Lenovo

Lenovo UEFI Diagnostics

Bootable Version - ARM

v04.43.000

FIT - Flextronics Institute of Technology

July 2025

Contents

1 Introduction **4**

2 Install the Lenovo UEFI Diagnostics **4**
2.1 Step 1: Download the Required Files . . . 4
2.2 Step 2: Create a Bootable USB Flash Drive . . . 4
2.3 Step 3: Running the UEFI Diagnostics . . . 4

3 Privacy Policy **4**

4 Application Overview **6**
4.1 Navigation . . . 6
4.2 Interface Components . . . 6

5 Module Diagnostic **8**
5.1 Execution Flow . . . 8
5.1.1 Select Device . . . 8
5.1.2 Select Algorithm . . . 10
5.1.3 Main Execution . . . 11
5.1.4 Diagnostic Result and Log . . . 12
5.2 Attended Diagnostics . . . 14

6 Run All Diagnostics **15**
6.1 Run All Options . . . 15
6.2 Run All Execution Flow . . . 16

7 Modules and Diagnostics **18**
7.1 CPU . . . 18
7.2 Keyboard . . . 19
7.2.1 Keyboard Diagnostic Interactions . . . 20
7.3 Memory . . . 21
7.4 Motherboard . . . 21
7.4.1 Motherboard Diagnostic Interactions . . . 22
7.5 Mouse . . . 22
7.5.1 Mouse Diagnostic Interactions . . . 22
7.6 Storage . . . 24
7.7 Wired Ethernet . . . 25

8 Lenovo UEFI Tools **25**
8.1 Available Tools . . . 26
8.1.1 System Information . . . 26
8.1.2 Bad Block Recovery . . . 27
8.1.3 Data Disposal . . . 30
8.1.4 SMART Information . . . 31

9 Features **32**
9.1 Times to Run . . . 32
9.2 Screenshot Button . . . 35
9.3 Help Button . . . 35
9.4 Hierarchical diagnostics . . . 36

10 Exit the application **37**

List of Figures

1 Privacy Policy Menu . . . 5
2 QR Code Result . . . 5
3 Home Screen Lenovo UEFI . . . 6
4 Lenovo Header . . . 7
5 Lenovo Footer . . . 7
6 Lenovo Footer - PgUp/PgDn and Home buttons . . . 7
7 Lenovo UEFI Screen Layout . . . 7
8 Lenovo Diagnostics Section . . . 8
9 Device Selection Screen . . . 9
10 Select the "i" option . . . 9
11 Device Additional Information . . . 10
12 Algorithm Selection Screen . . . 10
13 Algorithm Selection Screen Run Button . . . 11
14 Diag Execution Screen Layout . . . 11
15 Diagnostic Result Screen . . . 12
16 View Log Screen . . . 13
17 Sensitive Information Popup . . . 13
18 Log Successfully Saved Popup . . . 14
19 Attended Execution Example 01 . . . 14
20 Attended Execution Example 02 . . . 15
21 Run All Button in Home Screen . . . 15
22 Run All Options . . . 16
23 Run All Execution Screen . . . 16
24 Run All Result . . . 17
25 Run All Log . . . 17
26 Lenovo Diagnostics Section . . . 18
27 Keyboard Custom Execution Screen . . . 20
28 Keyboard Question Popup . . . 20
29 External Ports Execution Screen . . . 22
30 Custom Mouse Execution Screen . . . 23
31 Mouse Question Popup . . . 23
32 Mouse Precision Test Screen . . . 24
33 Tools . . . 26
34 System Information . . . 26
35 Sensitive Popup . . . 27
36 Bad Block Recovery Tool Device Selection . . . 28
37 Bad Block Recovery Tool Algorithm Selection . . . 28
38 Bad Block Recovery Tool Warning . . . 29
39 Bad Block Recovery Execution . . . 29
40 Bad Block Recovery Result . . . 30
41 Data Disposal Quick Information . . . 30
42 Data Disposal Full Information . . . 31
43 Data Disposal Full Information . . . 31
44 SMART Information Tool . . . 32
45 SMART Information Tool . . . 32
46 Times to Run - Module Configuration . . . 33
47 Times to Run - Run All Configuration . . . 33
48 Times to Run Execution Screen . . . 34
49 Times to Run Result Screen . . . 34
50 Times to Run View Log Screen . . . 35
51 Screenshot Button . . . 35
52 Help Popup . . . 36
53 Help Popup . . . 36
54 Hierarchical Popup . . . 37
55 Hierarchical Execution . . . 37
56 Button Exit . . . 38
57 Exit Popup . . . 38

1 Introduction

Lenovo UEFI Diagnostics is a hardware diagnosis tool that enables the users to verify, on their machines, if hardware pieces are presenting problems or malfunctioning, by performing a variety of tests of supported hardware components.

This document describes the main functionalities of the application and how to use them.

2 Install the Lenovo UEFI Diagnostics

2.1 Step 1: Download the Required Files

1. Go to www.Lenovo.com/diags.
2. Select **Downloads**.
3. Under "Lenovo Diagnostics UEFI Bootable Versions," click **How to Create Bootable USB**.
4. Download the appropriate UEFI Diagnostics ZIP file for your machine's architecture.
5. Download the **Bootable Generator ZIP file** and follow the installation instructions.

2.2 Step 2: Create a Bootable USB Flash Drive

1. Insert a USB flash drive into your computer.
2. Run **BootableGenerator.exe** from the desktop or Start menu.
3. Under "Select a device," choose your flash drive. You can rename the drive if needed.
4. Click **Search** and locate the Lenovo Diagnostics Tool file you downloaded earlier.
5. Click **Generate**.
6. A warning will appear stating that all data on the USB will be erased. If this is acceptable, click **Yes** to proceed.

2.3 Step 3: Running the UEFI Diagnostics

1. Insert the prepared USB flash drive into the machine to be diagnosed.
2. Restart the machine and press **F12** during startup to access the boot menu.
3. Select the USB flash drive from the boot menu and press **Enter**.
4. The UEFI diagnostics menu will appear.

3 Privacy Policy

When the user first runs the application, they will be presented with the Privacy Policy menu before accessing the home screen. In this menu, the user will be able to read Lenovo's privacy policies via a QR Code. It is essential that the user accepts this policy, as the application collects diagnostic and usage data necessary to keep Lenovo's UEFI application secure, up to date, and to provide troubleshooting and support.

Figure 1: Privacy Policy Menu

Accepting the Privacy Policy is crucial for the full functioning of the application. Without the user's consent, they will not be able to export diagnostic logs or send data to Lenovo. This data is vital for implementing improvements to the UEFI application. Therefore, it is highly recommended that the user accepts the policy to utilize all the features of the application and contribute to its ongoing maintenance and enhancement.

After completing diagnostics, a QR Code will be generated. Once the user scans this QR Code with the policy acceptance, the diagnostic log data and information about their machine will be automatically sent to Lenovo's servers. This data will undergo future analysis, helping to identify areas for improvement and ensuring that the application continues to meet requirements effectively.

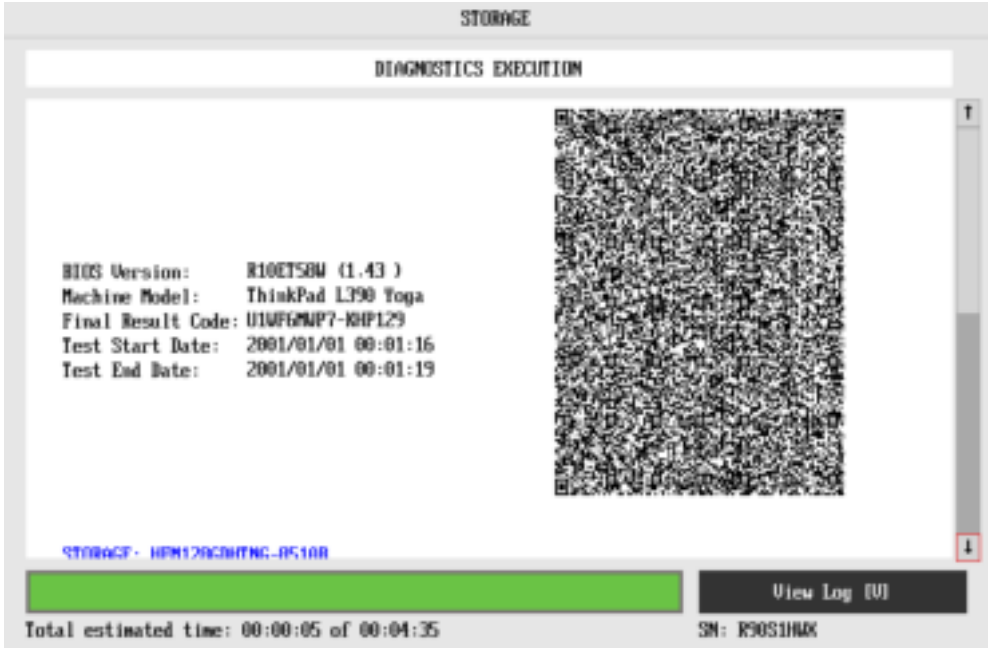

Figure 2: QR Code Result

The user can open the Privacy Policy settings and withdraw their acceptance at any time, at their discretion. By doing so, they will halt data collection and the ability to generate diagnostic logs.

4 Application Overview

When the Lenovo UEFI application starts, the first screen displayed is the home screen, where the user can access all functionalities.

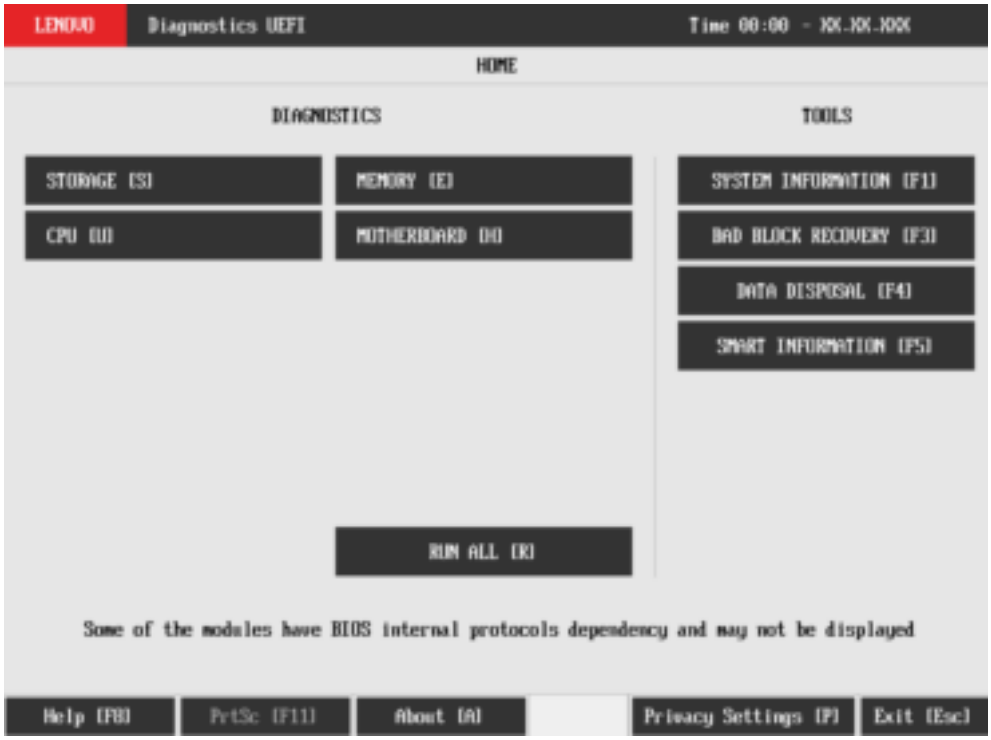

Figure 3: Home Screen Lenovo UEFI

4.1 Navigation

To navigate through the application options, the user can use multiple devices. The selected option will be highlighted in red. To change focus, the user can use a pointing device such as a mouse or touchpad, touch devices if available, or the keyboard arrows and tab key. To select an option, the user can click or touch with the pointing device, or use the enter or space key on the keyboard.

4.2 Interface Components

The Lenovo UEFI application screen layout can be divided into three main sections: the Header, Footer, and Content Area.

The Header displays the application vendor name, which is Lenovo Diagnostics UEFI. It also shows the current system time and the application version. If supported by the system, the header may also display the battery charge level. If this is not shown, it means the feature is not supported. Below the header there will be the current screen name. Check the image below.

Figure 4: Lenovo Header

The Footer provides the user with navigation tips and buttons. The content of the footer can change depending on the screen the user is viewing. If the user is on the home screen, the buttons are: Help, Print Screen, About, Exit and Privacy Policy buttons. However, if the user is on a screen with a scrollbar, there will be options for paging up and down, along with corresponding buttons. If the user is not on the home screen, the button that usually indicates "Exit" will change to a "Home" button, allowing the user to return to the home screen. See the footer options below.

Figure 5: Lenovo Footer

Figure 6: Lenovo Footer - PgUp/PgDn and Home buttons

The Content Area is where the user interacts with the application. It displays the options available on the current page. Each application feature will be explained in future chapters. The content area will show users what they can interact with, such as listing modules, executing diagnostics, checking logs, and more. For the Home Screen, the content area will display all the modules currently supported by the application, allowing the user to access these modules to perform diagnostics on system components.

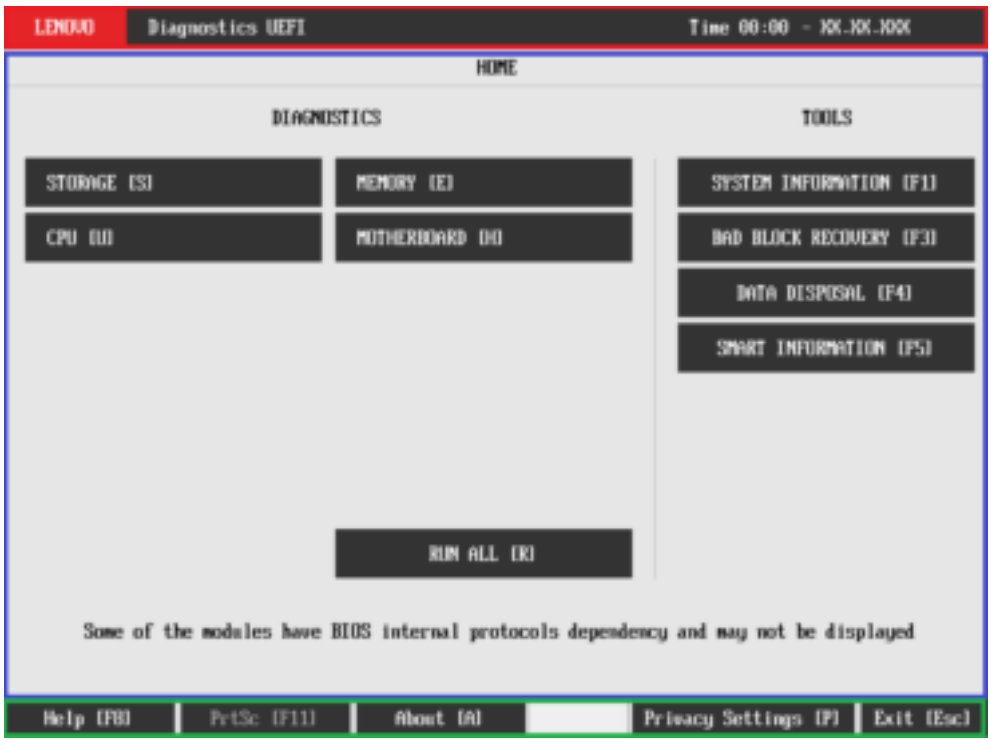

Figure 7: Lenovo UEFI Screen Layout

● Header ● Content ● Footer

5 Module Diagnostic

To perform a diagnostic, the user must first select a module from the Diagnostics section. All available modules are displayed on the Home Screen. Each module represents a device supported by the machine, but not all modules are available for every machine due to certain BIOS protocol requirements. If these protocols are missing, the module will not be accessible. Each module contains a set of diagnostics specific to the devices in the machine.

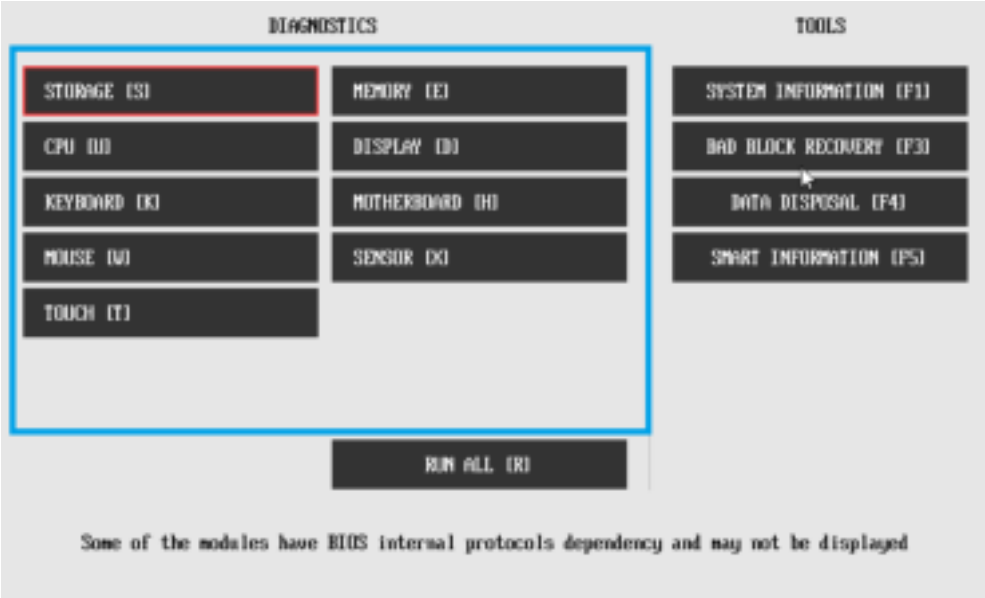

Figure 8: Lenovo Diagnostics Section

● Diagnostics Section

In this section, we will select the Storage module as an example. However, the execution flow is common for all modules.

5.1 Execution Flow

The execution process is similar for all modules, with minor differences for specific ones that will be explained later. This chapter outlines the main steps for most modules.

5.1.1 Select Device

Once the module is chosen, the next step is to select a device for the diagnostic. This is done on the Device Selection Screen, where all available devices for that module are listed.

Figure 9: Device Selection Screen

For some modules, an "i" option may appear next to the device name, allowing the user to view more information about the device by pressing the "i" key or selecting the button in any way.

Figure 10: Select the "i" option

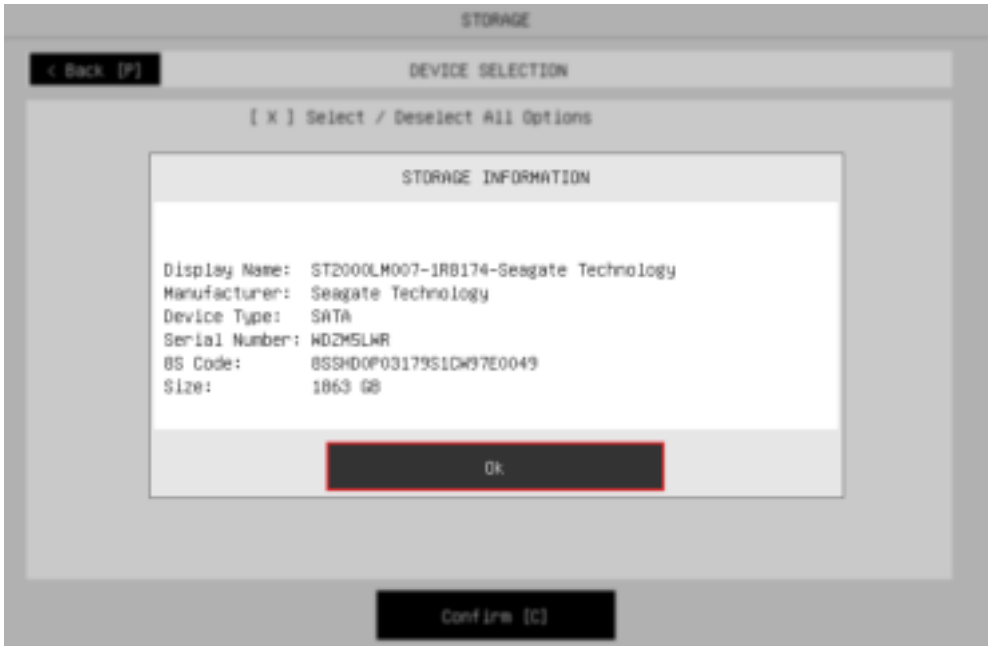

Figure 11: Device Additional Information

5.1.2 Select Algorithm

After the device is selected, the next screen is the Algorithm Selection. Here, the user chooses which algorithms to run. Each algorithm tests different aspects of the selected device. Users can run a single algorithm, all algorithms at once, or select a custom combination. The algorithms are divided between Quick and Extended. The main difference is the time needed to complete the diagnostics; the Quick option is for diagnostics that take less than 10 minutes, while the Extended option is for those that may take longer.

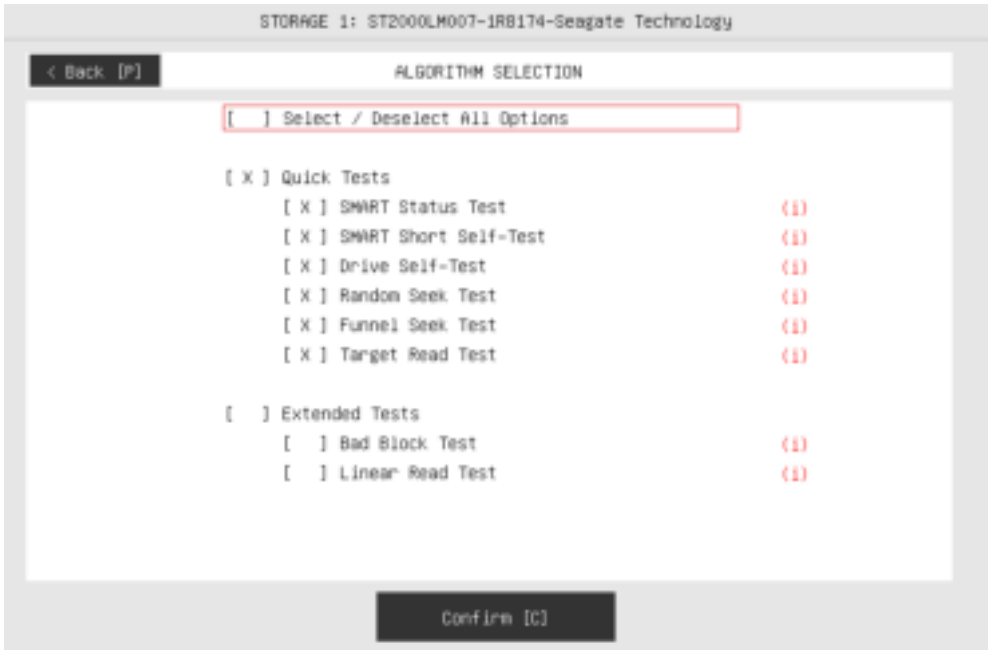

Figure 12: Algorithm Selection Screen

Each device will have a specific Algorithm Selection, and when the user is selecting the diagnostics for the last device, a "Run" button will appear along with a "Times to Run" configuration (explained in the Features section).

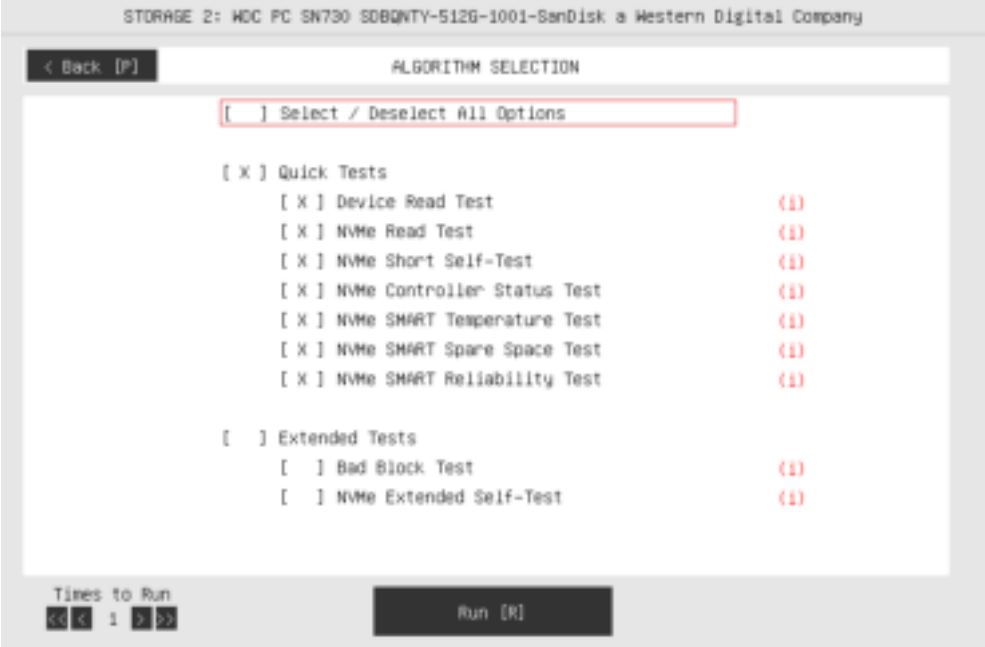

Figure 13: Algorithm Selection Screen Run Button

5.1.3 Main Execution

Once the diagnostics are chosen, the user starts the execution on the Diagnostics Execution Screen, which has three main sections.

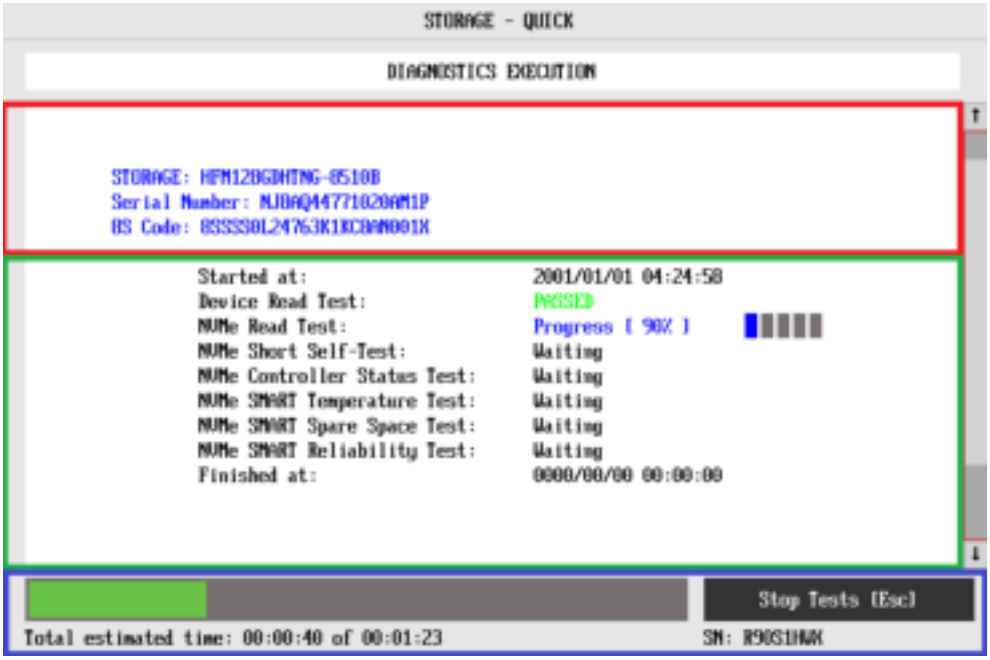

Figure 14: Diag Execution Screen Layout

● Device Information ● Diagnostic information ● Progress information

Device Information: This section shows details about the selected device.

Diagnostics Information: This section shows information about the selected algorithms, including the start time, current percentage of execution, and the status of previous diagnostics.

Progress Information: This section displays a progress bar, an estimated time for completion, and a "Stop Tests" button to cancel the execution.

5.1.4 Diagnostic Result and Log

Once all selected algorithms finish, the next screen is the Diagnostic Result Screen. This screen is similar to the Execution Screen but shows the status of each algorithm, the end time, the QR Code containing some machine information, and a "View Log" button.

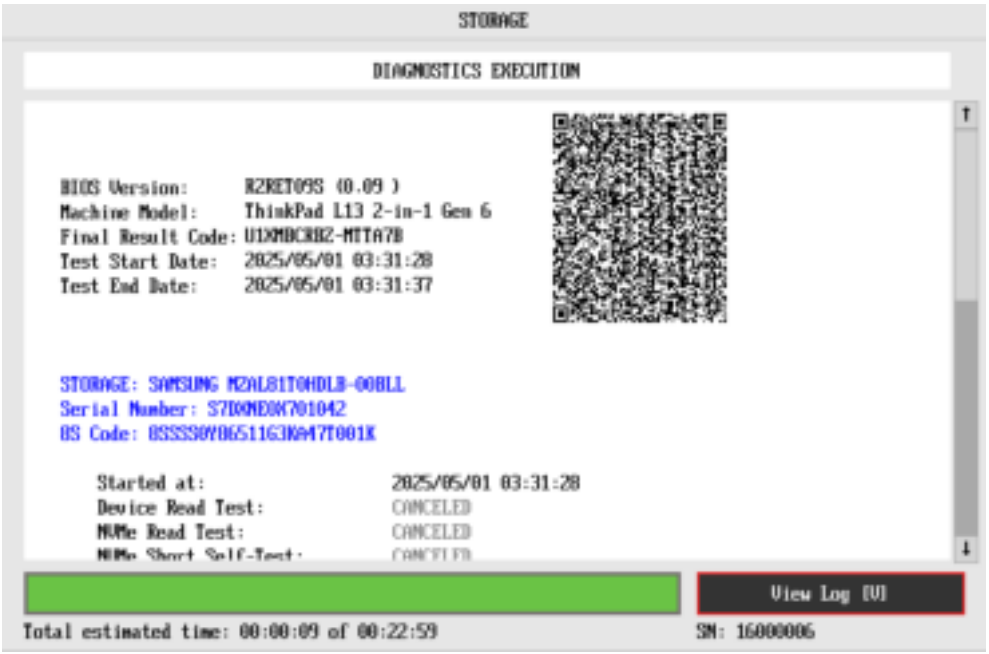

Figure 15: Diagnostic Result Screen

Possible status results include:

- **PASSED** : This status indicates that the algorithm was successfully executed and all validations were met. It means the tested device has no defects in the diagnosed area.
- **FAILED** : This status indicates that the diagnostics could not complete due to unexpected behavior or a malfunction in the device.
- **WARNING** : A warning means the device may have some non-critical issues that need attention, possibly indicating a misconfiguration in the system.
- **CANCELED** : This status is used when the user cancels the execution. It indicates that the process was halted by user action.
- **NOT APPLICABLE** : This status means that the conditions for validating the behavior were not met, making the executed algorithm not applicable.

When clicking the View Log button user is redirect to the Result Log screen, where the diagnostics results are more detailed.

The Result Log Screen provides all information for the selected devices, including a summary of executed diagnostics with start times, names, statuses, and durations.

Figure 16: View Log Screen

There is a "Save Log" button that exports a file to a connected flash drive. Before saving, the application will check if the user has accepted the Privacy Policy, in case user didn't accepted, the application warns the policy should be accepted in order to export logs with sensitive information. If the user agrees, the log is saved to the flash drive.

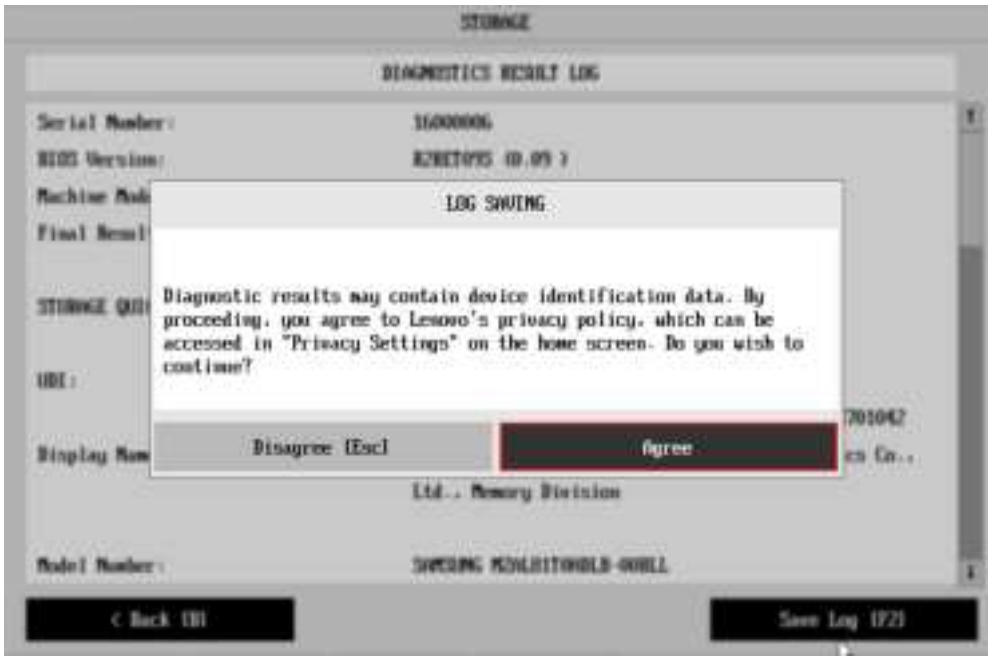

Figure 17: Sensitive Information Popup

Figure 18: Log Successfully Saved Popup

5.2 Attended Diagnostics

The algorithms in Lenovo UEFI Diagnostics application can be devided in two types of execution, which would be attended and unattended diagnostics. The attended diagnostics are those which need the user interaction to complete and get the results. The attended diagnostics have an execution flow a bit different from the explained above, that's because the Diagnostic Execution Screen is customized so the user can interact with.

Below images show examples of attended tests with the popup information that asks the user to perform certain action. The attended diagnostics flow will be explained for each module in the module specific section.

Figure 19: Attended Execution Example 01

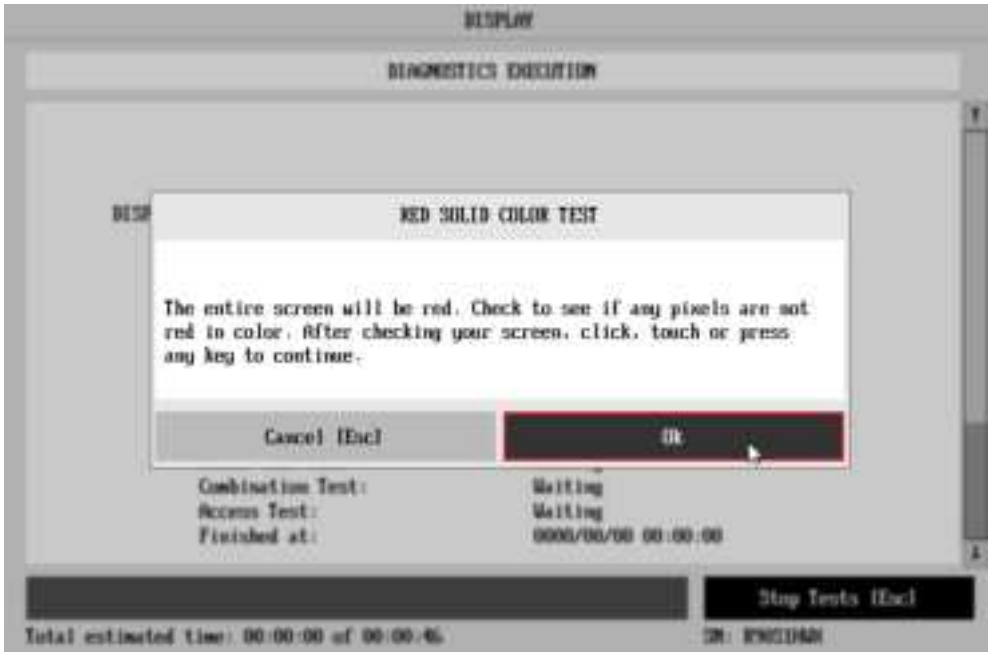

Figure 20: Attended Execution Example 02

6 Run All Diagnostics

In addition to the common execution flow described above, there is another way for the user to perform diagnostics in the Lenovo UEFI Diagnostics application, which is through the Run All section. The Run All option can be accessed from the Home screen by clicking the Run All button located below the Diagnostics Section

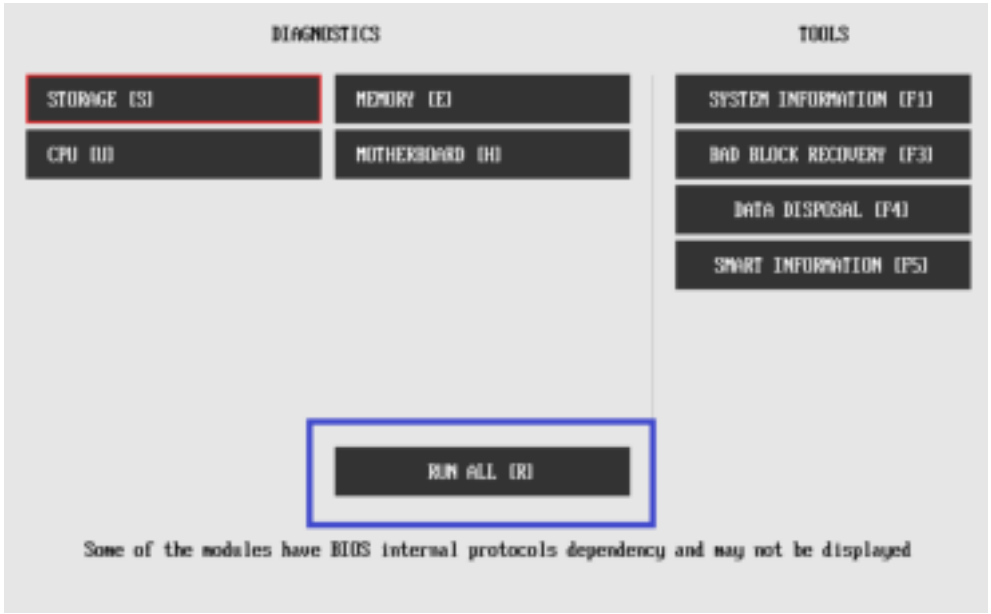

Figure 21: Run All Button in Home Screen

● Run All Button

6.1 Run All Options

When the user clicks the Run All button, the application redirects them to the Run All Diagnostics Type Screen. The Run All options combine diagnostics from all available modules in the application and execute all of them in a single run. There are different combinations for these diagnostics.

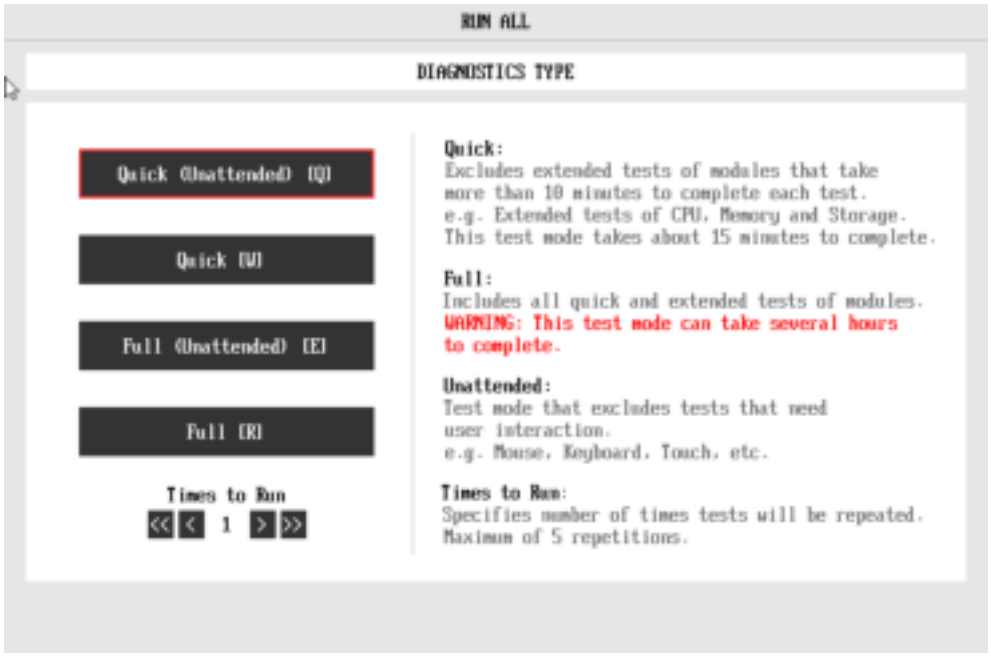

Figure 22: Run All Options

The **“Quick (Unattended)”** and **“Quick”** options select all available tests considered Quick, which, as mentioned in the previous chapter, are those that take less than 10 minutes to complete. The **“Full (Unattended)”** and **“Full”** options select all available diagnostics, including Quick and Extended executions. The options marked as “Unattended” represent diagnostics that do not require user interaction, while the other options require user interaction during some of the diagnostics.

6.2 Run All Execution Flow

After choosing one of the options, the application will redirect the user to the Run All Execution Screen. This screen is the same from the previously explained Execution Screen, as shown in the image below:

Figure 23: Run All Execution Screen

Once the execution is complete, the application shows the Run All Result Screen. The layout is similar to the Run All Execution Screen, but with the addition of a button to view the execution log, a QR Code containing some additional machine information and the Final Result Code. Also, on the left side of the QR Code it's possible to see some machine information and the Status result counter.

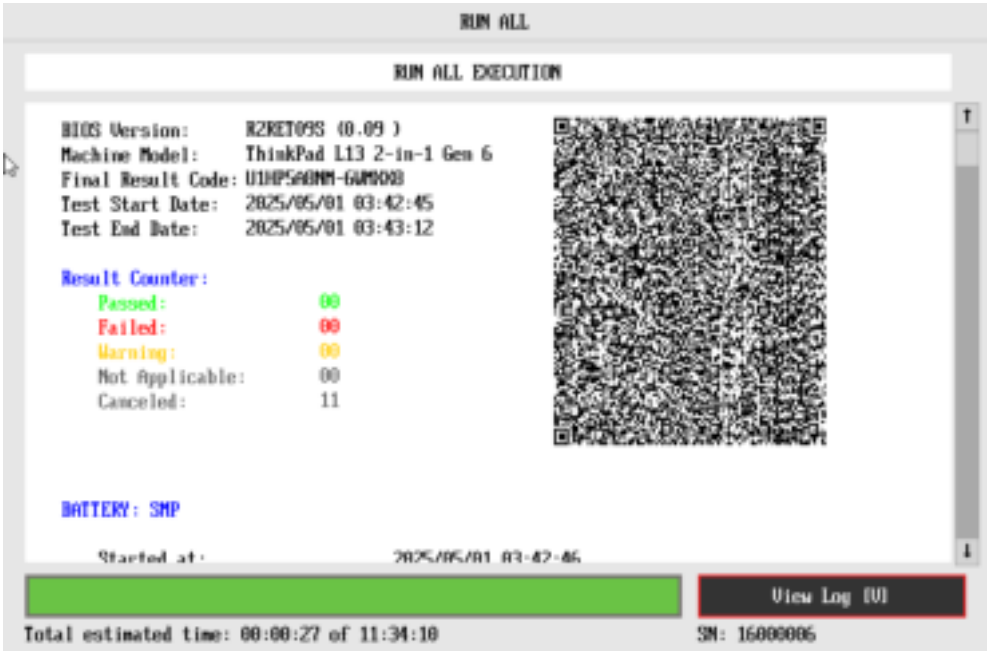

Figure 24: Run All Result

When the user clicks the View Log button, they are redirected to the Run All Diagnostics Result Log screen, where a Save Log button can be found. The structure of the Run All log and the behavior of the save button follow the same process described in the Diagnostic Result and Log section of the Module Diagnostic chapter.

Figure 25: Run All Log

7 Modules and Diagnostics

The Lenovo UEFI Diagnostics currently supports a list of seventeen different modules:

- CPU
- Keyboard
- Memory
- Motherboard
- Mouse
- Storage
- Wired Ethernet

In this chapter, we will cover each supported module and its diagnostics, along with explanations for the attended diagnostics, which focus on how the user should interact with the application during the diagnostics execution.

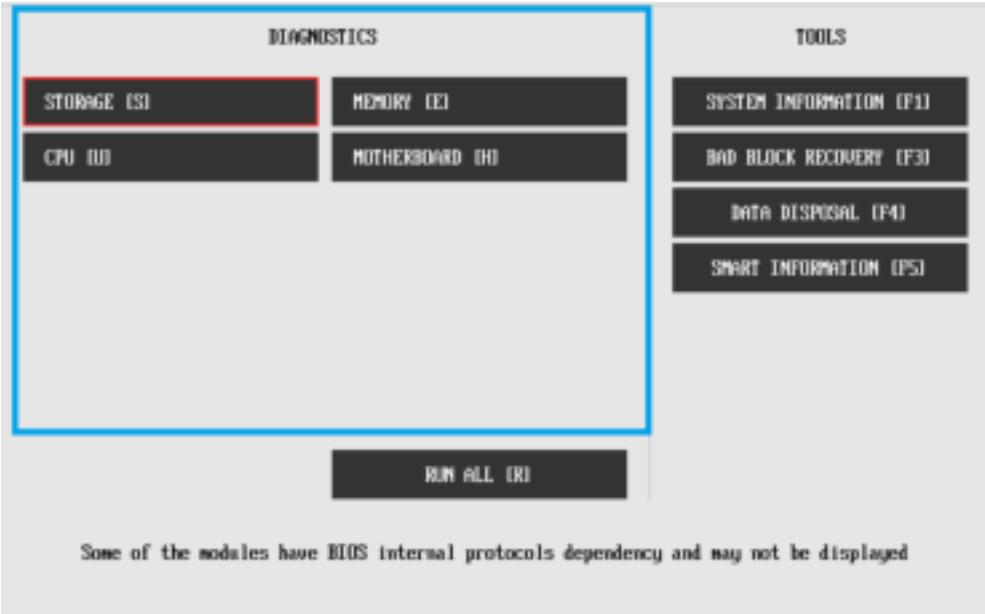

Figure 26: Lenovo Diagnostics Section

● Diagnostics Section

7.1 CPU

Quick Diagnostics:

- Math Test: This set of tests performs arithmetic operations (addition, subtraction, multiplication and division) of integer numbers using 32-bits and 64-bits registers.
- Floating Point Test: This set of tests is similar to the Math Test, it performs arithmetic operations of floating point numbers.
- Extension Instruction Test: This set of tests executes instructions to extend values in registers.
- Zero Register Test: This test checks the availability of the registers XZR (32-bits) and WZR (64-bits).
- Neon Test: Arm Neon technology is a 64-bit or 128-bit hybrid Single Instruction Multiple Data (SIMD) architecture that is designed to accelerate the performance of multimedia and signal processing applications.
- Register Test: The test makes various arithmetic operations in all available registers in order to verify if the values read in these registers are the same that were previously written.

- BT Instruction Test: The test checks if the BT (Bit Test) instruction is working properly.
- X87 Floating Point Test: The x87 instruction set includes instructions for basic floating point operations (such as addition, subtraction and comparison) and also for complex numerical operations (such as trigonometric functions).
- MMX Test: MMX technology is designed to accelerate the performance of advanced media and communications applications.
- 3DNow! Test (AMD Only): Tests the 3DNow! instructions on AMD processors.
- SSE Test: SSE Technology enhances the performance of processors for advanced 2-D and 3-D graphics, motion video, image processing, speech recognition, audio synthesis, telephony, and video.
- AES Test: The purpose of the instruction set is to improve the speed of applications performing encryption and decryption using the Advanced Encryption Standard.
- FMA Test: The FMA instruction set is an extension of SSE instructions to perform "fused multiply-add" operations.
- AVX2 Test: AVX2 (Advanced Vector Extensions 2) is 256-bit instruction set that expands the AVX (Intel Advanced Vector Extensions) Instruction Set, and its purpose is to extend the vector processing capabilities across floating-point and integer data domains.
- Cache Test: Cache test is to actually perform some indirect test via code implemented in C language, which will force the cache hierarchy to work.
- AVX512 Test: AVX-512 is a SIMD (Single Instruction, Multiple Data) instruction set that can process up to 512 bit of data in a single fetch.
- CLMUL Test: The CLMUL is an extension to the x86 instruction set that provides a hardware instruction to perform a carry-less multiplication.
- SHA Test: The SHA Extensions are a family of seven Streaming SIMD Extensions (SSE) based instructions that are used together to accelerate the performance of processing SHA-1 and SHA-256 on x86 architecture processors.
- TSX Test: The TSX is an extension of the x86 instruction set architecture that adds hardware transactional memory support, speeding up execution of multi-threaded software through lock elision

Extended Diagnostics:

- Stress Test: In Stress Test, all Features and Instructions quick tests, which are supported by the processor, are executed cyclically in each core at the same time.

7.2 Keyboard

Keyboard module is only supported for ThinkPad machines

Quick Diagnostics:

- PS/2 Test: This test performs a reset operation on the hardware executing the Simple Text Protocol reset command for PS/2 devices.
- USB Test: Checks if all necessary UEFI protocols and structures are correctly allocated and working with the specified handle of the actual physical keyboard device.
- Key Test: Is an attended test where the user checks whether the keys are properly working for PS/2 keyboards.
- USB Key Test: Is an attended test where the user checks whether the keys are properly working for USB keyboards.

7.2.1 Keyboard Diagnostic Interactions

For the keyboard interaction in Key Test and USB Key Test a customized execution screen will appear for the user so he can validate the keys of his device. The idea is that for each pressed key the application will show in a text box if the key was properly recognized. At the end of test when user quit the application will ask if the keys were correclty identified. Wrong user answers will lead to wrong test results.

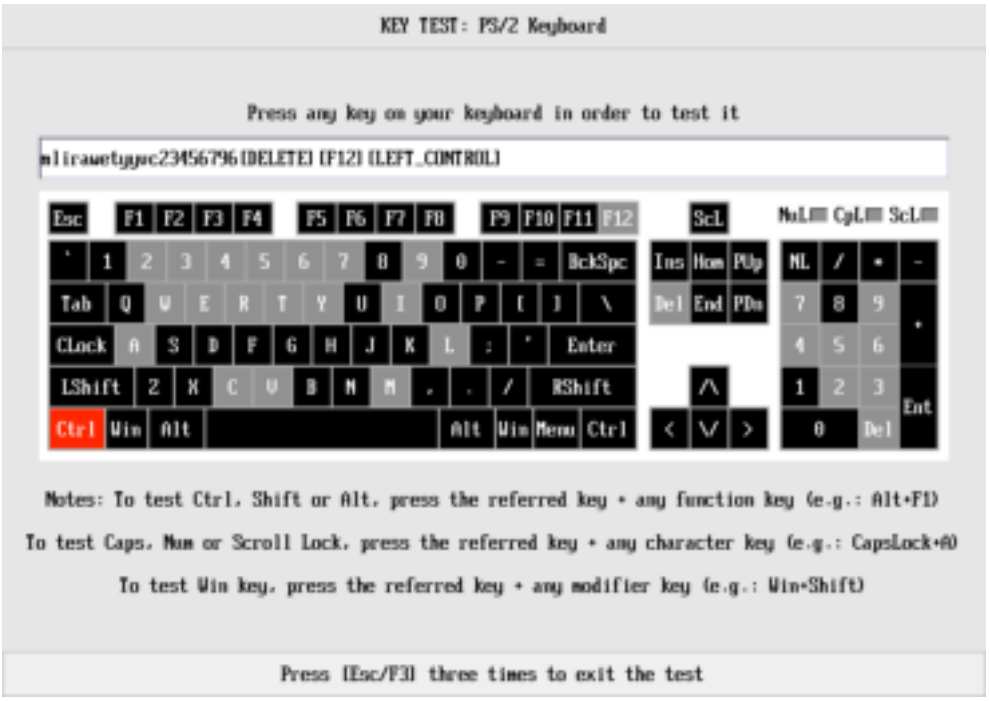

Figure 27: Keyboard Custom Execution Screen

Figure 28: Keyboard Question Popup

7.3 Memory

Quick Diagnostics:

- Quick Random Pattern Test: The test consists in filling the memory from every block with a random generated pattern and then checking if such pattern was correctly written.

Extended Diagnostics:

- Advanced Integrity Test: The Advanced Integrity Test is based in March C Minus algorithm, up to this date the LDiags UEFI uses three variations of March C Minus: two variations of Modified March C- and one of Enhanced March C-.
- Address Test: This test consists in writing to each memory address its own address, for each memory block. After that, the algorithm reads the memory previously written and checks if they still store their own address.
- Bit High Test: This test consists in filling the memory buffer from every block with a pattern where all bits are 1 and then checking it.
- Bit Low Test: This test consists in filling the memory buffer from every block with a pattern where all bits are 0 and then checking it.
- Walking Ones Right Test: The Walking Ones Right Test consists of writing a pattern where only the leftmost bit is set (e.g. 10000000), then shift this pattern to the right (e.g. 01000000) until the end of the size of a byte, writing it again at the same memory address each time such pattern is shifted.
- Walking Ones Left Test: The Walking Ones Left Test consists on writing a pattern where only the rightmost bit is set (e.g. 00000001), then shift this pattern to the left (e.g. 00000010) until the end of the size of a byte, writing it again at the same memory address each time such pattern is shifted.
- Modulo-20 Test: The test consists in writing into an interval of 20 memory locations for each block with a pattern and filling all other locations with its complement 6 times.
- Moving Inversions - 8-bit Test: The test consists in filling the memory with the 8 bit wide pattern: 10000000 and then checking if such pattern was correctly written.
- Moving Inversions - 32-bit Test: This test fills all the accessible memory with a shifting pattern, that is, a value which is binary left shifted as it is written out through the accessible memory of every memory block.
- Random Pattern Test: The test consists in filling the memory from every block with a random generated pattern and then checking if such pattern was correctly written.
- Random Number Sequence Test: The test consists in filling the memory with one different random generated pattern for each memory address and then checking if the pattern was correctly written.
- Block Move Test: The algorithm consists of moving memory data around within memory blocks.
- Bit Fade Test: The Bit Fade test consists of filling all the accessible memory with 0s, waiting for a time period and checking if these values haven't changed.

7.4 Motherboard

Quick Diagnostics:

- PCI/PCIe Test: Validates the integrity of different PCI and PCI-e capabilities.
- RTC Test: This test checks the accuracy of the RTC when date and time rollovers occur besides verify the accuracy of the RTC on the motherboard and compares it to the CPU clock to determine if the motherboard and CPU clock are out of sync.
- USB Test: Assesses the functionality of USB ports.
- USB Replaceable Type-C Test: Detects issues with replaceable USB-C ports by checking the status of each detected USB-C port. If a failure is detected, only the connector or small board needs to be replaced, not the entire motherboard.
- External Ports Test: Validates the connectivity of the external ports (only USB supported for now).

7.4.1 Motherboard Diagnostic Interactions

For the External Ports Test, there is a custom execution screen that counts the number of external port connections during the test run. When the user connects a device to the external ports (USB-A or USB-C), the counter increments, indicating to the user that the port has been validated.

Figure 29: External Ports Execution Screen

7.5 Mouse

Mouse module is only supported for ThinkPad machines

Quick Diagnostics:

- Reset Test: This test performs a "reset" operation on the hardware.
- Mouse Test: Attented test that lets the user test the click of Left and Right buttons of the mouse and the mouse movement detection, if available.
- USB External Mouse Test: Same as Mouse Test but for the USB external mouse.
- Precision Test: An attended test that validates the mouse's ability to hold click and drag.

7.5.1 Mouse Diagnostic Interactions

For the Mouse Test and USB External Mouse Test a customized screen will be displayed to the user so he can validate the mouse behaviros of movement and button clicks. The customized screen will indicate the movements and buttons user interact with. After the user quits the application will ask if everything work as expected. Wrong answers lead to wrong results.

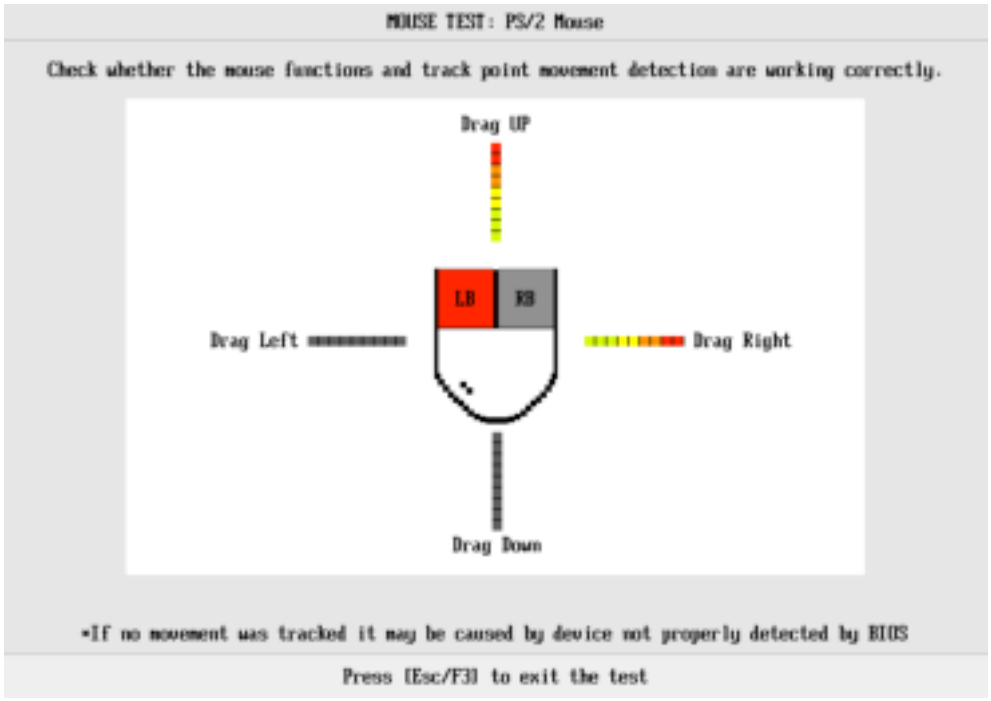

Figure 30: Custom Mouse Execution Screen

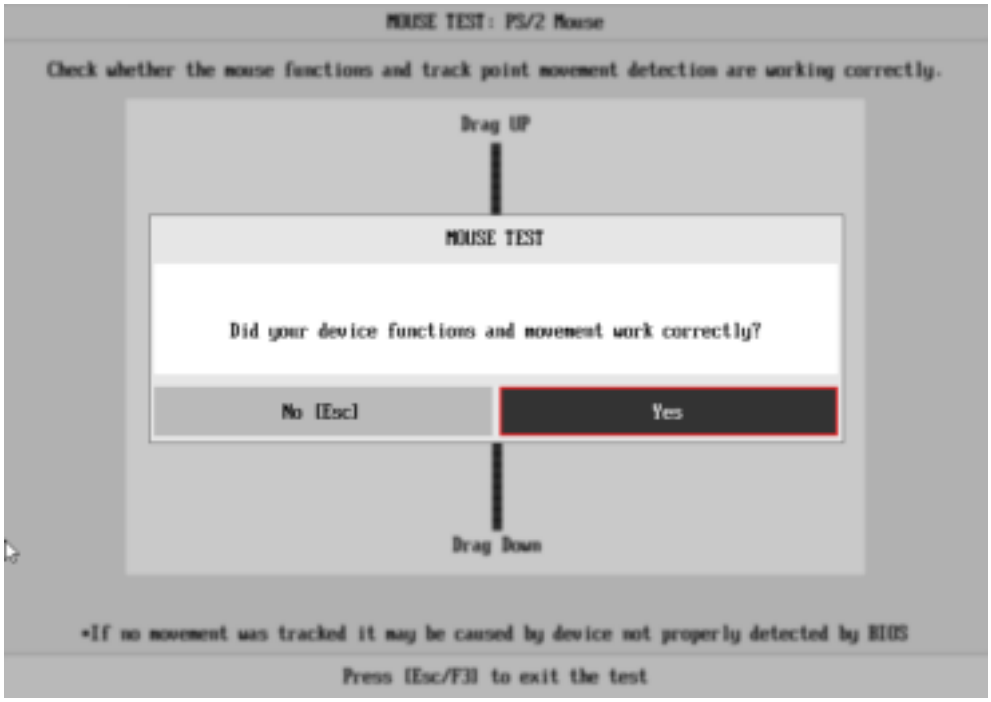

Figure 31: Mouse Question Popup

The Precision Test also utilizes a customized screen to perform the diagnostic. The intent of this test is to validate the mouse's functionality when holding the click and dragging the cursor. To successfully complete the test, the user must drag a scrollbar down. If the user is unable to drag the scrollbar down, the diagnostic will be considered a failure.

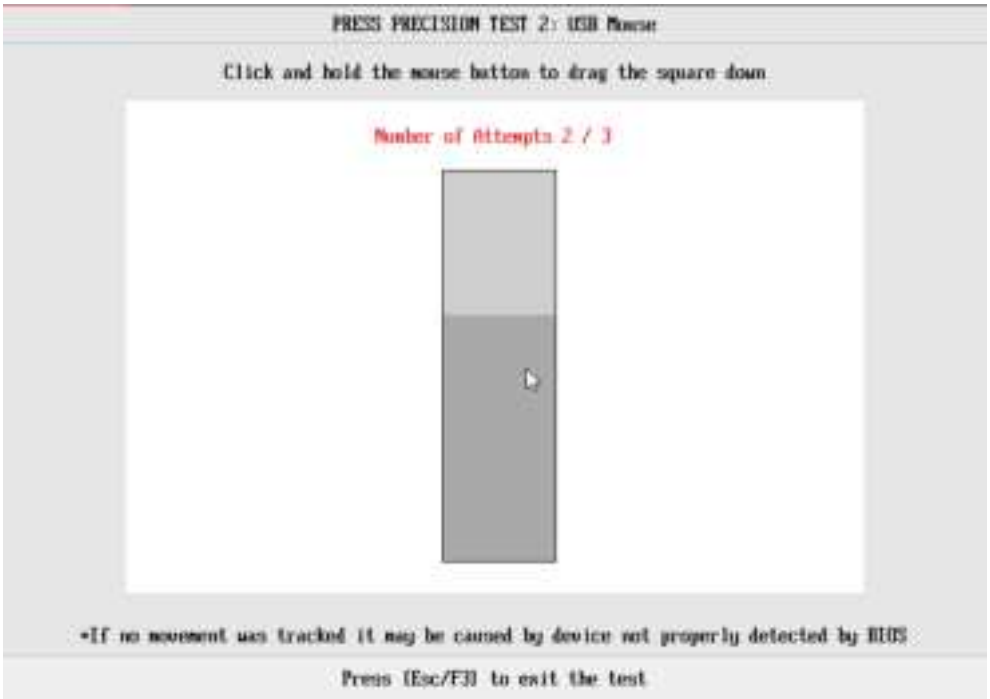

Figure 32: Mouse Precision Test Screen

7.6 Storage

The storage devices connected as RAID will be disabled by UEFI diagnostics application, therefore they can't be tested on the Storage.

Quick Diagnostics:

SATA:

- SMART Status Test: The application will check de SMART registers present on the storage device.
- SMART Short Self-Test: The application sends a command to the storage device, to start a Self-Test routine in background mode
- Drive Self-Test: The Drive Self-Test follows the same flow as the SMART Self-Test.
- Random Seek Test: Checks sectors at several randomly chosen addresses.
- Funnel Seek Test: The application will try to find defective sectors of the first 100 sectors.
- Target Read Test: The application sends a command to the storage device, to read the device log.

SSD:

- SMART Status Test: The application will check de SMART registers present on the storage device.
- SMART Short Self-Test: The application sends a command to the storage device, to start a Self-Test routine in background mode
- Target Read Test: The application sends a command to the storage device, to read the device log.
- SMART Wearout Test: Checks the wearout level of the attached SSD device by reading SMART attributes and tells you whether the device is in good condition or has reached its wearout limit.

eMMC:

- Device Read Test: Verifies a storage device in order to detect possible disturbs regarding the reading process

NVMe:

- Device Read Test: Verifies a storage device in order to detect possible disturbs regarding the reading process
- NVMe Controller Status Test: Checks the status of the controlller by reading the Status register
- NVMe SMART Temperature Test: Checks if a given NVMe device has passed the critical temperature threshold.
- NVMe SMART Spare Space Test: Check if the spare space of a given NVMe device is near to end.
- NVMe SMART Reliability Test: Checks if a given NVMe device is still reliable, based on SMART metrics
- NVMe Short Self-Test: Starts a Self-Test routine in background mode.

Optane:

- Device Read Test: Verifies a storage device in order to detect possible disturbs regarding the reading process
- Optane Health Test: This "Health Test" verifies if the Optane device is healthy.

UFS:

- Device Read Test: Verifies a storage device in order to detect possible disturbs regarding the reading process

Extended Diagnostics:

- Bad Block Test (For SATA, SSD, and NVMe): Identifies defective sectors on the drive.
- Linear Read Test (For SATA and SSD): Scans the whole drive using a pre-defined block size that is used on each step of the test.
- NVMe Extended Self-Test (For NVMe): Starts a Self-Test routine in background mode.
- UFS Read Test (For UFS): The purpose of this test is to verify the integrity of the UFS device, access its read capabilities, and identify any potential issues or errors.

7.7 Wired Ethernet

Quick Diagnostics:

- Internet Connection Test: The purpose of the Internet Connection Test for Wired Ethernet adapters is to check the internet connection and if the hardware device is working properly by performing a "ping" operation in a list of DNS addresses.

8 Lenovo UEFI Tools

In addition to the diagnostics available within the modules and the "Run All" option, the Lenovo UEFI Application provides several tools for performing specific operations on the machine. These tools can be accessed from the application Home Screen. To execute a tool, the user should select the desired option and follow the respective workflow.

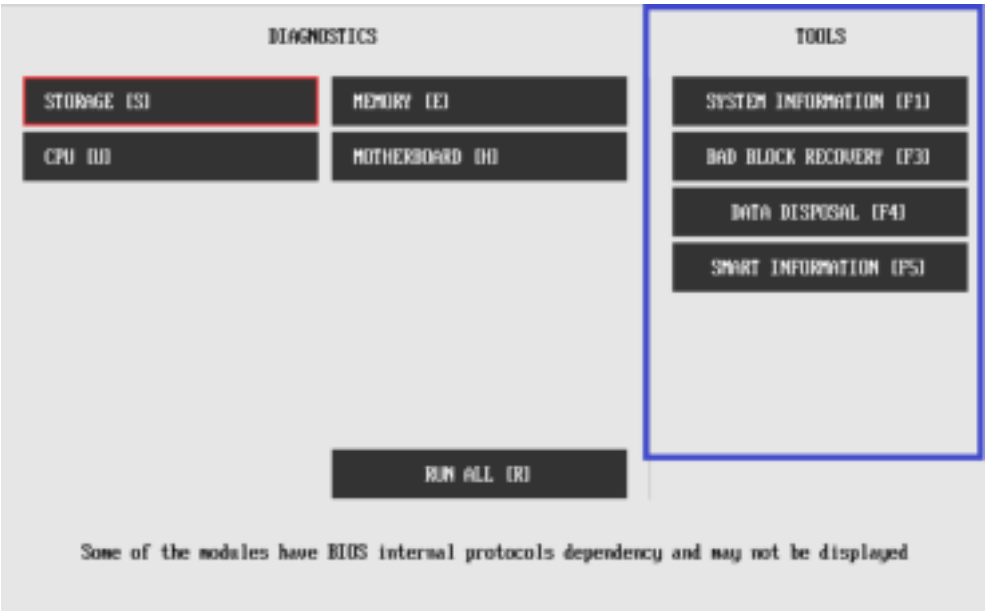

Figure 33: Tools

● Tools Section

8.1 Available Tools

Not all tools are available on every machine. Similar to the modules, some tools depend on specific BIOS protocols. If a machine does not meet the required dependencies, the tool will not be supported or displayed.

- System Information
- Bad Block Recovery
- Data Disposal
- SMART Information

8.1.1 System Information

The System Information tool provides users with detailed information about all supported modules. Each module will appear on the screen with specific details.

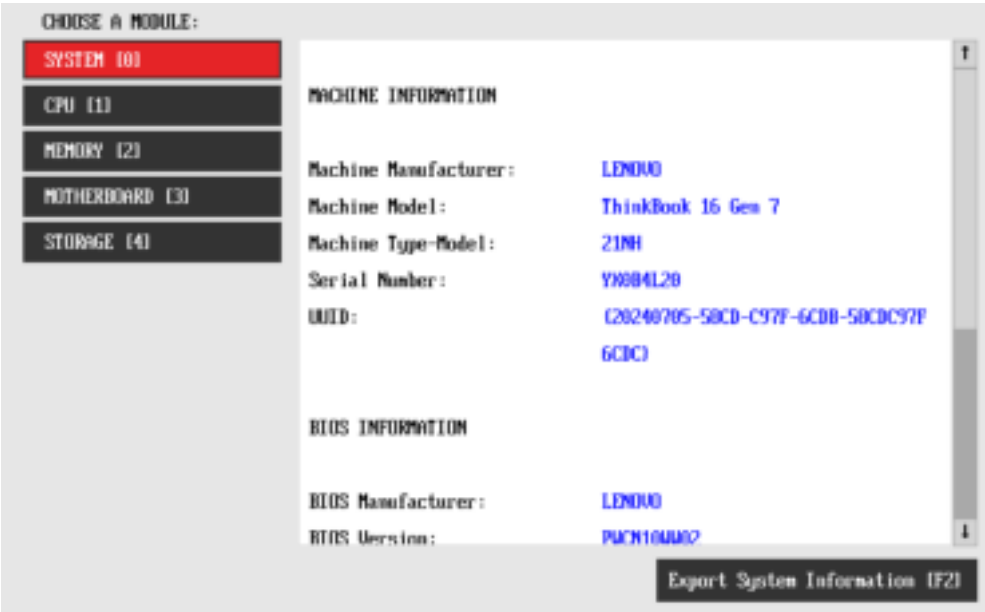

Figure 34: System Information

In addition to displaying information, the System Information tool allows users to export the displayed data as a text file. To do this, the user must select the “Export System Information" button or press the shortcut F2. A popup will then alert the user about the sensitive nature of the information being exported. If the user consents, the file will be generated.

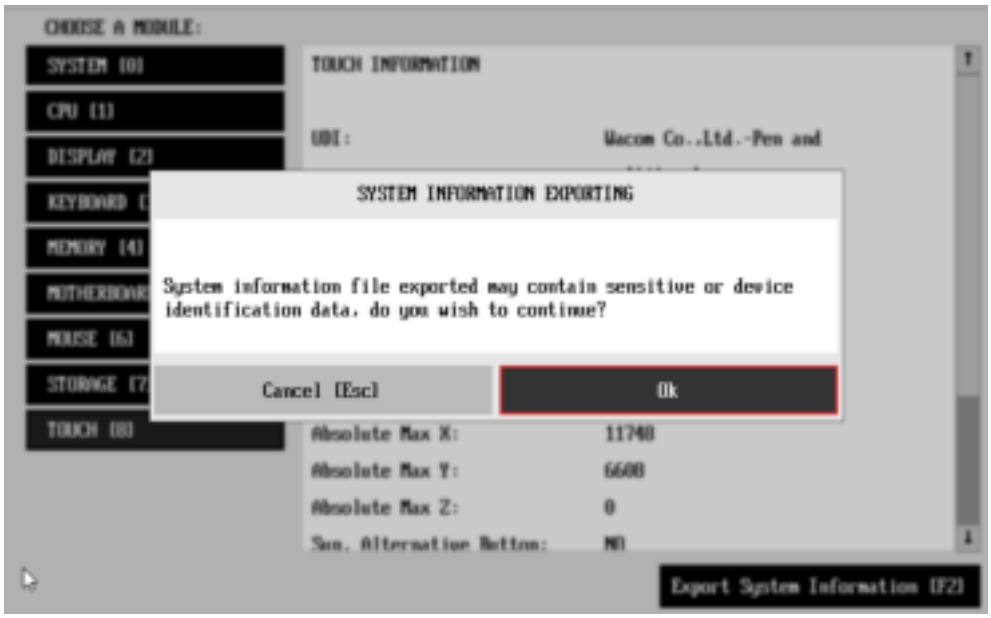

Figure 35: Sensitive Popup

8.1.2 Bad Block Recovery

The Lenovo UEFI Bad Block tool is designed to read a storage device while searching for potential bad blocks. A "block" refers to the smallest addressable region on a storage device. This tool not only identifies bad block sectors but also attempts to repair them, making them valid addresses again.

The workflow for this tool is similar to the diagnostics execution flow. It begins with the user selecting the device on which to run the Bad Block Recovery tool.

Figure 36: Bad Block Recovery Tool Device Selection

After selecting the device, the tool will redirect the user to choose an algorithm for the Bad Block Recovery. In this screen, users can access information about the algorithm used.

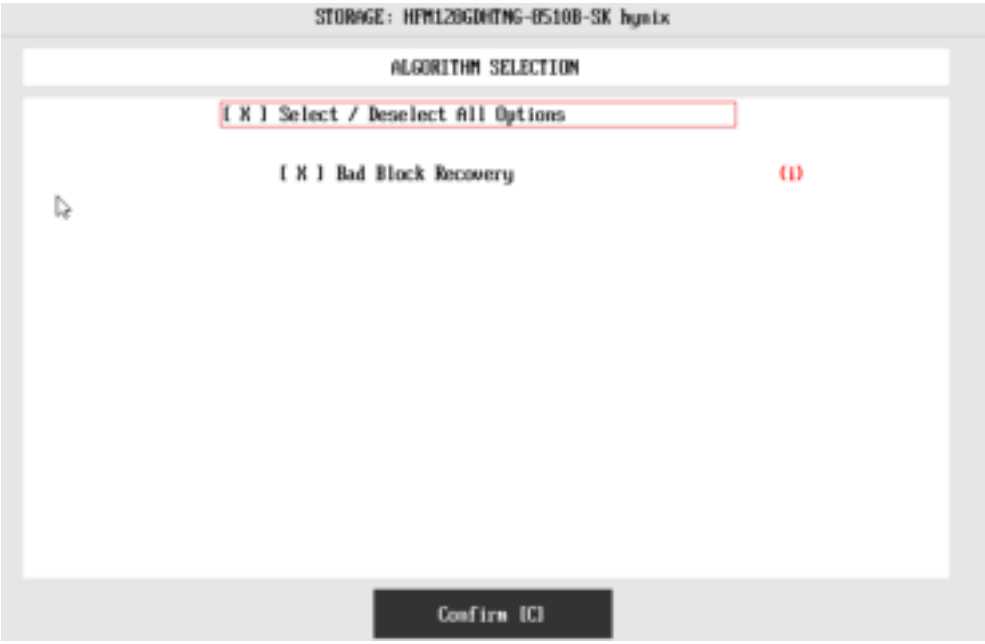

Figure 37: Bad Block Recovery Tool Algorithm Selection

Once the algorithm is selected, the tool will begin searching for bad blocks and attempting repairs. Before starting, a warning popup will remind the user to back up their data. If the user consents, the process will start.

Figure 38: Bad Block Recovery Tool Warning

The execution flow and view log option are identical to those described in the “Executing Diagnostics" chapter. The difference is in the Bad Block Recovery Result Screen, which does not contain the QR Code or Final Result Code.

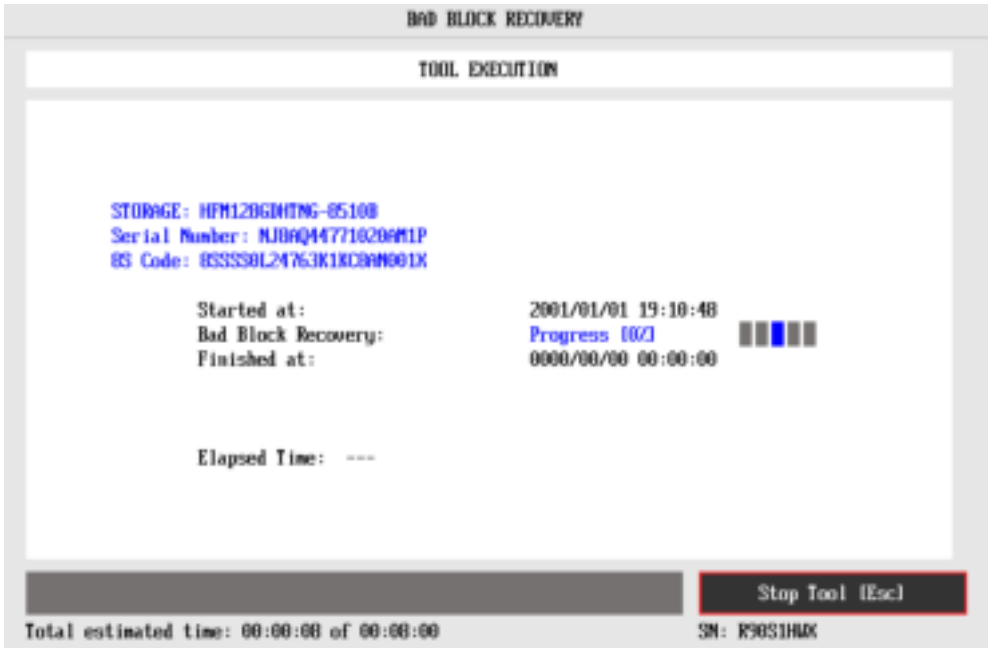

Figure 39: Bad Block Recovery Execution

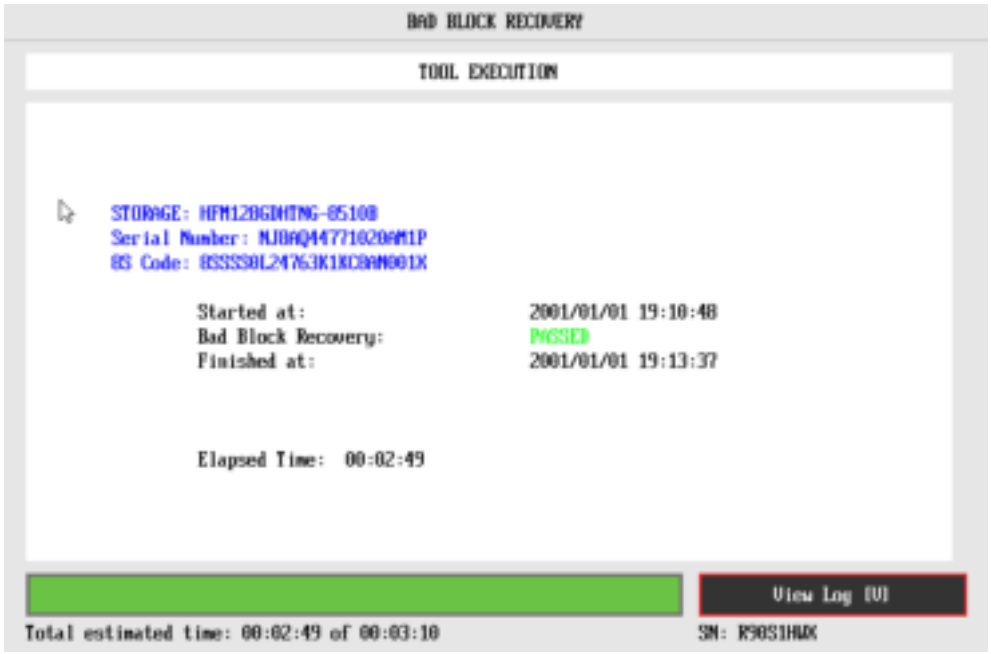

Figure 40: Bad Block Recovery Result

8.1.3 Data Disposal

The Data Disposal tool is responsible for erasing all data in storage partitions. It offers two algorithms: Data Disposal Quick and Data Disposal Full. The Full option erases all data from the storage device, while the Quick option writes zeros to the MBR, GPT, and GPT Backup table across the entire disk.

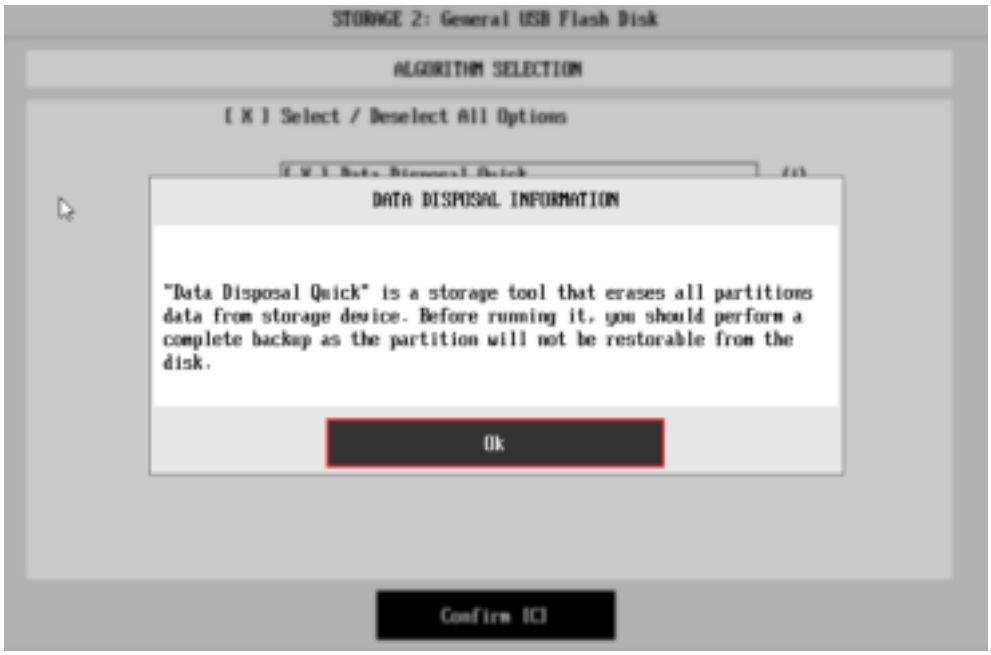

Figure 41: Data Disposal Quick Information

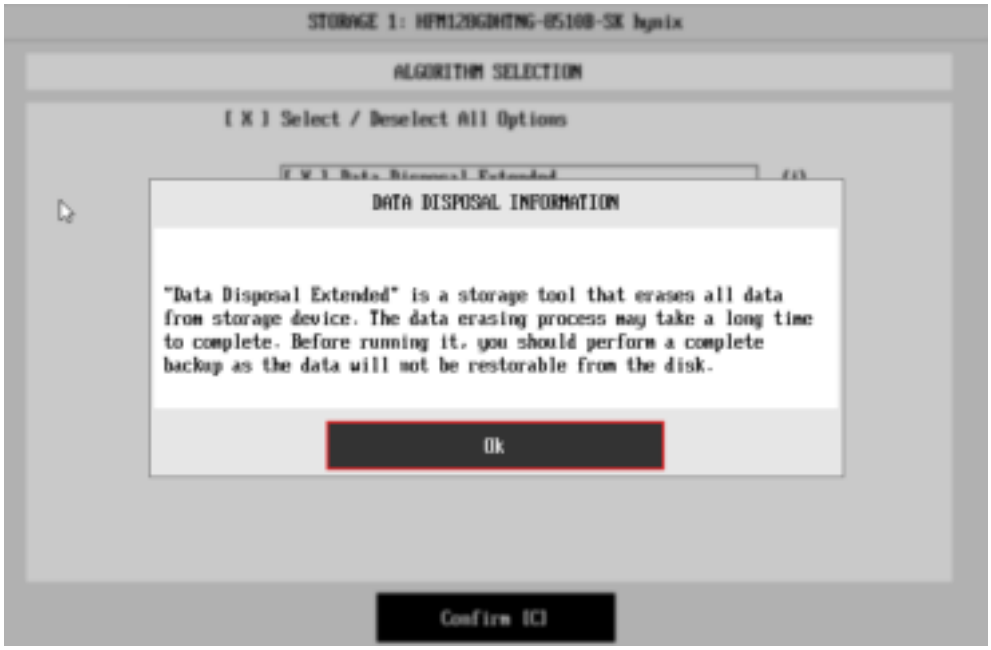

Figure 42: Data Disposal Full Information

Similar to the Bad Block Recovery tool, this tool follows an execution flow outlined in the “Executing Diagnostics" chapter. The user begins by selecting a device and then chooses the desired algorithm, either Quick or Full, if supported. Once everything is selected the execution can start, and before it starts a warning popup will appear. The user

Figure 43: Data Disposal Full Information

8.1.4 SMART Information

The SMART Information tool provides insights into the hardware condition as reported by the S.M.A.R.T. (Self-Monitoring, Analysis, and Reporting Technology) monitoring system for HDDs, SSDs, and NVMe devices, helping to prevent imminent hardware failures.

Figure 44: SMART Information Tool

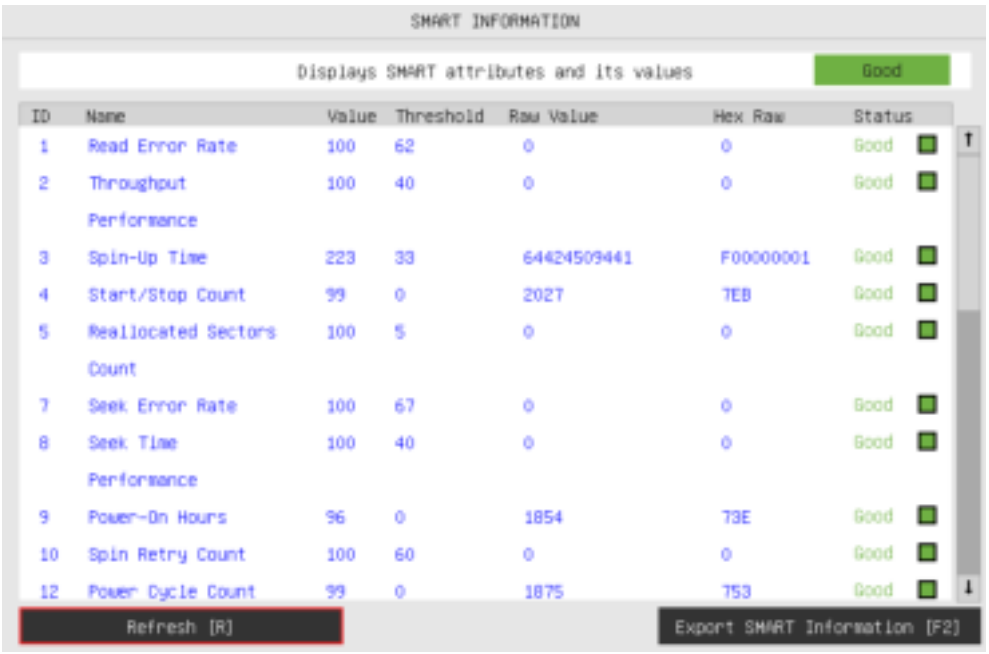

Figure 45: SMART Information Tool

9 Features

This section outlines the various features available in the application, providing users with tools to enhance their experience and streamline operations.

9.1 Times to Run

The Times to Run feature allows users to run the same suite of diagnostics multiple times. It essentially repeats the selected diagnostics based on the number specified in the Times To Run section. This section can

be found in the Algorithm Selection for standard modules and in the Run All section. The Run All configuration has a limit of 5 executions, while the maximum for standard modules is 50.

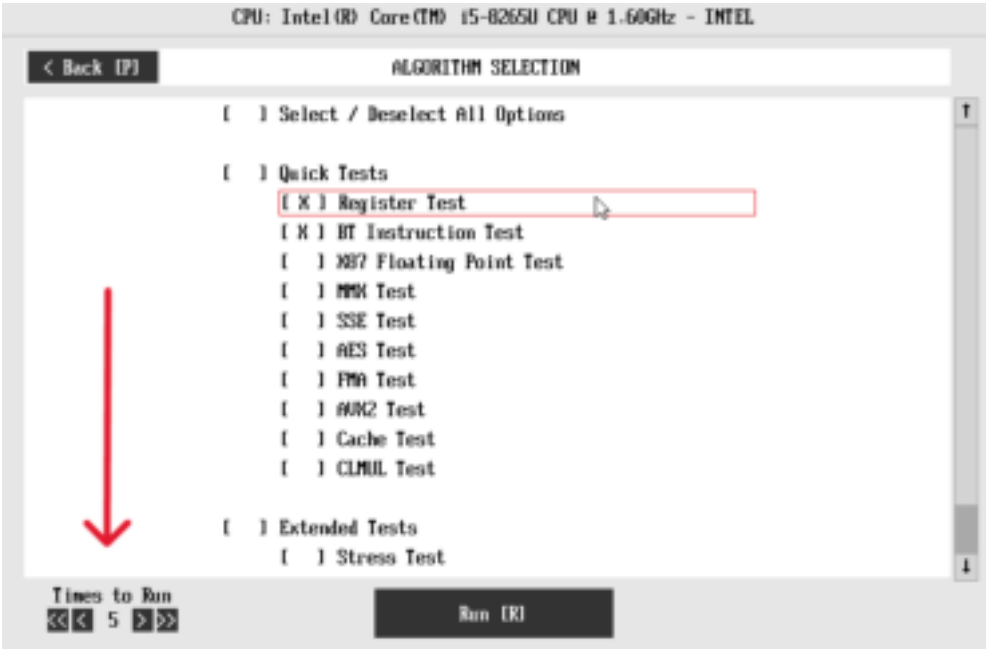

Figure 46: Times to Run - Module Configuration

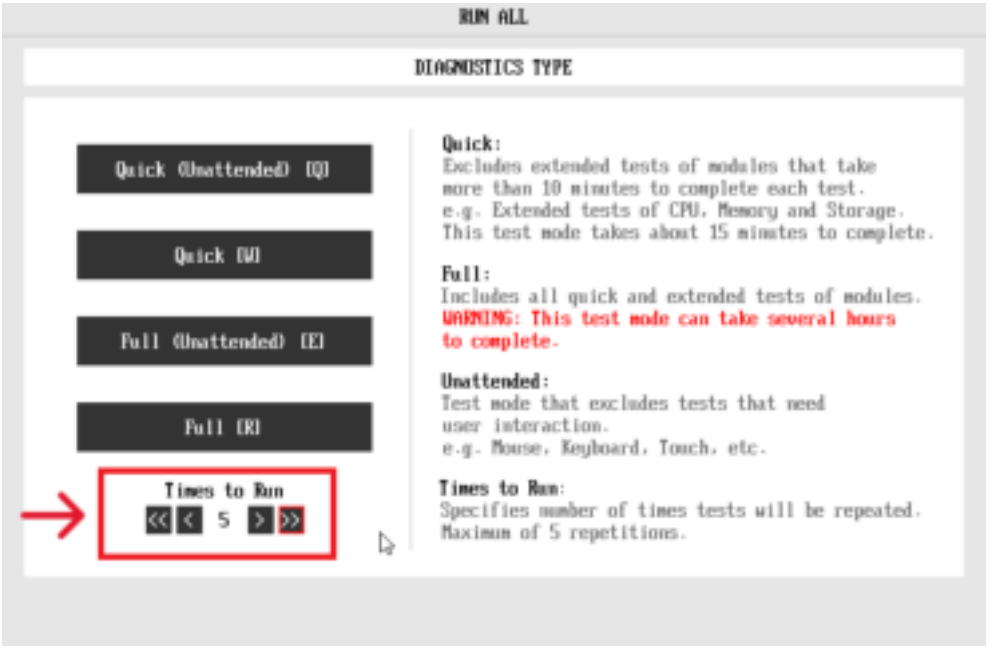

Figure 47: Times to Run - Run All Configuration

When the Times to Run is set to a value greater than one, the application will paginate the executions. Each execution of the suite will be displayed on a unique diagnostic screen, and each execution will have its own log and QR code.

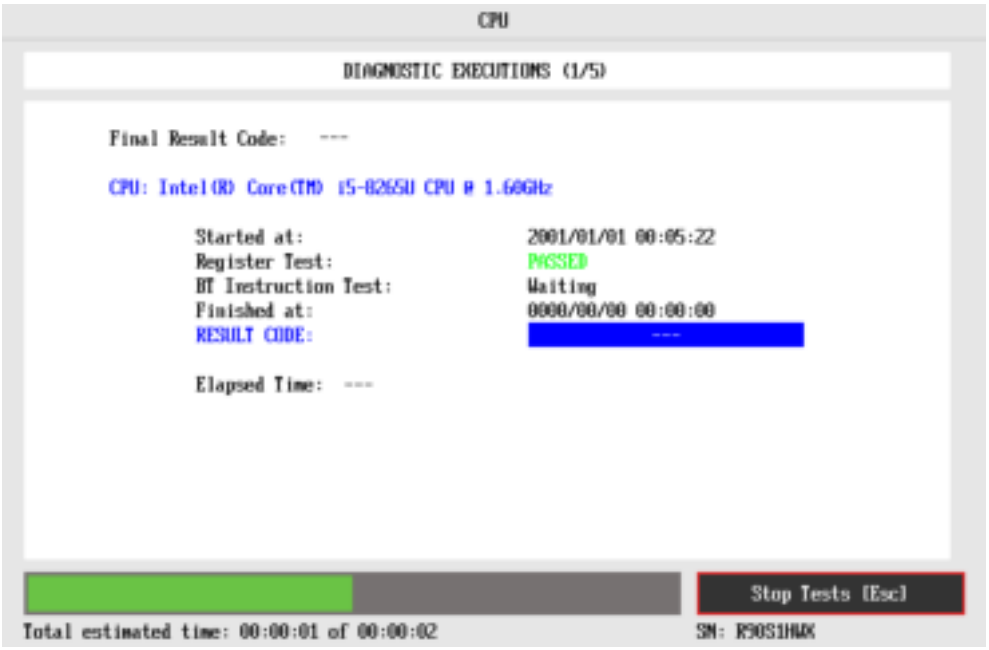

Figure 48: Times to Run Execution Screen

Users can navigate through the pages after the execution is complete using the **Next (N)** and **Back (P)** buttons to review the results of each run. On the log page, users can also browse through the logs for each execution.

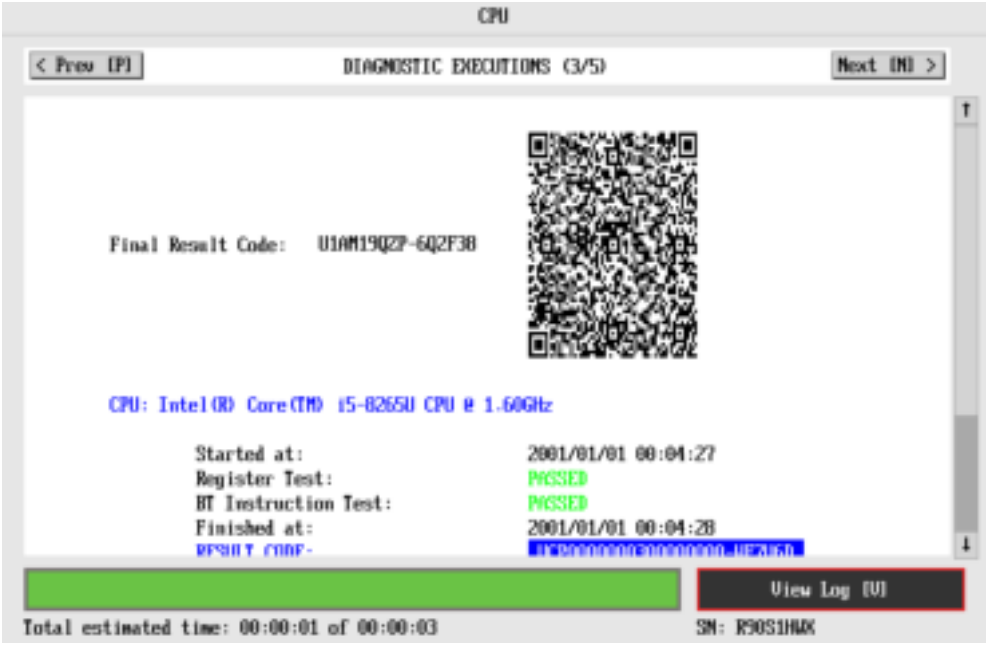

Figure 49: Times to Run Result Screen

Logs can be saved in two ways: users can save the specific log they are viewing to a file or export all concatenated execution logs into a single file. These options are available through the "Save" and "Save All" buttons, as shown in the placeholders below:

Figure 50: Times to Run View Log Screen

9.2 Screenshot Button

The Screenshot button is located in the application footer. Users can click the button or use the F11 shortcut to capture the current screen. This function saves the screenshot directly to the flash drive.

Figure 51: Screenshot Button

9.3 Help Button

The Help Button is a useful tool for users looking for guidance on specific information. This button is available on all screens of the application. For most screens, it mainly provides navigation tips, helping users understand how to move around the app, move focus, and select options.

Figure 52: Help Popup

In addition to navigation assistance, the Help Button also offers context-specific information tailored to individual screens. For instance, when users click the Help Button on the algorithm selection screen, they receive essential details that aid in executing diagnostics effectively.

Figure 53: Help Popup

9.4 Hierarchical diagnostics

When a specific module fails during testing, the following message/question will be displayed, prompting the user to respond with either "No" or "Yes":

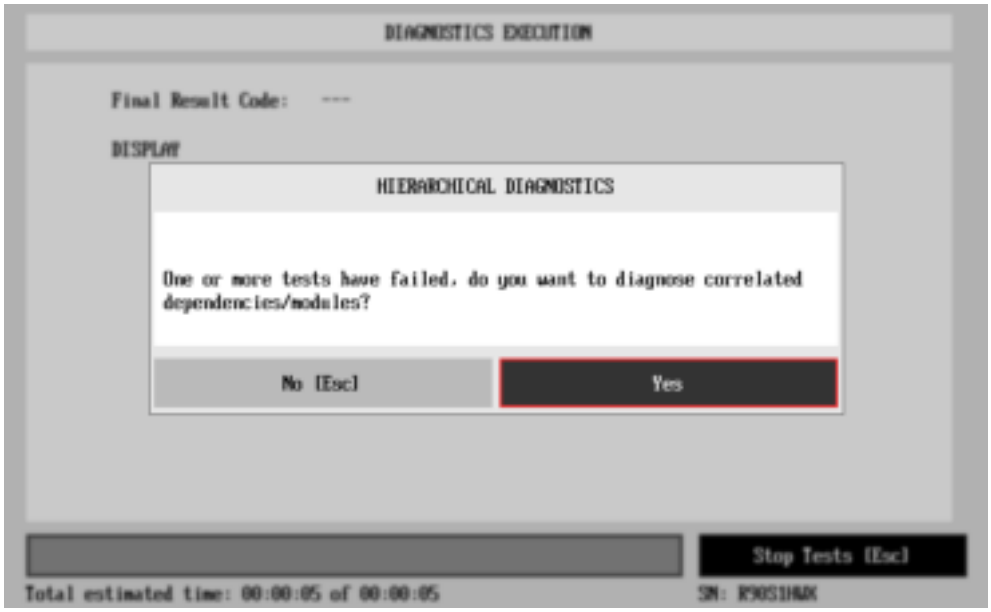

Figure 54: Hierarchical Popup

The hierarchical diagnostics feature executes tests in a structured manner, prioritizing more fundamental modules first. This functionality also facilitates the automatic identification of modules that may be the root cause of the failure when diagnosing a specific module. The example below illustrates the Hierarchical Diagnostics process for the Display Module:

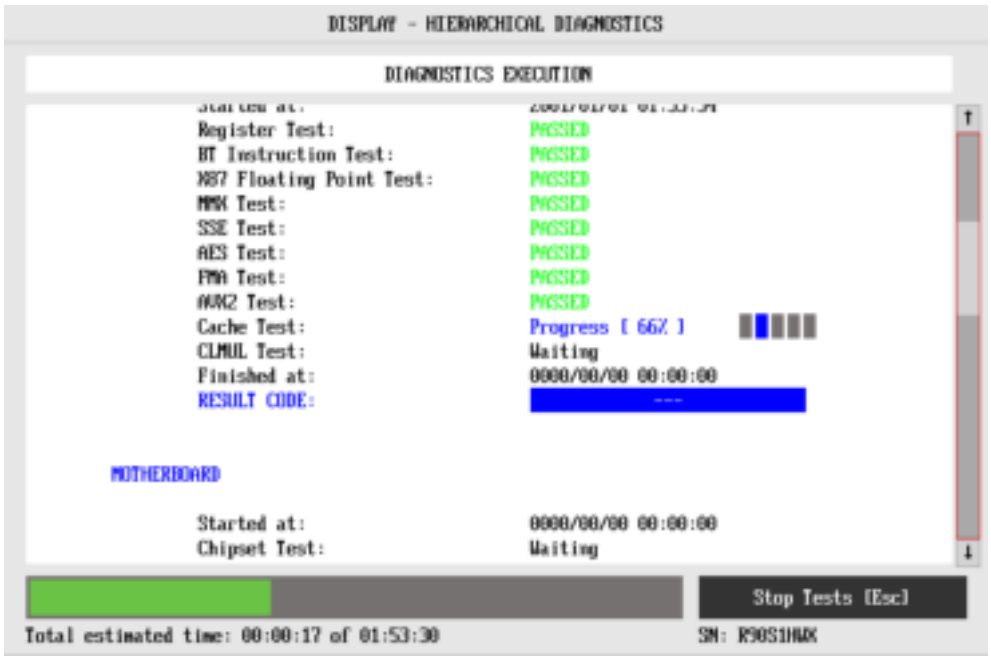

Figure 55: Hierarchical Execution

10 Exit the application

To exit the application, the user must select the option "Exit" on the Home screen and press the ENTER key. Then, the interface will be closed and the machine will be reset.

Figure 56: Button Exit

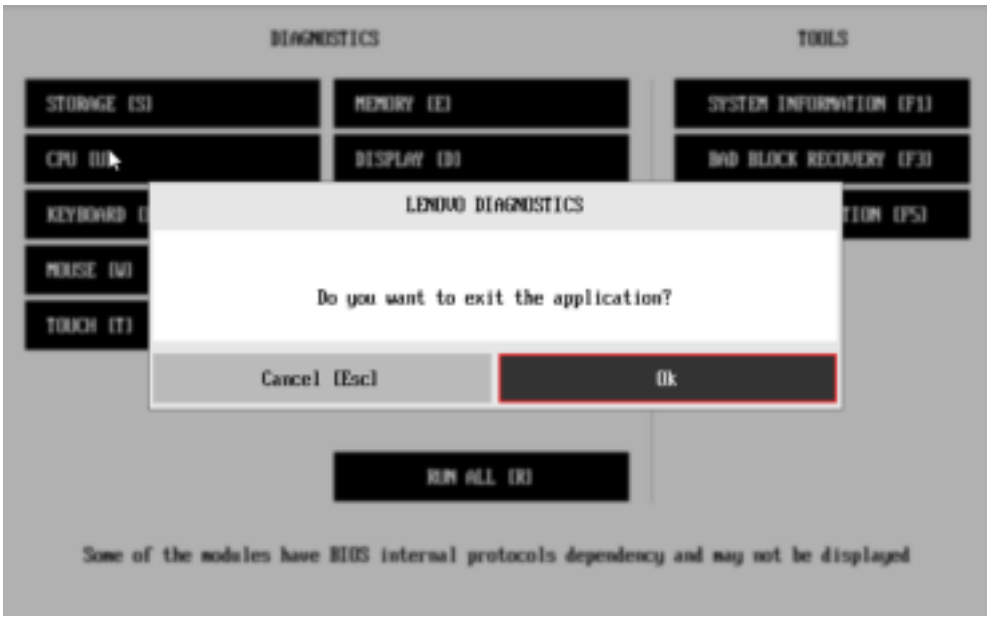

Figure 57: Exit Popup