, it is possible that a USB port is faulty and the loopback plugs has failed to be detected and the number of tests run is less than the required number of tests. When using this option, some type of validation should be done to verify that all of the loopback plugs are correctly connected , e.g. check the Tx and Rx LEDs flash on all plugs during the test.

Maximum number of USB ports

This is the maximum number of USB plugs BurnInTest will attempt to connect to. A max of 20 plugs can be used at the same time.

Run benchmark test at start

This will run a short benchmark test at the start of the USB test and the max read and write speeds will be logged.

USB3 Connection Type

This is the enumeration speed to use for the USB3 plugs: Super Speed 5Gb/s, High Speed 480Mb/s, Full Speed 12 Mb/s or Cycle.

Selecting Cycle will change the speed each cycle (from Super Speed -> High Speed -> Full Speed ...etc).

Refresh list

Select this option to detect all of the PassMark Loopback plug currently connected to the system.

7.11 Sound Preferences

Device Name

Sound device name used for testing. Default is "plughw" but can also be set to "default" (or other sound alias for system, should be left as plughw in most cases).

Max Distortion

Select the maximum distortion allowable for the loopback test. The values available are between 1 and 20 percent. 1% represents a low level of distortion.

7.12 Pre-test Preferences

You may specify the default behaviour of BurnInTest before it begins any test cycle.

None (start immediately)

Immediately starts test(s) without any warning message or reminders.

Display warning message (default)

Displays the default warning message reminding the user of some basic precautions, such as possible port conflicts and the presence of loopback plugs and cables.

Run external application and wait for it to exit

Run an external application and wait for it to exit before starting the tests.

7.13 Post-Test Preferences

Action: Testing PASSED / Action: Testing FAILED

There are different actions that may be taken based on whether a test Passes or Fails, and whether the test is automatically stopped (i.e. the test has run for the set duration) or manually stopped by the user pressing the Stop button.

Auto Stop

Actions specific to automatically stopped tests are described below.

Stop Tests (Normal behavior)

This option will stop all the running tests and close all test windows. The Result window indicating a PASS or FAIL will be displayed. This is the normal default setting.

Run external application and exit

To help with the process of test automation it is possible to have BurnInTest run an external application, and exit automatically, after the tests have been completed. This could be used, for example, to call an application or batch file that would undertake a different type of testing (e.g. PassMark PerformanceTest) or customer specific reporting.

Exit and Reboot PC

This will exit BurnInTest and issue a system reboot command.

Manual Stop

Actions specific to manually stopped tests are described below.

Stop Tests (Normal behavior)

This option will stop all the running tests and close all test windows. This is the normal default setting.

Display option to run application

This option allows tests to be stopped prior to reaching the test duration, while maintaining a post test process that is consistent with the automatic stop actions (such as customer specific reporting). BurnInTest can display an option to the user asking whether they want to run the Post test application after a test run is manually stopped. If the user selects to run the external application, BurnInTest will exit after running the application. If the user selects not to run the external application, then BurnInTest will not exit.

Always display result Window

By default, this option is unchecked, and the result window indicating PASS/FAIL is only displayed after the Auto Stop, Stop Tests option. The user may specify to display this result window for the other options (i.e. manually stopped tests, and running an external application). If this option is checked the result window will be displayed after any external application is run, and before exiting BurnInTest.

Use System Beep on auto stop

When selected, the system beep will occur when a test is automatically stopped. This is to alert a nearby user that the test is completed.

External Application

This specifies the application to be run if the appropriate option above is selected.

Parameters

This specifies any parameters to launch the external application with.

A number of variable parameters are supported. These parameters will be substituted with fixed strings enclosed in double quotes (""). These are:

%RESULT - "PASS" or "FAIL" will be substituted if the test is stopped automatically. If the test is manually stopped by the user, "PASS (manual abort)" or "FAIL (manual abort)" will be substituted.

%SERIAL – The serial number will be substituted.

%MACHINETYPE – The machine type will be substituted.

%NOTES – The notes will be substituted.

7.14 Error Handling Preferences

Action on Error

When BurnInTest detects an error, three options are available:

Continue

The error counter is incremented and testing continues.

Beep and Continue

The error counter is incremented, the computer beeps and testing continues.

Auto Stop Tests

All windows (except the main window) are closed and all the tests are stopped. The failed window is then displayed.

7.15 Logging Preferences

Automatic logging

Logging to disk is switched on and off with the "Logging on" option. When disk logging is on, BurnInTest will log System information, results, result summaries, and detailed error descriptions to files based on the following user settings. This log can be viewed by clicking on the Event Log tab.

Log file

The log directory allows the directory and filename prefix to be specified.

A new set of log files will be created when Logging On is switched on, any of the logging options (Log Directory, Detail Level, ASCII/HTML Format) are changed, or when BurnInTest is started with Logging switched on. The only exception is when BurnInTest is automatically stopped with a REBOOT or REBOOTEND scripting command, and then the original log files will be re-opened and appended to.

The log files are closed and available for use by another application when Logging On is switched off, any of the logging options (Log Directory, Detail Level, ASCII/HTML Format) are changed, or when BurnInTest is exited or automatically stopped with a REBOOT or REBOOTEND scripting command.

Note: Text and HTML disk logs can be manually created using "File", "Save results report". This will save System information, a result summary and the recent Detailed Error log history (up to 1000 entries).

Special Flags

Several special flags are available to be used in the logging path and filename;

%SYSTEMSERIAL% - This will be replaced with either the serial number as set in the Report Information and if this value has not been set, the system serial number as retrieved by the dmidecode tool. For example "logs\BitLog_%SYSTEMSERIAL%" can become "logs\BitLog_SN123456"

%DATE% - This will be replaced with the date in YYYYMMDD format.

Pre-fix

If you wish to prefix the standard BurnInTest Text or HTML log filenames you can type a prefix string into the “Prefix” edit box. For example, if you want to prefix the standard BurnInTest log filenames with the text “SPECIAL_TEST_” then just enter, SPECIAL_TEST_.

Multiple files (time-stamped) / Single File

If 'multiple files' option is selected, the files will be appended with _YYMMDD_HHMMSS.log, .htm or .trace, specifying the Year, Month, Day, Hour, Minute & second that the disk log file was created.

Report type

The format of the Result summary and Normal log files may be specified to be either ASCII text or HTML via the “Text (ASCII)”, “HTML” or “Customer certificate” buttons. This setting will also determine the format of the Trace logs.

Report detail**Log detail level**

A “Log detail level” may be selected for logging. The options that may be selected are:

- Result Summary: Produces a .log or .htm file containing system information, a summary of the tests run (eg. Which tests, the number of operations and errors per test) and a summary of the serious and critical errors during a test run (cleared at the end of a non-scripted test run or at the end of the script for a scripted test run).
- Normal: Produces a .log or .htm file containing the “Result Summary only” information and detailed errors (1 or 2 lines per error).

Trace detail level

A “Trace level” may be selected for detailed logging. The options that may be selected are:

- No Trace log: No trace log is produced. This is the default setting.
- Activity Trace 1: Produces a .trace with the “Normal” information and a summary of test activities (eg. The starting and stopping of tests).
- Activity trace 2: Produces a .trace file with the “Activity Trace level 1” information and a detailed description of the test activity.

Max file size

Specifies the approximate maximum file size in terms of the number of lines in either the log or trace files.

Periodically log result summary reports during a test**Logging interval**

Change this value if you would like to log interim results summary reports during the test (every X minutes). This will log the number of operations, errors etc at that periodic point in time.

One use of this option is to help obtain information about the testing status of a system prior to a system crash on an unstable system. Note: All error events are written in real time, so the main use of this option will be if there are few errors occurring.

Pre-test logging options

Test results, including the detailed error and status log history, can either be cleared each time a new test run is started or test results can be accumulated across several test runs.

Three options are available for clearing/accumulating test results and creating/appending log files. These options are:

- Clear test results and create a new log
- Clear test results and append to the existing log
- Accumulate test results and append to the existing log: error count, cycle count and operations count values are the total values that have been accumulated across all test runs on this machine. Accumulation will also occur across scripted reboots, sleeps and execute commands.

7.16 Plugin preferences

You can specify up to 5 external plugin applications.

Run at the same time

Choose this option to run all plugins at the same time

Run each separately

Choose this option to run plugins separately, launching the next one when the previous finishes

Ignore "no operations" timeout

When selected, the "no operations" timeout message will not be logged

Specify plugin application 1..5

Application

Specifies the plugin filename and command line arguments. Due to the way the shared memory is accessed in Linux (it relies on the path and application name) each plugin needs to have a different name or be in a different path, you can't start multiple plugins using the same application path.

Run as pre-test plugin

Some plugin tests may require manual user testing, for example, testing a keyboard. You can specify that a plugin be run before the main set of tests, such that the plugin test will run, exit on completion of that test, and then the main set of tests will be run. This option is available for all plugins.

7.17 PCIe Slots

Automatically select all PCIe slots with PassMark loopback cards

When selected, this option determines the number of PCIe tests at the start of each test run by detecting the number of PassMark PCIe test cards correctly connected to the system at that time.

It should be noted that while this improves test automation, it is possible that a PCIe slot is faulty and the PCIe test card has failed to be detected and the number of tests run is less than the required number of tests. When using this option, some type of validation should be done to verify that all of the PCIe test cards are correctly connected.

Number of PCIe slots to test

This is the number of PCIe test cards that BurnInTest attempts to connect to. In general you will want to set this value to be equal to the number of PCIe test cards that you would typically have connected to the machine. A maximum of 10 cards may be used at the same time.

See the PCIe slot PCIe slot and PSU test for more details.

You will need to download, build and load the PCIe test card device driver for Linux before BurnInTest can access the card.

7.18 Management Console

Manage BurnInTest with the Console Manager

When this option is selected, BurnInTest will send system information, test status and test result information to the BurnInTest Management Console Web server application.

Server address

Specifies the address of the BurnInTest Management Console Web server application. For example:
`http://<Server address>/htdocs/BITConsoleMsgHandler.php`

Username

User name to be used for HTTP authentication

Password

Password to be used for HTTP authentication

See also

Management Console

7.19 Battery Preferences

BurnInTest can monitor battery levels and report an error if the battery is below a set capacity value in absolute or percentage terms.

Battery pre-test

Select this option to check the battery capacity before the main tests start, i.e. as a pre-test.

Battery main test

Select this option to check the battery capacity during the main test run, i.e. while the other main tests are running. The battery will be checked once per minute.

Use full charge capacity

Use the battery's current fully charged capacity (the battery capacity when the battery is reporting it is at 100% charge, from POWER_SUPPLY_ENERGY_FULL) instead of current capacity (POWER_SUPPLY_ENERGY_NOW) for comparison. When compared to the design capacity this gives an indication of how the battery has aged.

Use current charge

Use the current battery charge capacity (POWER_SUPPLY_ENERGY_NOW) instead of full charged capacity (POWER_SUPPLY_ENERGY_FULL) for comparison.

Use threshold (percent)

Select this option to use a percentage threshold value, eg 90%. If the measured value (full charge or current charge) drops below this level when compared to the battery's design capacity an error will be logged.

Use absolute threshold

Leave unchecked to use an absolute battery value.

Threshold

The absolute (e.g. 30000mA) or percentage battery value (e.g. 80%), eg 5000 mWh. If the measured value (full charge or current charge) when compared to the battery's design capacity has a difference greater than this value then an error will be logged.

7.20 Temperature Preferences

It is possible to monitor, graph and log the temperature of some hardware components while running BurnInTest tests. Temperature values are displayed and graphed on the Temperature tab and included in BurnInTest reports. Temperature values will also be stored in the log files at periodic intervals if a periodic logging interval is specified.

Automatically select temperature sources

When this option is selected up to 64 temperature sources will be automatically at the start of each test run. This allows for portability of configuration files across systems with different temperature sources, as well as reduced configuration time.

BurnInTest will determine the number of CPU and Hard disk temperature sources supported (i.e. sources per temperature type) at the start of a test run. BurnInTest will attempt to select at least one temperature source from each type of temperature source (e.g. at least 1 CPU, 1 HDD) supported. BurnInTest will also attempt to use as many temperature sources as possible e.g. 2 x CPU, 4 x HDD. Temperature thresholds are set to the value of 90C and are not based on the hardware type.

Temperature sources / Event thresholds

Up to 64 temperature sources may be selected for monitoring.

It is possible to raise a temperature event (error), "Temperature threshold exceeded", on a user defined temperature threshold being exceeded. For each temperature source a maximum temperature value can be specified. If the temperature exceeds this value, a temperature event will be displayed and logged.

Stop testing on temperature threshold exceeded

Select this option to automatically stop testing on the temperature threshold being exceeded event. This could prevent system shutdown or in extreme cases prevent hardware damage from over temperature (e.g. due to a CPU fan failure).

Sample collection interval

The frequency that the temperature sources are checked for the temperature values can be configured by specifying the frequency in seconds.

8 Test Selection and Duty Cycles

From this window it is possible to configure the automatic stopping of tests, select the tests to be performed and the level of load placed on the system.

Test selection and duty cycles [X]

Auto stop after minutes or cycles (0 means run forever)

CPU Math <input checked="" type="checkbox"/>	<input type="range" value="50"/>	50	2D Graphics <input checked="" type="checkbox"/>	<input type="range" value="50"/>	50
CD/DVD/BD <input checked="" type="checkbox"/>	<input type="range" value="50"/>	50	3D Graphics <input type="checkbox"/>	<input type="range" value="50"/>	50
RAM <input checked="" type="checkbox"/>	<input type="range" value="50"/>	50	Disk(s) <input checked="" type="checkbox"/>	<input type="range" value="100"/>	100
Com Port(s) <input type="checkbox"/>	<input type="range" value="50"/>	50	Network <input checked="" type="checkbox"/>	<input type="range" value="50"/>	50
Parallel Port <input type="checkbox"/>	<input type="range" value="50"/>	50	USB <input type="checkbox"/>	<input type="range" value="50"/>	50
Sound <input type="checkbox"/>	<input type="range" value="50"/>	50	Plug-in <input type="checkbox"/>	<input type="range" value="50"/>	50
PCIe <input type="checkbox"/>	<input type="range" value="50"/>	50			

Battery ☐ Temperature ☐

Windows Update Interval

Update windows after every: (100-10000 msecs)

Select the tests to perform and their Duty Cycle. (1 = Min. load, 100 = Max. load)

Auto Stop

BurnInTest can be set to automatically stop a test run by either configuring a test duration or a number of test cycles.

Test duration

This option allows the user to select the period of time that tests are to run for. After the time has expired all windows (except the main window) are closed and all the tests are stopped.

The duration, in minutes should be entered in the “Auto Stop after n Minutes” field. A value of 0 in this field means that the tests will run until a manual Stop command is issued via the button bar/menu or the number of test cycles has been reached. The maximum test time in the unregistered shareware version is 15 minutes per test. After the software is registered the maximum auto stop time is 10 days. (the registered version can also run forever if the manual stop option is selected).

Number of test cycles

This option allows the user to select the number of cycles that tests are to run for. After all current tests have undergone the configured number of cycles in testing, all windows (except the main window) are closed and all the tests are stopped.

The number of cycles should be entered in the “or n Cycles” field. A value of 0 in this field means that the tests will run until a manual Stop command is issued via the button bar/menu or the test duration has been reached.

NOTES:

- 1) Both options are based on the current test run and not accumulated results.
- 2) The test duration and number of cycles are both used to automatically stop a test run (with a logical OR). For example, if duration is set to 15 minutes and number of cycles is set to 3. The test run will be stopped when the first of either 15 minutes OR 3 cycles is reached. If you want to automatically stop a test run based only on duration, set the number of cycles to 0 (ie. Run for the set duration OR forever). If you want to automatically stop the test run only based on the number of test cycles, set the test duration to 0 (ie. Run forever OR for the set number of cycles).
- 3) If you need to run each test a single time, one after the other (ie. In series, rather than in parallel), a script should be created to: Set the number of cycles, run test1, run test2, ..., run test n. See SETCYCLES.

Test Check Boxes

Each test has an associated check box that can be used to turn the test on or off.

Slide bars and Duty cycle

Each test has a slide bar and an associated edit box. The slide bar allows the user to determine the “duty cycle” for each test. A low duty cycle means that a delay will be inserted during the execution of the test, reducing the load on the system and reducing the number of operations performed during any particular period of time. A high duty cycle corresponds to higher load.

A value can also be directly entered into the edit box. The background color of the edit box will vary from dark red to white to bright green depending on the duty cycle selected.

Buttons

The buttons at the bottom of the window allow all tests to be activated or deactivated with a single click. The default values can also be restored. The settings selected are saved when the OK button is clicked.

Selecting which tests to run

In order to help select which tests to run, here are some general guidelines:

Only select tests that match your hardware

The tests you select should correspond to the hardware installed in the PC. For example, if your computer doesn't have a serial port, there is no point having the serial port test enabled. This will only result in a lot of errors being generated.

Selecting tests for specific testing

If you suspect a problem with a particular device, (eg the RAM), leave the other tests turned off and just run this particular test at 100%. This will maximise the load on this element.

Selecting tests for general burn in testing

For a general burn in, select a variety of different tests. The RAM and Disk tests are the most important. Select the other tests based on how you plan to use the computer. For example if the machine is to be used as a server, then the Network test, Tape drive test, CPU tests and CD test should also be enabled. In general it is better to test those elements that will receive the most usage once the machine is put into active use. For example the floppy drive test could be left off, (or tested at a low duty cycle), if the floppy drive is not critical for the machines intended use.

Optimise the load

As all the tests run at the same time in different threads. Some care should be taken to ensure that important tests are not starved of the CPU and run too slowly. Thus it can be advantages to initially leave the CPU tests off and run with the other tests. Then add in the CPU tests but adjust the duty cycle down until the CPU load just hits 100%.

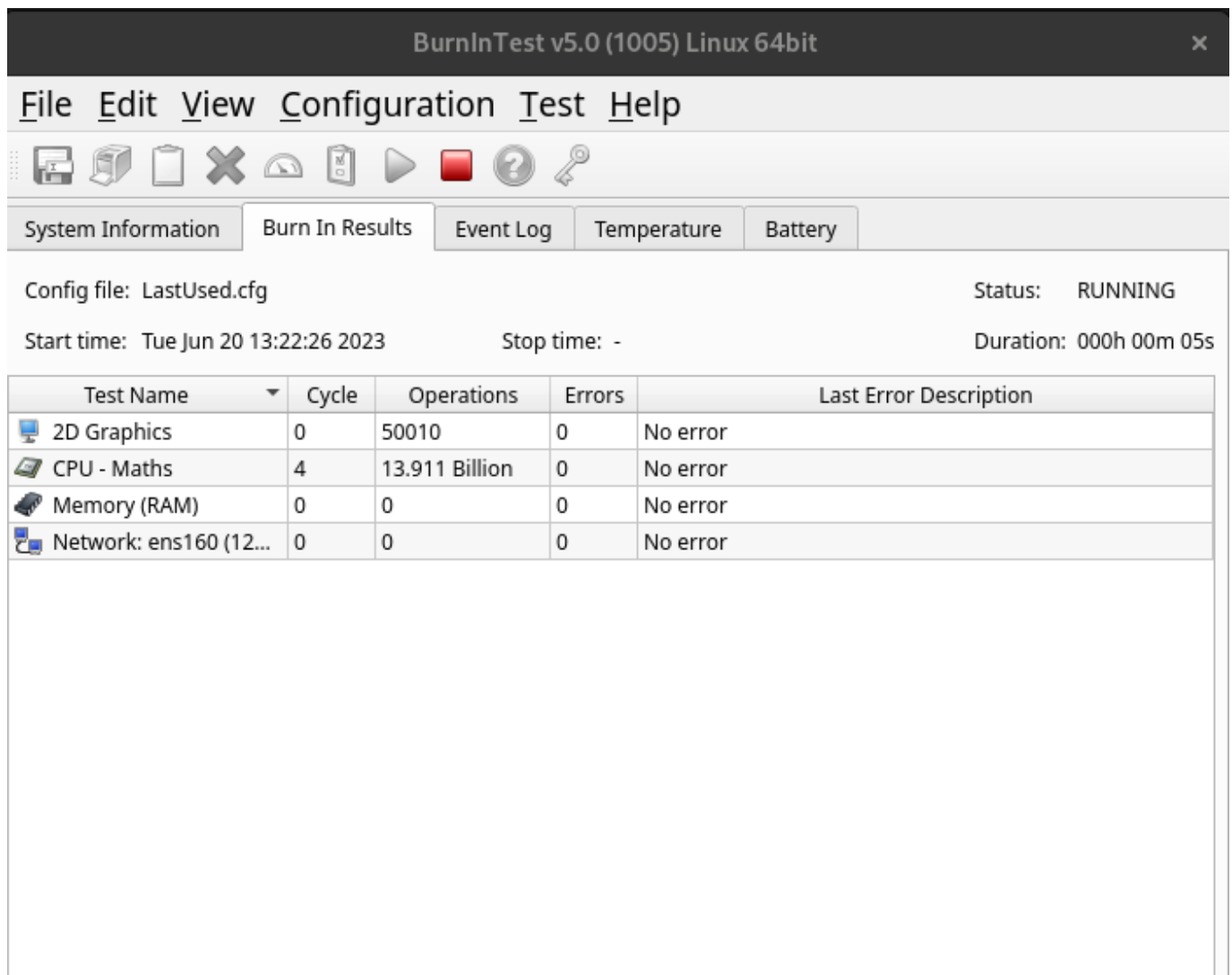
For example, assuming that the CD and disk drive are critical parts of your system, run just these tests at 100% duty cycle, then note the load on the CPU. Then add in the RAM and CPU tests at a lower duty cycle in order to fully load the CPU.

Note that there is no point trying load up the CPU to more than 100%. Adding more load once the CPU is running at 100% doesn't result in any more processing being done. The CPUs available processing time is just redistributed and all the running processes run more slowly.

Experiment

As just about everyone has different requirements don't hesitate to experiment with the settings to obtain the best result in your environment.

9 Interpreting the results



The screenshot shows the BurnInTest v5.0 (1005) Linux 64bit application window. The window has a menu bar with File, Edit, View, Configuration, Test, and Help. Below the menu bar is a toolbar with icons for file operations, test execution, and help. The main window is divided into tabs: System Information, Burn In Results, Event Log, Temperature, and Battery. The Burn In Results tab is active, showing the following information:

- Config file: LastUsed.cfg
- Status: RUNNING
- Start time: Tue Jun 20 13:22:26 2023
- Stop time: -
- Duration: 000h 00m 05s

Test Name	Cycle	Operations	Errors	Last Error Description
2D Graphics	0	50010	0	No error
CPU - Maths	4	13.911 Billion	0	No error
Memory (RAM)	0	0	0	No error
Network: ens160 (12...	0	0	0	No error

Status display

In the main window there is a display that shows a summary of all the results of all the tests that are currently running.

Test Name

This column shows a picture depicting the test type and the name of the test. Only those tests actually running are displayed.

Cycle

The number of test cycles that have been executed for a particular test. The meaning of a 'test cycle' varies from test to test. For example, for the Hard disk test it is the number of file write / verify cycles that have occurred. See the test description (chapter 3) for more details about the significance of this field.

Ops (Operations)

The number of test operations that have been executed for a particular test. The meaning of a 'Operation' varies from test to test. For example, for the Hard disk test it is the number of bytes that have been written or verified. See the test description (chapter 3) for more details about the significance of this field. Note: The values are expressed in Units, Millions, Billions, Trillions and Quadrillions.

Default Error View

Errors

The number of errors that have been encountered while the test has been executing. This value should normally stay at zero. A value of greater than zero indicates there has been an error in the hardware or the software controlling the hardware. In some cases it is possible for the computer to self-detect an error. (such as the math's and disk tests). In other cases the user must check themselves that no error has occurred (eg. Is there sound coming from the speakers? Are printouts complete, clear and legible?).

Last error description

This is the description of the last error that occurred. This will give some indication as to the cause of the error. See common error messages for a description of the errors that may be encountered.

Error by Category View

Errors are tabulated based on their severity as defined in the file BITErrorClassification.txt. The number of errors for each test, for each level of severity: Critical, Serious, Warning and Informational will be displayed:

BurnInTest v5.0 (1005) Linux 64bit

File Edit View Configuration Test Help

System Information Burn In Results Event Log Temperature Battery

Config file: LastUsed.cfg Status: IDLE

Start time: Tue Jun 20 13:22:26 2023 Stop time: Tue Jun 20 13:23:23 2023 Duration: 000:00:57

Test Name	Cycle	Operations	Critical	Serious	Warning	Info
2D Graphics	6	1.184 Million	0	0	0	0
CPU - Maths	72	174 Billion	0	0	0	0
Memory (RAM)	1	6.428 Billion	0	0	0	0
Network: ens160 (12...	1	9240	0	0	0	0

Display errors by category

See Detailed Error log history for an explanation of the logged output.

See Test Window Display for examples of each test window.

9.1 Test Window display

Each test type has its own display window. The contents of the windows are described in the various tests. An example of some of these test windows is included below.

CPU - Maths [X]

Millions of Operations

Test	Completed	Verified
General	1.6	9341.6
Floating	3.6	3323.6
Extensions	9.7	9959.7
Primes	10.83	10.83

*Extensions: SSE SSE2

Memory (RAM) [X]

Cycle 0: Testing

8%

Pattern: Sequence (0,1,2...255)

Total: 7927.36 MB

Free: 2635.69 MB

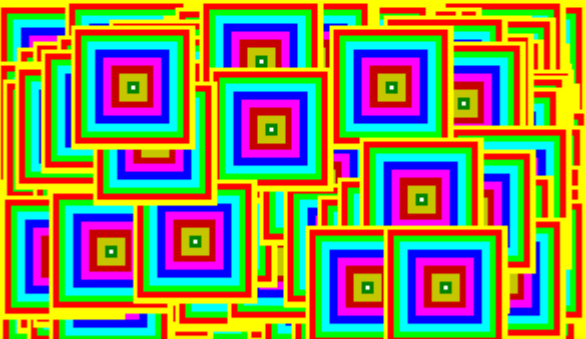
Test: 1200.00 MB

Written: 100.50 MB

Verified: 0.00 MB

2D Graphics [X]

Squares Drawn: 0



Network Test [X]

Network: 127.0.0.1

Cycle: Cycle 0:

NIC (Server): ens160 (127.0.0.1)

Sent (pkts): 5

Received (pkts): 4

Average delay: 500.39 ms

Maximum delay: 500.43 ms

Current delay: 500.43 ms

Flow rate: 1.00 pkt/s

9.2 Detailed Error log history

The detailed error log history can be opened from the 'View / Error log file' menu. It contains a record of recent errors and status messages that have occurred during a test run. It is possible to save a copy of this log of recent history using the 'File / Save test log' menu item.

The log can contain up to 1000 messages. After 1000, the oldest messages are overwritten by the newer ones. The log window is updated in real time and can left open while a test run is in progress.

Time stamp

The time stamp records the time when the event occurred. The format is, YYYY-MM-DD HH:MM:SS

Test type

The test type refers to the name of the test which originated the event. Examples are, "RAM", Floppy Disk". The word "Status", indicated that the event is not an error but just a status message.

Error Classification

Where logging records an error, the severity of this error (CRITICAL, SERIOUS, WARNING or INFORMATION) is logged. Where an informational line (related to this error) follows the error line, the error classification is not repeated (to aid determination of the number of unique errors per classification).

Details

This is a text field that contains additional information about the error (or status message). See Appendix C for a description of the errors that may be encountered. In some cases a log entry may be spread across two lines. The 1st line will contain some general information about the error and the 2nd line will contain more detail.

Some error messages may contain internal program variables (and thus be fair cryptic). These are not always intended to be interpreted by the user but help us track down faults in our software.

Configuring Error Classifications

Each of the standard errors detected by BurnInTest can be configured to include a customer specific string and error classification. The file BITErrorClassification.txt (in the software directory) contains the error text and classification used by BurnInTest.

The file is formatted as a Comma Separated File with four fields:

1. Error number – this must not be changed by the customer;
2. Error description text – this may be changed. This must be a string enclosed by quotes;
3. Error classification – this may be changed. This must be one of the following text values (not in quotes): CRITICAL, SERIOUS, WARNING, or INFORMATION

A number of errors may be configured to be ignored (not reported as errors), as they may be considered 'below the radar'. These are configured by setting the Error classification to NONE. These are:

- 138,"Warning: Could not lock parallel port",WARNING,PP
- 146,"Frame could not be displayed and was skipped",INFORMATION,
- 182,"Video Playback is dropping samples",WARNING,

Other errors may not be configured to be ignored in this way.

4. Customer comment – this is optional and may be changed – it is ignored by BurnInTest.

Comment lines may be inserted by using a "#" at the start of the line. These lines will be ignored by BurnInTest.

An example is shown below:

```
# PassMark BurnInTest,,,  
# Error message text and classification,,,  
0,"No errors",NONE,  
1,"Incorrect mathematical addition",SERIOUS,  
2,"Incorrect mathematical subtraction",SERIOUS,  
3,"Incorrect mathematical division",SERIOUS,  
4,"Incorrect mathematical multiply",SERIOUS,
```

10 Menu Options

File

Load Test Configuration

A test configuration that has been previously saved to disk can be loaded using this menu option. The loaded configuration will set all the parameters available in the 'test duty cycles' and 'test preferences' windows (see above) to the values contained within the file. Using this menu option in conjunction with the 'Save Config as' function it is possible to create a number of standard test configurations. See section 0, Maintaining multiple test configurations, for more information.

Save Test Configuration

This option allows the current configuration settings to be stored in a file for later use. All the values available in the 'test duty cycles' and 'test preferences' windows (see above) will be saved into a file, whose name is selected by the user. Using this menu option in conjunction with the 'Load config' function it is possible to create a number of standard test configurations. See section 0, Maintaining multiple test configurations, for more information. Saved configuration files can also be loaded automatically by including them on the command line.

Save Results Report

Save the current results to a location as specified by user.

Print

Allows the contents of the main window to be printed

Exit

Exits from the application.

Edit

Copy Results to Clipboard

Allows the contents of the main window to be copied to the clipboard in HTML Format. The results can then be pasted as a graphical image into other documents that support this format.

Clear All Results

Clear the results produced so far, resetting all values to zero. This is done automatically each time a new test run is started (unless the accumulated logging option is set in the preferences window).

View

Error by category

Toggles the main display between the following two error table views:

1. Number of error and last error description per test, and
2. Number of errors for each category per test (that is, CRITICAL, SERIOUS, WARNING AND INFORMATION).

Configuration

Test Selection & Duty Cycles

Displays the test set-up dialog window that allows a subset of tests to be selected for execution and a duty cycle to be set for each test. See here for more details.

Test Preferences

This option allows the user to customize some aspects of the way BurnInTest works. See here for more details.

Report Information

Displays the report information window, which allows general report information such as the machine type, serial number, customer name and technician's name to be entered. See this section for more information.

Edit Error Classifications

Opens the error classification file default text edition to allow errors to be re-categorized and the descriptions changed.

Test**Start Test Run**

Runs the tests selected in the 'Test duty cycles' window. Each test will be run in it's own window using it's own execution thread. This allows multiple tests to be executed simultaneously. The results from each test are summarized in the main window. Depending on what tests are being started, the speed of the machine and the selected duty cycles; the tests may take several seconds to start.

After tests have been started, menus and buttons are disabled, and no another actions can take place until the tests are stopped manually or automatically.

Stop Test Run

Stops all the tests that are currently running. It may take several seconds to stop all the tests, remove test files from the disk and de-allocate system resources. If one of the test windows is the active window, Alt-F4 can also be used to stop all the tests. Individual tests cannot be stopped after they have been started, as all tests will be stopped at the same time.

Execute Script

Load a script file to execute, see scripting for more information.

Help**Help**

Opens this documentation.

BurnInTest Homepage

Opens the PassMark webpage for BurnInTest Linux.

About

Displays registration information and version information for the copy of BurnInTest.

11 Scripting

BurnInTest includes a simple scripting language that allows tests to be executed in a sequence.

A PASS/FAIL indication for all of the tests included in the script can be displayed at the end of the scripted test run. To achieve this you should select "Accumulate logs until manually cleared" from the Logging Preferences and results should be cleared before or at the start of the scripted test run.

Script files are ASCII text files that you can create with a text editor (e.g. notepad). The file name must end with the extension '.bits'. To start a script use the 'Test / Execute script' menu item.

Each script command must appear on its own line in the text file and the entire command must appear on a single line. (i.e. a single command can not be split across multiple lines).

The command and its parameters must be separated by one or more spaces. Comments can be included by starting the line with the '#' character.

The scripting is similar to the Windows version of BurnInTest but a few keywords are not available (REBOOT, REBOOTEND, SETPLUGIN, EXECUTEWAIT).

The following commands are available in the current version of the software;

EXECUTE
EXIT
LOAD
LOADXML
LOG
LOOP
SETDURATION
SETCYCLES
SETDUTYCYCLE
SETSERIAL
SETMACHINETYPE
SETNOTES
SETCUSTOMER
SETTECHNICIAN
SETMANAGEMENT
SETLOG
SETPCIE
SETDISK
SETDISKDESTRUCTIVE
SETSERIALPORTS
SLEEP
RUN

EXECUTE COMMAND

Overview:

Executes an external file and continues processing the script.

Syntax:

EXECUTE <Filename> <Parameters>

Parameters:

<Filename> The name of the file to execute. The file name must be enclosed in double quotes (""). The file must be an executable.

<Parameters> Any command line parameters which you wish to pass to your executable. Each new parameter should be separated with a ',' character , eg param1,param2.

Examples:

#Start up text editor with some results before starting tests.

EXECUTE "/usr/bin/emacs" "/tmp/doc.txts"

RUN CONFIG

EXIT COMMAND

Syntax:

EXIT

Parameters:

none

Note: BurnInTest will exit with a return code indicating whether the tests run during the script passed or failed. An exit code of 0 means SUCCESS, while an exit code of 1 means FAILURE.

Examples:

RUN CONFIG

#Exit BurnInTest and return an exit code to the batch file

EXIT

LOAD COMMAND

Syntax:

LOAD "<File name>"

Parameters:

<File name> The full path name to a configuration file. This configuration file is expected to be in BurnInTest binary format (previously created from within BurnInTest GUI version). A partial path name can be used to load a configuration file from the BurnInTest executable directory. A loaded configuration file will overwrite all preferences currently selected. The file name must be enclosed in double quotes ("")

Examples:

#Load the low load configuration file that we created earlier

LOAD "/tmp/LowLoad.bitcfg"

LOADXML COMMAND

Syntax:

LOADXML "<File name>"

Parameters:

<File name> The full path name to a configuration file. This configuration file is expected to be in BurnInTest text format, such as one created for use with the BurnInTest command line version. A partial path name can be used to load a configuration file from the current BurnInTest executable directory. A loaded configuration file will overwrite all preferences currently selected. The file name must be enclosed in double quotes ("")

Examples:

#Load the low load configuration file that we created earlier

LOAD "/tmp/LowLoad.txt"

LOG COMMAND

Overview:

Writes text to the detailed error and status log history.

Syntax:

LOG <Text>

Parameters:

Text The text to be added to the detailed error and status log history.

Examples:

#Start an S1 sleep and log the start and stop times

```
LOG "Sleep S1 Duration 60 seconds starting"
EXECUTEWAIT SLEEPER -S1 -D 60
LOG "Sleep S1 Duration 60seconds complete"
```

LOOP

Syntax:

```
LOOP <number of iterations>
    <Tasks to repeat>
}
```

Parameters:

< number of iterations > The number of times the Tasks within brackets will be repeated.

Examples:

```
LOG "Start"
SETDURATION 1
LOOP 3
    LOG "CPU Test"
    RUN MMX
}
```

SETDURATION COMMAND

Syntax:

```
SETDURATION <Duration>
```

Parameters:

<Duration> Sets the test duration in minutes. Using this command is the same as changing the auto-stop period from the preferences window.

Examples:

```
#Set the test duration to 90 seconds
SETDURATION 1.5
```

SETCYCLES COMMAND

Syntax:

```
SETCYCLES <Number of test cycles>
```

Parameters:

< Number of test cycles > Sets the number of test cycles that will lead to an automatic stopping of the test runs after all selected tests have reached or exceeded this number of test cycles. Using this command is the same as changing the auto-stop number of cycles from the preferences window.

Examples

```
#Set the number of test cycles to 1
SETCYCLES 1
```

```
#Run each test one cycle in series.
SETCYCLES 1
SETDURATION 0
LOG "Run Maths Test"
RUN MATHS
LOG "Run MMX Test"
```

RUN MMX
Etc...

SETDUTYCYCLE COMMAND

Syntax:

SETDUTYCYCLE <Test Name> <Duty setting>

Parameters:

<Test Name> See below for a list of all test names.
<Duty setting> Sets the duty cycle for the specified test to the value specified. Values must be between 1 and 100. Using this command is the same as changing the duty cycle value from the Test duty cycle window.

Examples:

#Set the disk test to maximum load
SETDUTYCYCLE DISK 100
#Set the CPU test to medium load
SETDUTYCYCLE DISK 65

SETSERIAL COMMAND

Syntax:

SETSERIAL < Serial Number >

Parameters:

<Serial Number> The serial number string. Must be enclosed in double quotes ("").

Examples:

#Set the serial number
SETSERIAL "1234-shdfgdhs-GHGHG"

SETMACHINETYPE COMMAND

Syntax:

SETMACHINETYPE < Machine Name>

Parameters:

< Machine Name > The Machine Name. Must be enclosed in double quotes ("").

Examples:

#Set Machine type
SETMACHINETYPE "Dell XPS800"

SETNOTES COMMAND

Syntax:

SETNOTES <Notes >

Parameters:

< Notes > The Notes. Must be enclosed in double quotes ("").

Examples:

#Set Notes
SETNOTES "Test notes defined by the external application."

SETCUSTOMER COMMAND

Syntax:

SETCUSTOMER < Customer name >

Parameters:

<Customer name> The customer name string. Must be enclosed in double quotes (""). Can include an environment variable that will be expanded.

Examples:

#Set the customer name

SETCUSTOMER "The big PC store"

SETTECHNICIAN COMMAND

Syntax:

SETTECHNICIAN < Technician name >

Parameters:

<Technician name> The technician (or testers) name. Must be enclosed in double quotes (""). Can include an environment variable that will be expanded.

Examples:

#Set the technician name

SETTECHNICIAN "Bill Smith"

SETMANAGEMENT COMMAND

Overview:

Set the BurnInTest Management Console preferences.

Syntax:

SETMANAGEMENT MANAGE <parameter> SERVER_ADDRESS <parameter> ID <parameter>

Parameters:

MANAGE (Turn BurnInTest Management Console usage on/off): [YES | NO]

SERVER_ADDRESS: specify the BurnInTest Management Console URL within double quotes.

ID: Specify the machine ID used when connecting to the management console

Examples:

SETMANAGEMENT MANAGE yes SERVER_ADDRESS

"http://www.passmark.com/burnintest/management/mgtconsolemsgHandler.php" ID Test1234

SETLOG COMMAND

Syntax:

SETLOG "<Filename>"

Parameters:

<Filename> The name of the log file. The file name must be enclosed in double quotes (""). This can be a path, eg "/tmp/bitlog/" and the default filename will be used, or a filename, eg "/tmp/bitlog/scriptlog" and BurnInTest will append .log to the filename before creation.

Examples:

```
#Set log file with full path
SETLOG "/tmp/plugin_log"
#Set log file with path relative to BurnInTest application path
SETLOG "plugin_log"
```

Note:

Loading a saved configuration file will overwrite this setting. The currently saved logging settings, such as trace settings, are used when loading a script.

SETPCIE COMMAND

Syntax:

```
SETPCIE <Num >
```

Parameters:

< Num > Number of PCIe cards to test

Examples:

```
#Set Notes
SETPCIE 2
```

SETDISK COMMAND

Overview:

Set the disk test options. The first parameter for the command is always disk mount point except for 'ALL' parameter which tests all mounted drives.

This can only be used with mounted devices.

Syntax:

```
SETDISK DISK <parameter> MODE <parameter> FILE <parameter> BLOCK <parameter> SEEK <parameter>
DUTY <parameter> MAXCYCLES <parameter>
SETDISK ALL <parameter>
```

Parameters:

DISK: Disk Name (mount point)
This parameter must be specified first.

MODE (Test Mode):
CYCLIC (Cyclic)
SEQUENTIAL (Sequential data pattern)
SEEKING (Random data with random seeking)
HIGHLOW (High low frequency data overwrite)
BINARY1 (Binary data pattern 1)
BINARY2 (Binary data pattern 2)
ZEROS (Zeros data pattern)
ONES (Ones data pattern)
RANDOM (Random data pattern)

FILE (File Size): Value in % of disk space
BLOCK (Block size): Value in KB
SEEK (Seek count): Value
DUTY (Duty cycle override): Value in %

MAXCYCLES (Max test cycles to run disk test for): Value (eg 10)

The following parameters are global (i.e. not for a particular disk):
ALL (Select all hard drives): [YES | NO]

Examples:

```
SETDISK DISK /mnt/disk1 MODE binary1 FILE 2 BLOCK 512
SETDISK ALL yes MODE binary1 FILE 2 BLOCK 512
```

SETDISKDESTRUCTIVE COMMAND

Overview:

Set the disk test options for a RAW (destructive) disk test. The first parameter for the command is always disk name (physical device name).
This test will destroy any information already saved on the disk.

Parameters:

See above

Examples:

```
SETDISKDESTRUCTIVE DISK /dev/sda MODE binary1 FILE 2 BLOCK 512
```

SETSERIALPORTS COMMAND

Overview:

Set the serial test options and port names

Syntax:

```
SETSERIALPORTS PORT <parameter> MODE <parameter>
SETSERIALPORTS DISABLERTS <parameter> DISABLEDSR <parameter> TIMEOUT <parameter>
SPEED <parameter>
```

Parameters:

PORT: The name of the serial port eg ttyS0
MODE: [DETECT | LOOP | NONE] DETECT - detect only, LOOP - detect and loop back, NONE - remove port from test list

The following are global COM port test parameters (i.e. not specific to a COM port):

DISABLERTS (Disable RTS/CTS test phase): [YES | NO]
DISABLEDSR (Disable DSR/DTR test phase): [YES | NO]
TIMEOUT (Send and recv timeout): Value in ms
PORT_SPEED (Kb/s): [110 | 300 | 600 | 1200 | 2400 | 4800 | 9600 | 14400 | 19200 | 38400 | 56000 | 57600 | 115200 | CYCLE115 | 128000 | 256000 | CYCLE230 | 921600]

Examples:

```
SETSERIALPORTS PORT /dev/ttyS0 MODE LOOP
SETSERIALPORTS DISABLERTS no DISABLEDSR no TIMEOUT 200 SPEED 19200
```

SLEEP COMMAND

Syntax:

```
SLEEP <Delay period>
```

Parameters:

< Delay period > An integer that represents the number of milliseconds to pause before continuing with the next command in the script..

Examples:

```
#Pause 2 seconds
SLEEP 2000
```

RUN COMMAND

Syntax:

```
RUN <Test Name>
```

Parameters:

<Test Name> See below for a list of all test names.

Examples:

```
#Run the CD test with the current settings
RUN CD
#Run all the tests in the current configuration simultaneously
RUN CONFIG
```

<Test Name> Parameter

The test name parameter can takes the following values. The first value "CONFIG" is special because it does not refer to the name of an individual test. When used with the RUN command it causes all tests in the current configuration file to be started simultaneously.

```
CONFIG
MATHS
CD
DISK
MEMORY
NETWORK
PARALLEL
SERIAL
USB
2D
3D
PCIE
```

Example

```
#Load my preferred test configuration
LOAD "MyConfiguration1.bitcfg"
#Override the test duration for all tests
SETDURATION 60
MESSAGE "Click on OK to start test run"
RUN MATHS
MESSAGE "Insert test discs into both the CD and DVD drive"
RUN CD
#Load my preferred test configuration for disk testing
LOAD "MyDiskConfig.bitcfg"
RUN CONFIG
```

12 Advanced options

- Maintaining multiple test configurations
- Report Information (Machine ID and Notes)
- Command Line Parameters
- Preparing a Customer Test Certificate

12.1 Maintaining Multiple Test Configurations

If you are in the business to testing many different computers then you may benefit from maintaining different test configuration. By saving a particular configuration to a file, you can avoid resetting all the parameters each time BurnInTest is used with a different computer set-up.

A configuration file contains all the parameters available in the test duty cycles and test preferences windows. The saving and loading of configuration files can be done from the menu options, 'Load Config' and 'Save Config As'. Configuration files can also be used from the command line.

Configuration files normally end in the extension 'cfg'. For example the current configuration is stored automatically in the file 'newcurrentbit.cfg'

Using this feature it is possible to make your own configuration for each computer set-up that your business encounters / builds. eg. The heavy load situation, the Disk test only configuration.

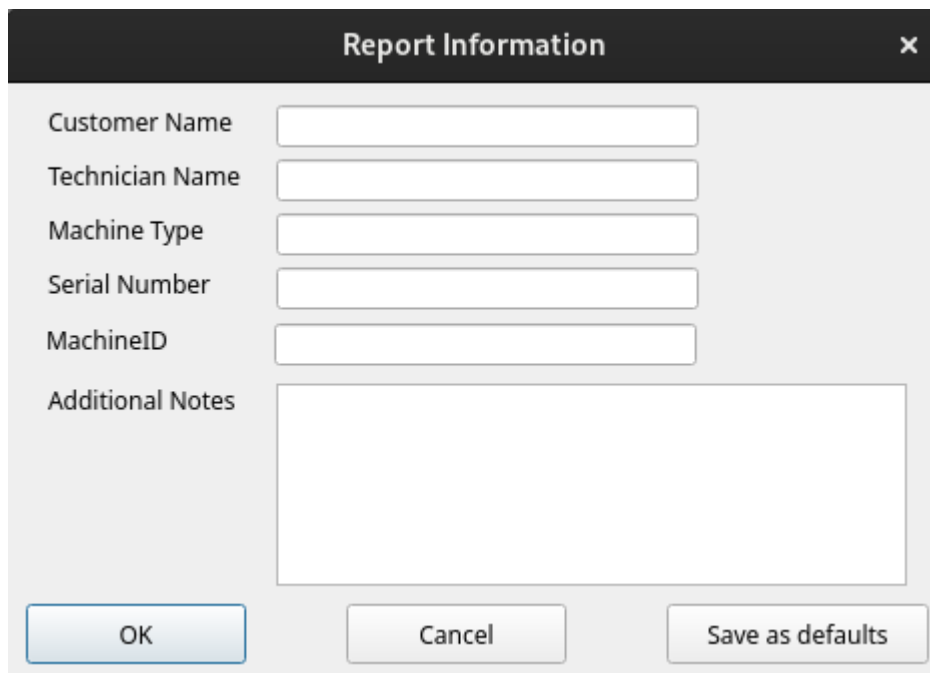
After the configuration files have been created they can be transferred between computers where BurnInTest is installed

Compatibility issues

The configuration files are not compatible between the Windows version and the Linux version.

12.2 Report Information (Machine ID and Notes)

The fields in this section allow the entry of customer specific information for the system under test. This information is used in the Customer Test Certificate report. See Preparing a Customer Test Certificate for more information about configuring a test certificate for your company.

A screenshot of a 'Report Information' dialog box. The dialog has a dark title bar with the text 'Report Information' and a close button (X). The main area is light gray and contains several input fields: 'Customer Name', 'Technician Name', 'Machine Type', 'Serial Number', and 'MachineID', each with a white text box. Below these is a larger white text box for 'Additional Notes'. At the bottom, there are three buttons: 'OK' (highlighted with a blue border), 'Cancel', and 'Save as defaults'.

The information entered into this window will be displayed in saved result files, in the main window and on printed results. You can display this window automatically at startup by running BurnInTest with the command line parameter "-M".

Customer Name

The name of the Customer you are testing the system for.

Technician Name

The name of the tester.

Machine Type

Typically the make and model of the machine would be entered in this field, e.g. Dell Dimension T800 XPS

Serial Number

Typically the serial number would be entered here, e.g. D345-789-YT99

Additional Notes

This is a spare text field that users may make use of to save any additional information that needs to be associated with the machine being tested or the organization doing the testing. No more than 300 characters may be entered into this field.

Save As Defaults

This button will save the contents of these three fields to disk. Each time BurnInTest starts the fields will automatically be reloaded. This can save retyping the same information over and over again.

12.3 Command Line Parameters

-C [configname]

BurnInTest will load the config file specified.

-D [duration]

Sets the test duration to the value specified in minutes

-X [duration]

Sets the test duration to the value specified in cycles

-d

Enables debug mode which will create a debug.log file in the same directory as the BurnInTest executable

-S [script]

Load the script file specified, the script will start automatically when using the option.

-R [milliseconds]

Automatically start starts test after specified value in milliseconds

-v

Print version information and exit

-J

Cycle Disk test patterns between test files (when cyclic set). This option has been added to allow multiple test patterns to be used across very large disks, without waiting for the disk to be completely tested with one pattern before moving on to the next disk test pattern.

-K

Keep the disk test data files on disk when an error is logged. Best used in conjunction with the "Stop on Error" option. This setting will also save the random data blocks used if running the "Random Data" and "Random Data and Seeking" disk tests.

-b [blocksize]

Set disk test block size as KB (eg -b 16 is a 16384 byte block size).

-M

Automatically display the Machine ID Window when BurnInTest is started. This can be useful in a production line scenario to allow the tester to enter test specific information in a more automated fashion.

-MGT [MachineID]

Set the Configuration->Report information->MachineID value that is used as the main identifier in the Management console. This value will be used even if another MachineID has been saved as the default MachineID. [Machine ID] can be fixed text or a Linux environment variable that has been set.

This ID needs to be a minimum of 8 characters long and can be a maximum of 15 characters. Characters should be alpha numeric and special characters avoided, invalid characters such as "< > & ' " " will be removed from the ID.

For example:

```
bit.exe -MGT MYMACHINEID
```

```
bit.exe -MGT $MAC_ADDRESS
```

12.4 Preparing a Customer Test Certificate

A Test Certificate is a report targeted at your customer. It includes your company details, Technician information, System summary, Result summary and a Certification statement. The report would typically be created after a test run. It is produced by (i) selecting File -> Save results report menu and the Test Certificate option or (ii) selecting Configuration -> Test Preferences and selecting a Report type of Customer certificate in Logging.

The Test Certificate is a HTML template. This HTML template will need to be modified with your company's details.

The items in the report that are specific to your customer, and likely to change frequently, can be modified from Configuration -> Report Information. This is the Customer's Name and Technician's name.

The items in the report that are specific to your company, and unlikely to change frequently, can be modified in the HTML template.

A description of the HTML template follows.

Test certificate HTML template

The default location of the HTML test certificate template is the BurnInTest directory. It must be named BITCertificateTemplate.html. This file may be modified in the BurnInTest directory. A sample BITCertificateTemplate.html is provided in the installation package.

Style

The HTML file defines the Styles for the Report in the <style> block. You can change the style, such as font size and colors by modifying the style block.

Company Logo

The first item in the <body> of the HTML file is a <table> containing your company name and logo as a gif file.

For example:

```
<p>Put your company logo here:</p>

```

You should modify the text in the first line to be your Company name, and change the logo gif filename to a file containing your company logo.

BurnInTest generated report components

BurnInTest will automatically insert 3 sections where the following marker is placed in the HTML file,

```
<!--BITCERTIFICATE-->          <!--Do not edit this line as it will be replaced by the results
report-->
```

The sections are:

BurnInTest Certificate: Includes Customer name, date, technician name and BurnInTest version.

System summary: Includes summary information about the CPU,OS, RAM, video card and drives.

Result summary: Includes test start and stop times, duration, and a list of tests run with a PASS or FAIL indication.

Certification statement

The certification heading and title can be modified in the HTML file by editing the following sections:

```
<h2>Certification</h2>
<p>This document certifies that the Tests described above have been carried out by a
suitably qualified technician on the System described </p>
```

12.5 Management Console

For BurnInTest users testing large numbers of systems, the BurnInTest Management Console Web server application allows system information, test status and test result information to be managed centrally for BurnInTest tests.

Use of the Management Console requires the Web server application to be separately obtained from PassMark Software and installed on a Web server.

Via a browser the BurnInTest Management Console allows information about BurnInTest tests to be displayed. This includes the status of currently running tests, system information, current and previous test results, test reports and test statistics.

Please refer to the BurnInTest Management Console documentation for further information.

13 Command Line Version

The command line version of BurnInTest does not have a Graphical User Interface and as such does not require any graphical libraries to be installed but require ncurses to display information. This also means it is missing some of the tests from the GUI version, the 2D and 3D tests, and has a text based config file so it can be edited in a text editor. The 64 bit version (bit_cmd_linex64) included in the download.

Use the number keys to navigate to the different test information pages while the tests are running and use Esc to exit BurnInTest.

Notes:

Due to the limited amount of available space on the network test screen the values for the send and received packets are abbreviated to display as K (thousand), M (million) and B (billion).

Text Based Configuration File

The text based configuration file (cmdline_config.txt) can be edited in any text editor. The “#” symbol is used to comment out lines and any lines starting with a “#” will be ignored. Each section of the config file begins with a <SectionName> tag and the valid sections are;

- <Logging>
- <Notes>
- <PreTest>
- <PostTest>
- <DutyCycles>
- <Memory>
- <Disk>
- <CPU>
- <USB>
- <Network>
- <CDRWDVD>
- <Serial>
- <Parallel> (x86 only)
- <Sound>
- <PCIE> (x86 only)
- <Management>
- <Battery>
- <Temperature>
- <Plugin>
- <Errors>

For sections that can have multiple tests (eg Disk, Serial) a <Test> tag is used to add a new device and </Test> used to signify the close of that particular section. The default config file is commented with expected values for each section.

Logging Section

Flag	Meaning	Valid Keywords / Range
LogFilename	Location (path) of log file	Valid path to place log file
LogPrefix	String to prefix to log name	Text string

Format	Format of log	Ascii, HTML or certificate
LogDetailLevel	Detail level of information logged	summary, normal
TraceLevel	Trace level of information logged	0, 1, 2
MaxFileLines	Max file lines per log	100 - 10000
LogMode	Logging mode, single/multiple files	ClearAppend, AccumulateAppend, ClearNew, Timestamp

Notes Section

Flag	Meaning	Valid Keywords / Range
MachineType	One line description of the machine type	Text string
MachineSerial	Custom serial No. for this machine	Text string
Notes	One line description of extra notes	Text string

PreTest Section

Flag	Meaning	Valid Keywords / Range
Action	Action before tests start	None or ExternalWait
ExternalPath	If using ExternalWait this is the path of the file or script to launch and wait for the end of it's execution	Valid path to external script or executable to launch and wait for it to complete

PostTest Section

Flag	Meaning	Valid Keywords / Range
AutoStopPassed	Auto stop action when tests have passed	Stop, RunApp or Reboot
ManualStopPassed	Manual stop action when tests have passed	Stop or RunApp
ExternalPathPassed	If using AutoStopPassed RunApp this is the path of the file or script to launch before exiting	Valid path to external script or executable
ExternalParamsPassed	If using AutoStopPassed RunApp this is the parameters to pass to the application being launched	Text string
UseSystemBeepPassed	Emit a system beep on pass	
DisplayResultsPassed	Display the results window on pass	
AutoStopFailed	Auto stop action when tests have failed	Stop, RunApp or Reboot
ManualStopFailed	Manual stop action when tests have failed	Stop or RunApp
ExternalPathFailed	If using AutoStopFailed RunApp this is the path of the file or script to launch before exiting	Valid path to external script or executable
ExternalParamsFailed	If using AutoStopFailed RunApp this is the parameters to pass to the application being launched	Text string
UseSystemBeepFailed	Emit a system beep on fail	
DisplayResultsFailed	Display the results window on fail	

Duty Cycles Section

Flag	Meaning	Valid Keywords / Range
AutoStopMinutes	Auto stop after specified minutes	0+ (0 for run forever)
AutoStopCycles	Auto stop after specified cycles	0+ (0 for run forever)
CPU	Duty cycle load	1-100
CDDVD	Duty cycle load	1-100

Memory	Duty cycle load	1-100
Serial	Duty cycle load	1-100
Parallel	Duty cycle load	1-100
Disk	Duty cycle load	1-100
Network	Duty cycle load	1-100
USB	Duty cycle load	1-100
Sound	Duty cycle load	1-100
PCIE	Duty cycle load	1-100

Memory Section

Flag	Meaning	Valid Keywords / Range
<Test pattern name>	Pattern setting for test	Cycle, Sequence, Binary1, Binary2, Zeros, Ones

Disk Section

Each partition being tested and it's related parameters should be enclosed in a <Test>...</Test> section. For partitioned drives they must be mounted before BurnInTest can test them, they also need to be of partition type ext2, ext3, ext4, reiserfs, xfs, zfs, vfat or ntfs. Due to limited NTFS support in some Linux distributions you may need to use a non-default type (eg ntfs-3g) to mount NTFS drives with read + write permissions before you can test them.

Flag	Meaning	Valid Keywords / Range
Device	The device or partition to be tested	Valid partition eg /dev/sdb2 or raw un-partitioned device /dev/sdc
TestMode	The test pattern	Cyclic, Sequential, HighLow, Binary1, Binary2, Zeros, Ones, Random, RandomSeeking
FileSize	File size as % of disk space	1-15
DutyCycleOverride	Duty cycle override	0 - 100
SeekCount	Seek count	0-10000

If the config needs to be setup so all disks are tested these flags can be used, they need to be in the <Disk> Section and don't need to be enclosed in a <Test> section like individual partitions. The individual <Test> sections for specific partitions should not be used when using these flags:

Flag	Meaning	Valid Keywords / Range
TestAllDisks	Select all detected mounted partitions for testing	N/A
TestAllRAWDisks	Select all detected RAW (unpartitioned) disks for testing	N/A
FileSize	File size as % of disk space when using the TestAll options	1-15
TestMode	The test pattern when using the TestAll options	Cyclic, Sequential, HighLow, Binary1, Binary2, Zeros, Ones, Random, RandomSeeking
ExcludeDisks	Exclude mount point or device name from "Test all" option	A list of mount points (/) or device names (/dev/sda) seperated with ',' characters

CPU Section

Flag	Meaning	Valid Keywords / Range
General	Enable the general operations test	N/A
FPU	Enable the floating point operations test	N/A
SSE	Enable the SSE instructions test	N/A
SSE2	Enable the SSE2 instructions test	N/A
NEON	Enable the Neon instruction test	N/A
Prime	Enable the prime number test	N/A

Network Section

Flag	Meaning	Valid Keywords / Range
IP	IP to test, one IP flag for each test, up to 6	Valid IP or 127.0.0.1
BadPacketRatio	Ratio between good and bad packets before an error is logged	0-99, set to 0 for every bad packet to generate an error.
TimeOut	Timeout in milliseconds	100 - 200000
TestAllNICs	Bind each IP to a different NIC	N/A

CD-DVD Section

Each CD/DVD device being tested and it's related parameters should be enclosed in a <Test>...</Test> section.

Flag	Meaning	Valid Keywords / Range
Device	Device to test	Valid CD/DVD device
TestMode	Test to run	Music, Data, Passmark, No
Seek	Additional seek (Passmark test mode only)	0-1000

Serial Section

Each serial device being tested and it's test mode should be enclosed in a <Test>...</Test> section:

Flag	Meaning	Valid Keywords / Range
Device	Device to test	ttys0 - ttys63 (can also use ttyUSB)
TestMode	Test mode for device	Loopback, Detect

These parameters are only set once for the Serial section and apply to all serial tests:

DisableRTS	When specified disables RTS/CTS and DSR/DTR	N/A
SendReceive	Send/Receive timeout value in milliseconds	100-3000
BaudRate	Baud rate to set for test	110, 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200, 230400, Cycle115 (cycle to 115200), Cycle230 (cycle to 230400)

Parallel Section (x86 only)

Flag	Meaning	Valid Keywords / Range
Device	Device to test	valid parallel port lp0, lp1 etc
TestMode	Test mode for device	Loopback, Detect

USB Section

Flag	Meaning	Valid Keywords / Range
AutoSelectUSB	Automatically select all USB plugs found on the system for testing	N/A
NumUSB	Max number of USB devices to test	1-10
RunUSBBenchmark	Run benchmark test at the start of the USB test	N/A
USB3Connection	USB 3 connection speed Auto, SuperSpeed (5GB/s), HighSpeed (480Mb/s), FullSpeed (Mb/s), Cycle (Cycle between speeds)	Auto, SuperSpeed, HighSpeed, FullSpeed, Cycle

Sound Section

Flag	Meaning	Valid Keywords / Range
MaxDistortion	Max level of distortion before an error	1 - 20

SoundDevName Sound device name used for testing plughw, default (other sound alias for system, should be left as plughw in most cases)

PCIE Section (x86 only)

Flag	Meaning	Valid Keywords / Range
TestAllPCIE	Automatically select all PCIE slots with PassMark PCIE cards for testing	N/A
NumPCIE	Number of PCIE cards to tests	1-10

Management Section

Flag	Meaning	Valid Keywords / Range
Server	Management console server address	Valid http address for PassMark management console eg http://localhost/htdocs/mgtconsolemsgHandler.php
Username	Username if HTTP authentication is being used on the webserver	
Password	Password if HTTP authentication is being used on the webserver	

Battery Section

Flag	Meaning	Valid Keywords / Range
TestMode	Run test as main or pre-test	Main, Pretest
Charge	Comparison mode	Current, Full
ThresholdAbs	Absolute threshold when an error is logged. Only one of this and ThresholdPer should be defined	1-100000
ThresholdPer	Percentage threshold for when an error is logged.	1-99

Temperature Section

Flag	Meaning	Valid Keywords / Range
AutoSelectSensors	Auto select which temperature sensors to watch	N/A
SamplePeriod	SamplePeriod	5+
UseCelcius	Use celcius for temperature scale, only one of this or UseFahrenheit should be defined	N/A
UseFahrenheit	Use Fahrenheit for temperature scale	N/A
StopTestOnHighTemp	Stop testing if a sensor records a temperature higher than the threshold	N/A
<Sensors>	If not using AutoSelectSensors can add sensor names to look for here, maximum 64, threshold after a ',', eg CPU 0 Core 1,90 must match what BurnInTest displays eg "CPU 0 Core 1".	

Plugin Section

Each plugin being run should be enclosed in a <Test>...</Test> section.

Flag	Meaning	Valid Keywords / Range
Path	File path of plugin to launch	Valid system path to plugin file
PreTest	Run as a pre-test plugin	N/A

These parameters are only set once for the plugin section and apply to all plugins:

RunSeparately	Run each plugin separately	N/A
IgnoreTimeout	Ignore no-operations timeout for plugins	N/A

Error Section

Flag	Meaning	Valid Keywords / Range
ActionOnError	Action to take on error	Continue, Beep, Stop

Command Line Parameters

-B

BurnInTest will run in a silent mode and not create a ncurses/terminal window.

-C [configname]

BurnInTest will load the config file specified.

-x

Don't automatically exit BurnInTest after tests have finished, the default behavior is BurnInTest will exit once the test run has finished.

-k

Register your key stored in the key.dat file. This file needs to be created in the same directory as the BurnInTest executable. When using this flag BurnInTest will expect your username/key, from (and including) the "-----START_OF_KEY-----" to (and including) the "-----END_OF_KEY-----" flags, to be stored in a file called key.dat in the same directory as BurnInTest. You should paste the key as is. It should be 6 separate lines, 1 each for the start and end flags, 1 for the username and 3 for the key. You will need to restart BurnInTest after registration and you can delete the key.dat file you have created. **Do not alter the savedkey.dat file, this is a file maintained by BurnInTest and altering this file will corrupt it.**

-D [duration]

Sets the test duration to the value specified in minutes

-X [duration]

Sets the test duration to the value specified in cycles

-d

Enables debug mode which will create a debug.log file in the same directory as the BurnInTest executable

-p [time]

Sets the update interval (time between display updates) in milliseconds, allowable range 500 (.5 second) to 300,000 (5 minutes).

-J

Cycle Disk test patterns between test files (when cyclic set). This option has been added to allow multiple test patterns to be used across very large disks, without waiting for the disk to be completely tested with one pattern before moving on to the next disk test pattern.

-K

Keep the disk test data files on disk when an error is logged. Best used in conjunction with the "Stop on Error" option. This setting will also save the random data blocks used if running the "Random Data" and "Random Data and Seeking" disk tests.

-v

Print version information and exit

-MGT [MachineID]

Set the Configuration->Report information->MachineID value that is used as the main identifier in the Management console. This value will be used even if another MachineID has been saved as the default MachineID. [Machine ID] can be fixed text or a Linux environment variable that has been set.

This ID needs to be a minimum of 8 characters long and can be a maximum of 15 characters. Characters should be alpha numeric and special characters avoided, invalid characters such as "< > & ' " " will be removed from the ID.

For example:

bit.exe -MGT MYMACHINEID

bit.exe -MGT \$MAC_ADDRESS

14 Other Useful Information

- Contacting PassMark Software
- System Requirements
- Example ASCII Log file
- Precautions for thorough and careful testing
- Fault finding
- Common Error Messages
- USB Loopback Plugs
- Making Loopback plugs
- Serial port loopback plug
- Parallel port loop back plug

14.1 Contacting PassMark Software

On the Web

You can contact PassMark on the web at:

https://www.passmark.com/about/contact_us.php

E-Mail

For technical support questions, suggestions:

help@passmark.com

For sales & commercial issues

sales@passmark.com

To order online, visit this page:

<https://www.passmark.com/sales/index.php>

For any other issues

info@passmark.com

FAQ

You may also want to check out the list of Frequently Asked Questions.

<https://www.passmark.com/support/index.php>

14.2 System Requirements

For the list of supported Linux distributions please refer to this page:

https://www.passmark.com/products/burnintest/bitlinux_download.php

14.3 Example ASCII Log file

ASCII (Text) log files can be created using the logging option in the Test preferences dialog. The ASCII text log file contains all the information that is in the main test window plus some additional information such as the amount of RAM in the system and cache size for the CPU(s).

```
PassMark BurnInTest Log file - http://www.passmark.com
=====
```

```
BurnInTest v5.0 (1005) Linux 64bit
Logging detail level: Normal
```

```
*****
```

SYSTEM SUMMARY

```
*****
```

```
Linux 6.2.9-300.fc38.x86_64
4 x AMD Ryzen 9 3900 12-Core Processor
0 MB RAM
```

GENERAL

```
System Name: localhost-live
Motherboard Manufacturer: Intel Corporation
Motherboard Model: 440BX Desktop Reference Platform
Motherboard Version: None
Motherboard Serial: None
Bios Manufacturer: Phoenix Technologies LTD
Bios Version: 6.00
Bios Release Date: 11/12/2020
System Serial: VMware-56 4d 6d 5c c8 18 2a 91-60 3a 44 c8 61 c9 a3 d8
```

CPU

```
CPU Manufacturer: AuthenticAMD
CPU Type: AMD Ryzen 9 3900 12-Core Processor
Speed: 3100.0 MHz
CPUID: Family 23, Model 113, Stepping 0
Physical CPU's: 4
Cores per CPU: 1
Hyperthreading: Not capable
CPU Features: MMX SSE SSE2 SSE3 SSE4a SSE4.1 SSE4.2 PAE AES AVX AVX2
Cache per CPU package:
L1 Cache: 256 kB
L2 Cache: 2 kB
L3 Cache: 64 kB
```

MEMORY

```
Total Memory: 7927 MB
Available Memory: 2765 MB
Memory Devices:
RAM slot #0 : Not populated
```

GRAPHICS

```
VMware SVGA II Adapter (prog-if 00 [VGA controller])
Memory: 128 MB
```

DISK DRIVES

```
/dev/nvme0n1 Size: 20.0 GB
    nvme0n1p1
    nvme0n1p2 (/boot, ext4, 973 MB)
    nvme0n1p3 (/home, btrfs, 19.0 GB)
```

NETWORK:

```
ens160 VMware VMXNET3 Ethernet Controller (Speed: 10000Mb/s) (MAC: 00:0c:29:c9:a3:d8) (IPv4: 192.168.2.236) (IPv6: fe80::95c9:a4da:947c:145c)
```

PORTS:

```
Keyboard Port: Circular DIN-8 male
Keyboard Port: Circular DIN-8 male
```

USB HOST CONTROLLERS AND DEVICES:

```
VMware USB1.1 UHCI Controller (prog-if 00 [UHCI])
VMware USB2 EHCI Controller (prog-if 20 [EHCI])
VMware USB3 xHCI 1.0 Controller (prog-if 30 [XHCI])
VMware VMware Virtual USB Mouse
VMware Virtual Bluetooth Adapter
```

DISK CONTROLLERS:

```
Intel Corporation 82371AB/EB/MB PIIX4 IDE
```

```
*****
```

DETAILED EVENT LOG

```
*****
```

```
LOG NOTE: 2023-06-20 14:03:15, Status, Starting test run
LOG NOTE: 2023-06-20 14:03:16, Status, Completed started test run
LOG NOTE: 2023-06-20 14:04:18, Status, Test run stopped
```

```
*****
```

RESULT SUMMARY

```
*****
```

```
Test Start time: Tue Jun 20 14:03:15 2023
Test Stop time: Tue Jun 20 14:04:17 2023
Test Duration: 000h 01m 02s
```

	Test Name	Cycles	Operations	Result	Errors	Last Error
	CPU - Maths	80	193 Billion	PASS	0	No errors
	Memory (RAM)	1	7.219 Billion	PASS	0	No errors
	2D Graphics	6	1.305 Million	PASS	0	No errors
Network: ens160 (127.0.0.1)		1	10208	PASS	0	No errors

TEST RUN PASSED

14.4 Precautions for thorough and careful testing

For a hardware test to be useful several precautions need to be taken. Failure to take into account these factors may result in tests being misleading or other unwanted results.

- Stop all other applications before running BurnInTest. BurnInTest can be run in the background but it just doesn't seem prudent to do any important work when you're testing a computer to see if it will fail. In any case BurnInTest will place such a load on the system that any other applications will run at a snails pace. Not having other applications running also frees up more RAM that can be used by the Memory test.
- Back-up any important files before you start. BurnInTest can simulate many days of typical office PC use in a few hours, this increases the risk of hardware failure. Note that the testing process itself doesn't touch any existing files on the hard disk or floppy disk.
- When testing multiple disk drives at the same time you may not want to test multiple partitions that are on the same physical drive at the same time. Doing this can result in an enormous amount of seeking between partitions and not as much reading and writing.
- PassMark recommends running BurnInTest just after you install a PC for the first time, as this is the ideal time to find a problem. The PC will be still under warranty and you can't lose any of your data (because you haven't loaded any). Any disruption caused by a failure will be minimal.

- Remember that BurnInTest does not create problems in your hardware, it just helps you find them in a controlled manner. BurnInTest doesn't use any nasty programming tricks to try and make your hardware fail. It uses the same functions and procedures that standard Linux applications and file servers use. If your computer fails when running BurnInTest, it was going to fail in the near future anyway !
- If you only want to test a particular component of the computer, turn the other tests off. There's no point using the CD-Drive when you only wanted to test the new hard drive.
- Doing a successful test run doesn't mean that the computer will never fail. Software problems, viruses, and the fact that no computer component has an unlimited life span means that precautions need to be taken. Having good BurnInTest results is NOT a substitute for making good file backups in the future.
- Because BurnInTest doesn't delete any of the existing files from a disk, this occupied portion of the disk will not be tested. Thus the more free space that the disk has before the test, the larger the portion of the disk that will be tested.
- When using the CD test with a music CD verify that the music is being played clearly through the PC's sound system.
- On some new computers, the warranty may be voided if you open up the case. Check your warranty before you start poking around in the case.
- Old computers tend to fill up with dust over the period of many years. This dust layer can cause heat build up and even short circuits. Check for dust build-up in old computers before you start.
- Check the computer has adequate ventilation & check all the fans are in good working order.
- Check that the computer isn't full of bugs. (i.e. the insect type). Depending on where you live, insects can be a real problem. The term 'Computer Bug' was coined after a dead moth was found to have shorted out one of the first computers built. In Australia, cockroaches are the most common cause of failure in microwave ovens. Recent studies have even suggested that some insects are attracted to electro-magnetic fields. So watch out !
- There are many cases where a software bug may appear to be a problem with the hardware. Knowing who or what to blame isn't easy. Check with your hardware manufacturer(s) from time to time in case they have released new software that fixes some problems they may have found. The hardware components that in general have the most problems with their 'Driver' software are, Video cards (especially the 3D functions), Sound cards and CDs.
- By using the Network test you can test both your computer and the network it is connected to. If an error does occur it may be difficult to determine the location of the error. If you are using an Internet address then it is very likely that any transient errors are the result of problems on the Internet. The best compromise is probably to set the test address to the address of a machine on your local area network, (if you have one).
- When selecting a Serial or Parallel port to use for loop back testing, ensure that the port selected is not already in use by the system. (e.g. by a mouse or printer).
- Because of limitations in the memory test, faulty RAM errors may not be picked up by the test and faulty RAM can often manifest itself in different ways. These include disk I/O errors, system crashes and lockups.

Not all faults can be detected via software and a machine can not be deemed to be 100% fault free after testing with BurnInTest. Some reasons for this include;

- Faults may develop only after a certain period of time
- Faults may manifest themselves only in certain environmental conditions (e.g. excessive heat)

- Faults might only manifest themselves in an unusual or specific set of circumstances out of those which are tested
- Not all modes of operation for all possible hardware devices are tested
- Faults might only appear with certain combinations or versions of software / device drivers

BurnInTest is not fault-tolerant and is not designed or intended for use or resale as on-line control equipment in hazardous environments requiring fail-safe performance, such as in the operation of nuclear facilities, aircraft navigation or communication systems, air traffic control, direct life support machines, or weapons systems, in which failure of BurnInTest could lead directly to death, personal injury, or severe physical or environmental damage ("High Risk Activities"). PassMark and its suppliers specifically disclaim any express or implied warranty of fitness for High Risk Activities.

14.5 Fault finding

What follows are some hints on how to go about finding the cause of particular system instability. We don't want to try and explain the steps involved in each of these processes, they are just points that may warrant future investigation.

- Check you don't have any viruses.
- Check the drive for errors using the system's disk checking tool or with command line tools like 'fsck' or 'badblocks'.
- Check that space is available on the disk for the swap files.
- Have a look through the issues in the section, 'Precautions for thorough and careful testing'.
- Don't run all the BurnInTest tests at once. Run just the 2D graphics, then run just the 3D graphics, then just the disk, etc.. This will allow the problem to be isolated to one area.
- If you suspect hardware, and you know what you're doing, pull out all the "optional hardware", eg LAN cards, I/O cards and see if the system is more stable.
- Once again, if you know what you're doing, start swapping out components of the system to see if the fault can be localized. Obviously you'll need some spare hardware to do this.
- If you suspect Open GL problems, possibly with bad Open GL installation or missing libraries, consider running your system's OpenGL's screensaver or any OpenGL applications or games to see if problem persists.
- Faulty RAM may not always be detected by the memory test. It may manifest itself as a disk fault or system crash.
- If you're really stuck you may want to try a reinstallation of your operating system on a reformatted disk. Think carefully about this option before you attempt it, there are lots of good reasons why you don't want to reformat your hard disk.
- Make sure you've got the most up to date software drivers for your hardware. Drivers are a never-ending source of problems.
- Check that you haven't ended up with an over clocked CPU and don't know about it.
- Check that you haven't purchased the cheapest and nastiest hardware in the hope of saving a couple of dollars (or pounds, francs, etc). Often it may not be the cheap hardware that causes problems but the quality and support of the software drivers that comes with the hardware that are a problem. Don't shop on price alone, check out the support and product reviews.

14.6 Common Error Messages

When BurnInTest encounters an error during a test run a short description of the error is displayed in the main window. What follows is an explanation of the common errors that may be encountered.

Incorrect mathematical addition / subtraction / division / multiply

The execution of a mathematical operation came up with the wrong result (e.g. $1 + 1 = 3$). This is a very serious error for a computer as it means the computer is incapable of executing the same sequence of instructions to get the same result. Possible reasons are faulty RAM, System bus, CPU or overheating. There is a strong chance that your computer will crash or lock up just after an error like this because if the computer can't add two numbers correctly, there is a good chance that it can't continue to run a program either.

No free memory for buffer

Operating system does not have enough free memory to allow the allocation of a buffer.

Not enough free disk space

There is not enough free space on the disk to create a test file for the Disk tests.

Test file could not be created

BurnInTest was unable to create a file on the disk in order for it to be able to run the disk tests. Check permission access to the disk. Check file write and create permissions in the root directory of the disk under test.

Error while writing to the disk

The test file could not be completely written to the disk. This could indicate a hardware error or a sudden lack of free disk space

Error writing to disk - Disk is full

The test file could not be completely written to disk because the selected hard drive is full.

Test file could not be re-opened

The test file was created successfully, but could not be reopened for verification. This could indicate a hardware error or a resource problem within the operating system.

Error while reading from the disk

The test file created on the disk can no longer be read. This could indicate a hardware error or a problem within the operating system.

Data Verify failure

This is a serious error that indicates that the data read from the disk is NOT the same as the data that was written to the disk. This could indicate a hardware error.

Disk is full or user file system limit reached

A test file could not be created on the disk being tested because the disk is full or the capacity of the file system has been exhausted.

Could not set CD Time format

Could not get number of audio tracks

Could not play the audio track

CD-ROM failed to return status information

Cannot detect audio information on CD

No audio media in CD-ROM

These errors relate to one of the following problems.

There is not a CD in the first CD-ROM connected to the computer.

The CD in the CD-ROM is not a Music CD.
The CD is already in use by another application.
The CD has an error on it and is unreadable.
The CD is dirty, dusty or scratched and can not be read
The CD Software drivers are not correctly installed.
The CD-ROM has a hardware error.

Checksum failure for CD/DVD file

This indicates that the file read from the data CD/DVD being tested failed the checksum verification. This means that the CD/DVD Drive is unable to accurately read data from the CD/DVD being tested.

Could not open file on CD/DVD for reading

The CD/DVD drive selected could not be opened for reading as a data CD/DVD.

Error while reading file from CD/DVD

A full block of data could not be read from the data CD/DVD.

Error while searching for files on data CD/DVD

An error occurred while searching for files on a data CD/DVD. This can be the result of a corrupted (Error code 1117) or blank CD (Error code 21).

Data read from CD/DVD was incorrect

A block read from a specialized PassMark Test CD or DVD was incorrect. There was at least one byte in the block that was not the value expected. In the detailed error log there is additional information that give the number of bytes in error and displays the expected value and the value actual read from the disc. It may be that the drive is faulty but you should check that the disc is not scratched, dusty or damaged before assuming a hardware fault.

Could not determine type of Test CD/DVD

BurnInTest was expecting to find a specialized test CD/DVD in the drive selected. A specialized test CD/DVD has a specific set of files, which all have a specific file size. These disc are normally purchased from PassMark Software or made using the PassMark CD-Maker utility. BurnInTest was not able to find the correct files or the files appear to have the wrong file size. If you don't have a specialized test CD/DVD available, select one of the other two test options in the preferences window.

OpenGL is not installed or is not installed correctly

OpenGL is not installed or is not installed correctly on your system.

Error verifying data in RAM

The data written to memory is not the same as the information read from memory. This is a very serious error, much like the "Incorrect mathematical...", error above. It's highly likely that your computer is about to crash or lock up.

Error allocating RAM

The operating system was not able to allocate the amount of RAM requested by BurnInTest for the memory test. As the RAM must be allocated in a continuous block, this error can sometimes be seen as a result of free memory fragmentation.

Error connecting to network

Could not create a network socket

Could not set socket timeouts

During the establishment of a network connection an error was encountered. The Network address selected has no effect on if these errors occur. They can be the result of the following problems:

- There is no network connection configured for the computer
- The computer is physically disconnected from the network
- The Internet TCP/IP protocol is not installed on this computer

Could not allocate memory for packets

Operating system is low on resources and cannot allocate any more memory.

Could not resolve host name, check settings

The Network address selected does not seem to be correct. Try another address, or using a TCP/IP address directly. The address 127.0.0.1 is good for testing as it is an internal loop back.

Timeout sending packetTimeout waiting for packetError sending packetError receiving packetBad packet

These errors relate to one of the following problems.

The Host using the network address selected doesn't reply to 'ping' messages.

Try a different host.

There is a configuration problem in your network connection.

Your network card may be faulty.

The network itself is not reliably. The information send was not the same as the information echoed by the remote host.

The network is congested or faulty and the packets are not being echoed within the timeout period specified in the preferences window.

Network test alarm. Error ratio exceeded

The Bad Packet ratio specified in the Network test preferences has been exceeded.

Got someone else's packet

This is not really an error. It's more of an information message. Don't worry about this message.

Bad packet. Checksum incorrect

The checksum in the echoed, incoming data packet is not correct. This indicates data corruption or a fault on the remote machine. Note that this checksum is calculated by the remote machine.

Bad packet. Corrupt data

The contents of the incoming data packet are not correct. Normally this error would not been seen as the checksum should detect the incorrect data before this error occurs.

COM port is already in use by another program

The serial port selected for the test is already is use by the operating system. This may be for the mouse, a modem or another serial device.

The requested COM port could not be found

The serial port selected for the test does not exist in this computer. This could happen if COM4 (ttyS3) is selected but the computer has only 2 serial ports, COM1 (ttyS0) and COM2 (ttyS1).

Error while opening COM port

The operating system has reported an error while trying to open the serial port selected for the test. This could be a configuration problem in the operating system. This error should not normally been seen.

Error getting current COM port configuration

Error while setting new COM port configuration

The operating system has reported an error while trying to configure the selected serial port. The most common cause for this error would be the selection of a speed that is not supported by the serial port chips installed in the computer (the UART). Most chips only support speeds up to 115Kbit/s.

Corruption. Data received didn't match data sent

BurnInTest has detected that the data received from the serial port doesn't match the data sent. This could indicate that there is a hardware problem. This type of data corruption would however be a fairly rare type of event. The more common result of a hardware failure would be the total inability to send or receive data. (see below).

Error while setting current COM port timeouts

The operating system has reported an error while trying to set the timeout periods for the data transmission and reception. This could be a configuration problem in the operating system. This error should not normally be seen.

Error while sending data to the COM portError while receiving from the COM port

The operating system has reported an error while trying to send / receive data through the serial port. If a device (such as a loop back plug) is not connected to the serial port then no data can be sent. If the loop back plug is connected and is not faulty, then this error may indicate a hardware fault.

COM port Clear To Send (CTS) line stuck highCOM port Clear To Send (CTS) line stuck lowCOM port Data Set Ready (DSR) line stuck highCOM port Data Set Ready (DSR) line stuck low

The signal pin test phase of the serial test has failed. This might be because of an incorrectly wired up loop back plug, a cabling problem between the serial socket and the motherboard, a non standard COM port or a problem with the configuration for the COM port. A failure of the CTS pin may be caused by the associated RTS pin, to which it is looped. A failure of the DSR pin may be caused by the associated DTR pin, to which it is looped.

Parallel device driver could not detect port

This error usually occurs if the parallel port test attempts to access a parallel port, which doesn't exist (such as perhaps /dev/lp2 or /dev/lp3). It can also happen if the BIOS settings for the port are not correct. Note that the old 'bi-directional' BIOS mode is not supported. ECP or EPP mode is required.

Could not open parallel device driver

This error usually occurs if the parallel port test fails to access a parallel port, which does exist.

Corruption. Data received didn't match data sent

The data sent to the Parallel port was not the same as the data received. This may indicate a hardware problem or a missing or faulty loop back connector.

Error while sending data to the parallel portError while receiving data to the parallel port

The operating system has reported an error while trying send or receive data. This could be a configuration problem for the parallel port. This error should not normally be seen.

Could not detect the parallel port selected

BurnInTest was unable to find a parallel port at the location selected. Try picking another port and see the Parallel port test description (Chapter 3) for more details about port selection

File system not mounted

BurnInTest Linux does not perform mounting of drives/partitions. To run Disk Test on a drive/partition, it needs to be mounted. To find out what devices are mounted, use "df -ahPT" on the command line. A sample output is shown below:

```
[passmark@localhost src]$ df -ahPT
Filesystem      Type      Size  Used Avail Use% Mounted on
/dev/sdb2       ext3      20G   7.8G   11G   42% /
none           proc      0      0      0    -  /proc
none           sysfs     0      0      0    -  /sys
none           devpts    0      0      0    -  /dev/pts
usbfs          usbfs     0      0      0    -  /proc/bus/usb
none          tmpfs     506M    0   506M    0% /dev/shm
none          binfmt_misc 0      0      0    -  /proc/sys/fs/binfmt_misc
sunrpc         rpc_pipefs 0      0      0    -  /var/lib/nfs/rpc_pipefs
/dev/hdc       iso9660   701M   701M    0 100% /media/cdrom1
```

Permission error writing to disk

Disk Test attempts to write to and read from the disk's partitions that were configured to run. Hence, we need to have read/write access to these devices' partitions. From the command line, use "cat /etc/mtab" to see what permission you have on these devices. A sample output is shown below:

```
[passmark@localhost src]$ cat /etc/mtab
/dev/sdb2 / ext3 rw 0 0
none /proc proc rw 0 0
none /sys sysfs rw 0 0
none /dev/pts devpts rw,gid=5,mode=620 0 0
usbfs /proc/bus/usb usbfs rw 0 0
none /dev/shm tmpfs rw 0 0
none /proc/sys/fs/binfmt_misc binfmt_misc rw 0 0
sunrpc /var/lib/nfs/rpc_pipefs rpc_pipefs rw 0 0
/dev/hdc /media/cdrom1 iso9660 ro,nosuid,nodev,fscontext=system_u:object_r:removable_t,user=passmark 0 0
/dev/fd0 /media/floppy vfat ro,nosuid,nodev,fscontext=system_u:object_r:removable_t,user=passmark 0 0
```

It means the device /dev/sdb2 is mounted at "/" with "rw" (read/write permission) and is of file system type ext3. This appear to be our start-up disk and will appear as "Hard Disc (sdb2) [/]" in Disk Test Preferences. /dev/fd0 is mounted at /media/floppy with "ro" (read only permission) and is of file system type "vfat". This appear to be our floppy disk and will appear as "Media (floppy)" in Disk Test Preferences. Hence, if we are trying to run disk test on "Media (floppy)" in this case, we will get this error.

root(admin) access needed to run this test

For some tests, BurnInTest needs to be run as a super-user (i.e.system administrator or root user). To log in as a super-user, when the login-manager prompts you for a user name and a password during start-up, you will need to use the user name "root", followed by an administrator password. The administrator password is set-up during the installation of the operating system.

Alternatively, even if you are logged in as a non-root user from the login-manager, you can still run BurnInTest as a root user process by typing "su" (stands for super-user) at the command line from the terminal shell. You will then be prompted to enter your root password. Type "echo \$UID" to see what your user ID is. If it shows 0, it means you are "logged in" as root user in THAT SHELL. If it does not show 0, it means your password is incorrect. Check for "Caps Lock" and "Num Lock" as password is case sensitive. If you are "logged in" as root user in THAT SHELL, you can then run BurnInTest from this shell as a root user.

These tests are Serial Port test, Parallel Port test, Memory (RAM) test and Network test.

14.7 USB Loopback Plugs

USB 3.0 loopback and benchmarking plug

USB 3.0 Loopback plugs are one of the fastest methods to diagnose, troubleshoot and load test your PC's USB functions.

When used with BurnInTest is it possible to:

- Quickly check if a USB port is powered;
- Measure the voltage of the USB port, no driver is required for this as the plug will display this information directly on the LCD screen (accurate to $\pm 0.1V$)
- Check that data can be sent and received from the port;
- Verify that the system remains stable under long periods of load;
- Test self powered USB hubs and PC ports function correctly;
- Check USB cabling;
- Concurrently check multiple USB ports at the same time;
- Check if your PC ports are super speed 5Gbits/sec (USB 3.0), high speed 480Mbits/sec (USB 2.0) or full speed 12Mbits/sec (USB 1.x)

USB 2.0 loopback and benchmarking plug

USB 2 Loopback plugs can be used to help diagnose, troubleshoot and load test your PC's USB 1.x and 2.0 ports. Five status LEDs on the USB 2 Loopback plug will dramatically improve the troubleshooting and diagnostic ability of computer technicians and system integrators, who have previously been working blind. The Red, Green and Yellow LED's on the front of this USB adapter plug provide a quick visual indication of I/O activity and if the USB bus is powered. The Orange LED will provide a quick indication whether the PC port is currently supporting the higher USB 2.0 speed (up to 480Mb/s) or the lower speed found with USB 1.x (of up to 12 Mb/s). The leftmost red LED indicates whether recoverable bus error (retransmission of data) has occurred.

These plugs are unique! They are the only true USB 2.0 loop back adapters on the market and have been especially designed for use with our BurnInTest software package. They do not require an external power supply and do not require a connection to any other port on a PC. Each plug also has its own serial number stored in EPROM, on the plug, so it is possible to identify each plug when multiple plugs are connected.

Each plug also contains its own CPU and executes its own firmware (which can be upgraded in the future by download)

Technical Specifications

USB standard:	2.0 HighSpeed and FullSpeed (and is backwardly compatible with USB 1.1 and 1.0)
Plug and play:	Compliant
Physical connector:	Standard 4 wire USB (2 x Balanced Signal, 1 Voltage [VBUS], 1 Ground [GND])
USB Transfer modes:	Bulk
Voltage:	4.4V - 5.25V. (No external power supply is required)
Current:	250mA (typical operation)
Clock speed:	24Mhz
Device data buffers:	3 x 64-byte control. 4Kbyte data available
Device memory:	8KB Firmware code and data available
Case:	High impact MABS plastic
Size:	65mm x 50mm x 20mm (2.5 x 2.0 x 0.8 inches)
Weight:	35g (1.3oz)

Indicators:	Orange LED = USB2.0 (HighSpeed) or USB1.0 (FullSpeed) indicator
	Green LED = Transmit data
	Red LED 1 = Power from USB bus
	Yellow LED = Receive data
	Red LED 2 = USB bus transmission error indicator
EMC standards:	AS/NZS 3548:1995, EC
Storage Temperature:	-20 °C to + 70 °C
Usage Temperature:	0 °C to + 45°C
Max speed:	480 Mbits/sec
Model number:	PMUSB02 (rev2)
RoHS (Lead free):	Yes

For further details regarding any of the products in this section please see the PassMark website:
<https://www.passmark.com>

See also

USB ports test
 USB test preferences

14.8 Acknowledgements

The 3D test contains modified example code from published Qt examples.

\$QT_BEGIN_LICENSE:BSD\$

You may use this file under the terms of the BSD license as follows:

"Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name of Nokia Corporation and its Subsidiary(-ies) nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

HIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE."

\$QT_END_LICENSE\$

14.9 Making Loopback plugs

You can purchase high quality molded loop back plugs from PassMark or, for the serial and parallel versions, you can make your own. A loop back plug is a small connector that plugs into the serial, parallel or USB port on your computer in order to loop the data output line into the data input line. In the case of the serial and parallel plugs, it also sets a few control lines to trick the computer into thinking that an external device is connected to the port. In the case of the USB plug, an intelligent USB controller chip is required to loop the data.

You'll find below the instructions for making your own serial and parallel loop back plugs. These plugs are designed to be plugged into the small 9 pin serial port and 25 pin parallel port on your computer. In conjunction with software such as BurnInTest they allow the testing of the port.

- Serial Loopback Plug
- Parallel Loopback Plug

Warning: These are the pinouts of the plug that we have made and successfully tested with our own software on our own PC's. You use this information at your own risk! It's easy to make a mistake and we take no responsibility for the potentially serious consequences.

14.9.1 Serial port loopback plug

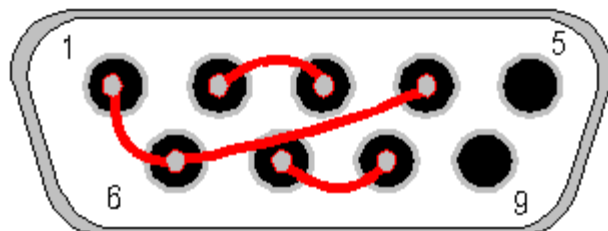
The serial port on IBM compatible PCs conform to the RS232 electrical standard and use a male DB-9 connector. (older PCs may use a DB-25 in which case a converter will be required). The DB-9 connector has the following 9 pins.

DB9 Pin	EIA Description	Description
1	DCD	Data carrier detect
2	RXD	Receive data
3	TXD	Transmit data
4	DTR	Data terminal ready
5	GND	Signal ground
6	DSR	Data set ready
7	RTS	Request to send
8	CTS	Clear to send
9	RI	Ring Indicator

To make the loop back plug the following pins need to be connected together:

- Receive and transmit (2 & 3)
- Request to send and Clear to send (7 & 8)
- Data carrier detect, Data set ready and Data terminal ready (1, 6 & 4)

This diagram shows the connections that need to be made. It's the rear view of the female DB-9 connector that's required for the plug. The red lines and gray dots show the connections that need to be made on the back of the plug.



Female DB-9 connector - Rear view

14.9.2 Parallel port loop back plug

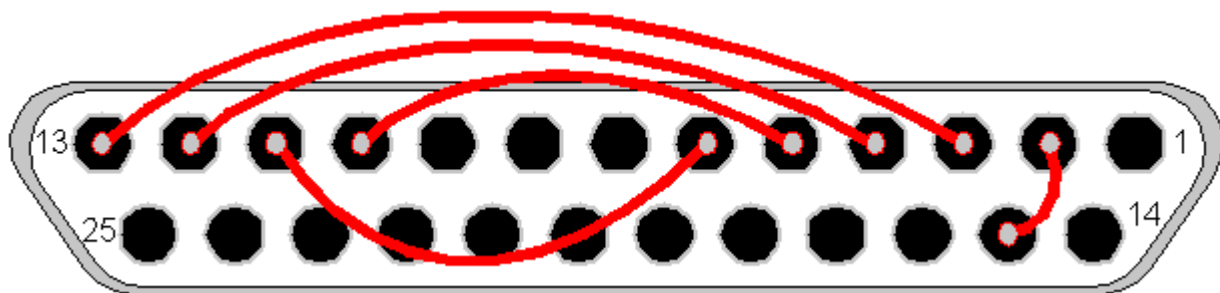
The parallel port on IBM compatible PCs have always used a DB-25 connector. However, over time several changes have taken place to the electrical interface with the introduction of enhanced and bi-directional parallel ports. We believe that this plug will work with all styles of parallel ports.

DB-25 Pin	Direction	Description
2	Out	Data bit 0 (Least significant bit)
3	Out	Data bit 1
4	Out	Data bit 2
5	Out	Data bit 3
6	Out	Data bit 4
10	In	Acknowledge status
11	In	Busy status
12	In	Paper out status
13	In	Select status
15	In	Error status

To make the loop back plug the following pins need to be connected together:

- Data 0 and Error status (Pin 2 & 15)
- Data 1 and Select status (3 & 13)
- Data 2 and Paper out status (4 & 12)
- Data 3 and Acknowledge status (5 & 10)
- Data 4 and Busy status (6 & 11)

This diagram shows the connections that need to be made. It's the rear view of the male DB-25 connector that's required for the plug. The red lines and grey dots show the connections that need to be made.



Male DB-25 connector - Rear view

Index

- A -

Addition 15

- B -

BurnInTest Management Console 63
Buy 8

- C -

CD-ROM/DVD Preferences 34
Command line arguments 61
command line parameter 60
Command line parameters 61
Command Line Version 64
Common error messages 75
Contact details 70
Contact Passmark 70
CPU 32
CPU math preferences 32

- D -

DB25 82, 83
DB9 82
Disk Preferences 30
Disk test suite 16
Division 15
Duty Cycles 42

- E -

Error messages 75
Error preferences 38

- F -

Fault finding 74
File size 16
Firewire 16
Floating point 15

- I -

Integer 15
Introduction 7

- L -

Licensing information 8
Loop back 80
Loopback 82, 83

- M -

Machine Identity 60
Maths test suite 15
Memory card readers 16
Memory preferences 32
Memory sticks 16
Menu 50
Multiplication 15

- N -

Network preferences 32
New Features 8
Notes 60

- O -

Order 8
Ordering information 8
Overview 7

- P -

Parallel 82, 83
parallel port preferences 35
Pattern 16
Pre-test preferences 36
Prices 8
Purchase 8

- R -

RAM 32
RAW disk test 30
Read (disk) 16
Registration information 8
Removable drives 16
RS232 82

- S -

Serial 82
Serial port 82
Serial port preferences 35
Started 8
Subtraction 15
Summary 8
Support 70

- T -

Test descriptions 9
Test Pattern 16
Test Selection 42
Test Selection and Duty Cycles 42
Text based configuration 64

- U -

Universal Serial Bus 80
Un-partitioned disk test 30
Upgrades 8
USB 80
USB 2.0 80
USB preferences 36
USB specifications 80

- W -

Whats new 8
write(disk) 16