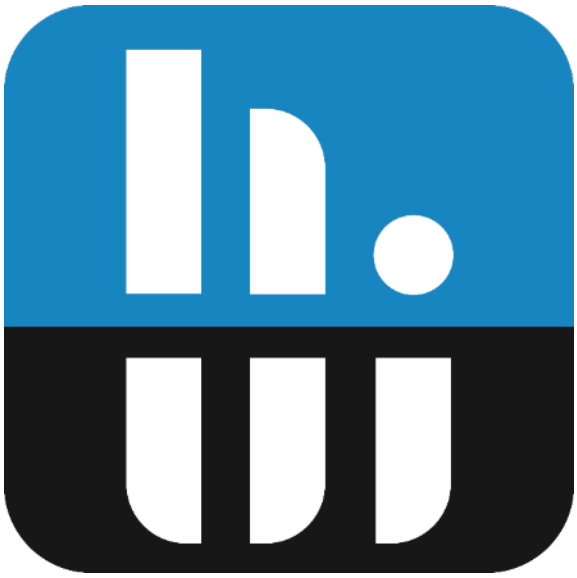


HWiNFO

Manual
For HWiNFO32 and HWiNFO64

Revision 1, 19th December 2023

Author: Martin Malík

Manual for HWiNFO32/64
Internet: www.hwinfo.com

Content

1. Introduction 5

2. System Requirements 6

3. Installation / Portable Version 6

4. Freeware Version vs. Pro Version 7

5. Program Start 9

5.1. Command Line Parameters (Pro Version) 10

5.2. Automatic Update (Pro Version) 12

6. Main Settings 13

6.1. General / User Interface 14

6.2. Safety 15

6.3. SMBus / I2C 16

6.4. Driver Management 16

6.5. License Management 17

7. Program Interface 17

7.1. Main Window 17

7.2. System Details (Main Window) 20
7.2.1. Central Processor(s) 20
7.2.2. Motherboard 20
7.2.3. Memory 21
7.2.4. Bus 22
7.2.5. Video Adapter 24
7.2.6. Monitor 25
7.2.7. Drives 25
7.2.7.1. ATA S.M.A.R.T. 26
7.2.7.2. ATA Device Statistics 27
7.2.7.3. NVMe Health Status 27
7.2.8. Audio 27
7.2.9. Network 28

7.2.10. Ports 28
7.2.11. Smart Battery 29
7.2.12. ACPI System 29

7.3. Reports 30

8. System Summary 32

8.1 Clock Window 34

9. Memory Timings 35

10. Sensor Status 37

10.1. Configure Sensors 43

10.2. Remote Sensor Monitoring 50

10.3. Custom User Sensors 51

10.4. Effective clock vs. instant (discrete) clock 53

10.5. Core Frequency reading in Windows 11 54

11. Shared Memory Support (Pro Version) 57

12. Add-ons 58

13. HWiNFO SDK 60

14. Support 60

14.1. HWiNFO Debug File 61

14.2. Forum & Community 62

15. Glossary 63

1. Introduction

HWiNFO is an all-in-one solution for hardware analysis and monitoring supporting a broad range of operating systems (DOS, Microsoft Windows 95 - Windows 11, WinPE) and platforms (i8086 - Xeon Platinum).

HWiNFO consists of reliable and highly accurate features that have made HWiNFO the renowned solution appreciated by many customers and companies today:

In-depth Hardware Information
From a quick overview unfolding into the depth of all hardware components. Always up-to date, supporting latest technologies and standards.

Real-Time System Monitoring
Accurate monitoring of all system components for actual status and failure prediction. Customizable interface with variety of options. Optimized for lowest overhead.

Extensive Reporting
Multiple types of reports, status logging and interfacing with other tools or add-ons.

Trusted and Reliable
Trusted by thousands of users including large enterprises, main hardware vendors, OEMs, ODMs and component manufacturers.
HWiNFO is designed to provide most accurate and reliable information.

Clean and Secure
HWiNFO doesn’t collect any personal information and doesn’t transmit any data over the network.

Frequent Updates
Updated frequently to support latest components, technologies and standards. Many systems are supported ahead of their launch.

2. System Requirements

HWiNFO requires relatively low system requirements, as no complex system suite is installed, but a compact and targeted application for selected usage scenarios. HWiNFO requires at minimum:

- Memory: approx. 50 MBytes (depends on the hardware configuration)
- Free disk space: approx. 10 MBytes
- Operating System:
  - HWiNFO32: Windows 95 or later. Not recommended for 64-bit systems
  - HWiNFO64: Windows XP 64-bit or later (any 64-bit Windows)
  - Windows PE/Windows RE
- Internet connection: only necessary for the automatic update function
- User account: Administrator rights required
- Other Software: it's not recommended to run other monitoring or tweaking software at the same time

3. Installation / Portable Version

HWiNFO can either be used as a portable package, where the ZIP file is simply downloaded and unpacked, and then the launch of the executable file starts the application.

The other option is a permanent installation on a target system, which is done through a downloadable installer. During installation the user can select the target location (default *C:\Program Files\HWiNFO64*) and the start menu folder (for example *HWiNFO64*) where HWiNFO will be installed. After installation the program can be launched either from Start Menu or from target directory on the installation drive. The installer automatically installs HWiNFO32 or HWiNFO64 depending on whether the operating system is 32- or 64-bit.

By providing both installation methods, the user achieves maximum flexibility – users that prefer an automatic installation with an installer or those who prefer to deploy HWiNFO manually on a portable medium (such as USB stick or external hard drive). In addition, the portable version in conjunction with the command line parameters of the Pro version allows automatic program launch with a login script during network logon to perform remote system diagnostics.

4. Freeware Version vs. Pro Version

Starting with HWiNFO64 version 7.00, a freeware version and a Pro version is available.

All versions are free to use in personal and non-commercial environments. Using HWiNFO64 in a commercial environment requires the HWiNFO64 Pro license. The following table contains the general function matrix:

	HWiNFO64	HWiNFO64 Pro	HWiNFO32	HWiNFO for DOS
Operating system	Windows XP and later (64-bit)	Windows XP and later (64-bit)	Windows 95 and later (32/64-bit) (*1)	MS-DOS or compatible
Non-Commercial use	✓	✓	✓	✓
Commercial use	✗ (*2)	✓ (*3)	✓	✓
Automatic report via command-line	✗	✓	✗	✓
Automatic sensor logging via command-line	✗	✓	✗	✗
Automatic Update	✗	✓	✗	✗
Shared Memory support	ⓘ 12-hour limit (*4)	✓ Unlimited	✓ Unlimited	✗
Remote Monitoring (max machines)	✓ (5) 12-hour limit	✓ (5 + n) (*5)	✗	✗

The following additional notes exist for this function matrix:

1 - Using HWiNFO32 on 64-bit systems is possible, but not recommended due to the lack of several advanced features, limited number of CPU cores supported and memory size limit.

2 - HWiNFO64 (non-Pro) can be used in commercial environment only for a 14-day evaluation period.

3 - HWiNFO64 Pro can be used in commercial environment only with the Engineer or Corporate License. The Personal license does not allow commercial use.

4 - HWiNFO64 (non-Pro) has a time-limited Shared Memory support to 12 hours. When this limit is reached during run-time, this feature will be automatically deactivated (and requires manual enabling again).

5 - Each HWiNFO64 Pro license increases the remote connection limit by the number of licenses. A Personal license by 5 computers, Corporate and Engineer by 1. For example, 2 Personal licenses will allow 5 + 2*5 = 15 connections. The maximum number of remote connections is currently 50. Remote Sensor Monitoring in non-Pro HWiNFO64 is limited to 12 hours as this feature is based on Shared Memory.

Furthermore, HWiNFO64 Pro is available with different types of licenses:

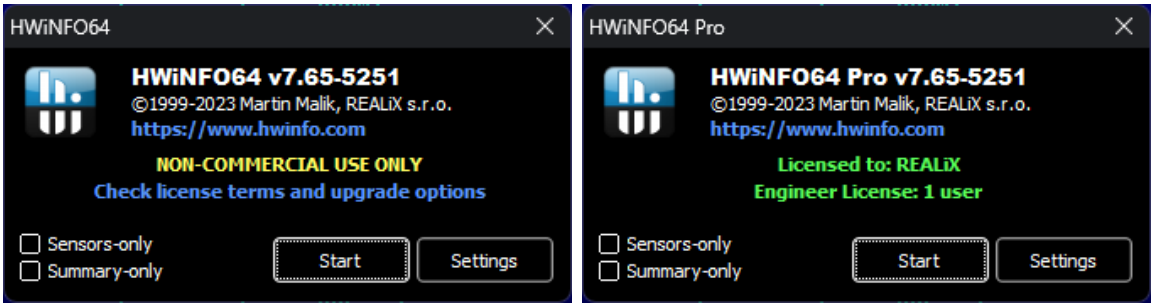
	Personal	Engineer	Corporate
Commercial use	✗	✓	✓
License seal	1 user, 5 computers	User-based	Computer-based
Volume Discount	✓	✓	✓

Any Pro version purchased doesn't expire and remains licensed perpetually. There are different types of maintenance licenses available: Subscription, or a Lifetime (Perpetual) license.

The subscription license includes 1 year of updates and already licensed versions won't expire. Renewal of the subscription license is available at a discounted price.

5. Program Start

During start of HWiNFO the User Account Control prompt for administrator access (UAC) is displayed. To provide accurate and extensive information, HWiNFO requires direct access to the hardware in many cases, which is only possible with administrative rights.



HWiNFO64 vs HWiNFO64 Pro

The initial (welcome) screen window includes the version used, license details, and offers three possible start modes:

- **Sensors-only** - starts HWiNFO and opens the Sensor Status directly, without detecting or displaying some system details that are not needed for this mode.

- **Summary-only** - starts HWiNFO and opens the System Summary directly without detecting or displaying sensor details.

- **None** - neither Sensors-only nor Summary-only: starts HWiNFO in the standard startup mode, which performs a full hardware scan and displays the main hardware window and summary window. In this mode the Sensor Status and System Summary windows can be opened from the toolbar in the main window.

The Sensors-only and Summary-only startup modes are optimized for faster startup time by skipping the detection of certain hardware. When HWiNFO is started in the Sensors-only mode, the main window cannot be reached as it requires a new full system scan. To show the main window, restarting HWiNFO with disabled Sensors-only mode is required.

The startup window contains two buttons:

- **Start** – starts HWiNFO in the selected startup mode.
- **Settings** - opens the settings window, which allows configuring several detailed options.

5.1. Command Line Parameters (Pro Version)

HWiNFO64 supports several command line parameters in the Pro version, where more advanced functions can be controlled and the program launch within a process is possible (for example by a batch file or a login script).

A possible scenario is, for example, an automatic report generation in the network, where HWiNFO64 is started within a login script with the command line parameter -r, -c, -x, -h or -m. The computer name can also be used for automatic naming of the report file name.

The following supported command line parameters are divided into the topics for automatic report generation, sensor logging and PresentMon.

<u>Automatically create a hardware-snapshot report file:</u>

-r create text report file
-c create comma-delimited report file (CSV)
-x create XML report file
-h create HTML report file
-m create MHTML report file (all images and icons are integrated)

Additional report parameters:

-utf8 use UTF-8 encoding (otherwise ANSI per current locale)
{comp_name} useable as filename and as substitution for computer name

Sensor logging:

-l automatically start sensors with logging. The following additional arguments are possible:

-log_delay=n	start automatic logging with a delay of n seconds
-max_time=n	stop automatic logging after n seconds
-poll_rate=n	polling period in milliseconds
-log_format=n	specify the log file format with n: default (no argument): value name in header; "sensor name" and "value name" at end 1 = single concatenated header with "sensor name|value name"

PresentMon:

-pm_proc_name=	specify process name to be monitored
-pm_proc_ex=	specify list of process names to be excluded separated by |

As can be seen from the parameters, a lot of possibilities exist, and more flexibility when combined with multiple parameters. Some examples are named below:

Example to automatically create a CSV report file:
HWiNFO64 -cMyLog.CSV

Example to automatically create a text report file with the computer name as filename:
HWiNFO64 -r{comp_name}.log

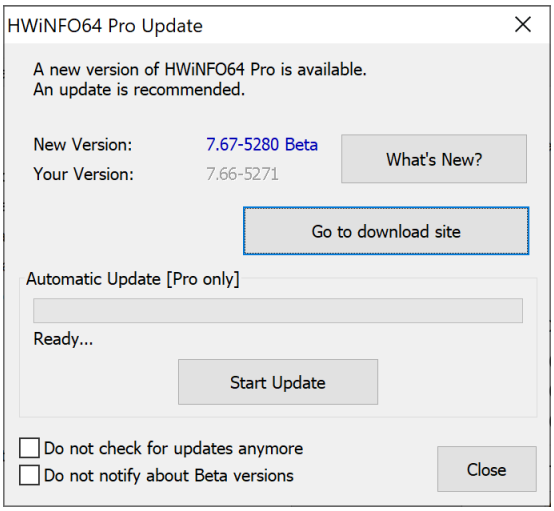
Example to automatically start sensor logging:
HWiNFO64 -lMySensors.CSV

Example to automatically start sensor logging in single header format:
HWiNFO64 -log_format=1 -lMySensors.CSV

5.2. Automatic Update (Pro Version)

The automatic update function is a convenient way to adapt to the continuously evolving IT market. HWiNFO is continuously updated with stable releases as well as beta releases. While the non-Pro version only notifies about new updates, the HWiNFO64 Pro can perform the whole update automatically by one click.

Checking for updates requires an internet connection. No data is transmitted outside the system during this process. Only a small file is downloaded from the server that contains information about the latest version available. Automatic update checking can be disabled in the settings dialog via the "Automatic Update" option. When enabled (by default), HWiNFO checks for new versions during program launch. Alternatively, the user can perform a manual update check directly from the program settings with the "Check for Updates" button.



If an update was found, the "HWiNFO64 Pro Update" window appears and the current and new versions are displayed. Furthermore, there are other buttons:

- "What's New?" - opens the HWiNFO64 website with the version history
- "Go to download site" - opens the website of HWiNFO64 with the downloads
- "Start Update" - downloads and installs the update in the Pro version. When finished, HWiNFO64 will restart itself to finalize the update process.

Additionally, there are 2 more options at the bottom of the window:

- "Do not check for updates anymore" - disables the automatic update function during program launch
- "Do not notify about Beta versions" - considers only full/stable versions for automatic update and not beta versions.

6. Main Settings

The main settings control the program behavior and various detection functions, and can be accessed either from the start dialog, the menu bar or the Windows notification area (system tray) located at the bottom right.

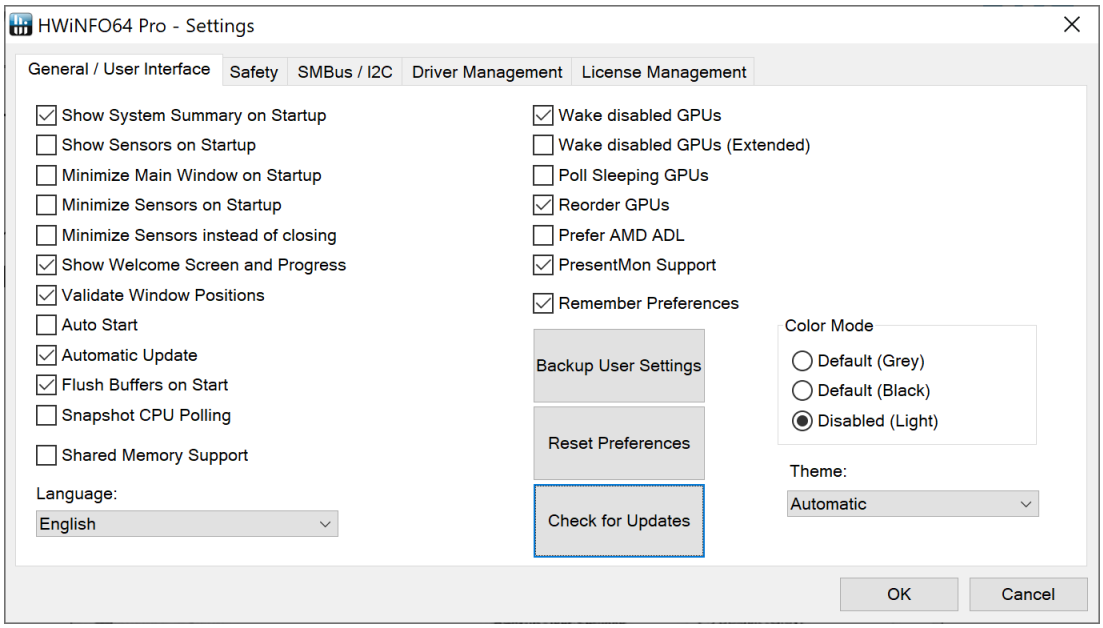
The various options are subdivided into tabs to achieve a thematic separation and better overview:

- General / User Interface
- Safety
- SMBus / I2C
- Driver Management
- License Management

As soon as a setting is changed, HWiNFO automatically creates the file HWiNFO32.INI for HWiNFO32 and HWINFO64.INI for HWiNFO64 in the program directory, which contains the portable program settings that differ from the default settings. The INI file can be exchanged if the user wants to load different program configurations at program start. These settings are considered portable, meaning settings that the user might want to use on other systems. Hence, they are stored in an easily accessible and editable (using a text editor) file that can be also deployed on portable media.

In addition, HWiNFO stores machine-specific settings in the Windows registry, such as the font, scaling, window positions and custom sensor layouts. HWiNFO32 stores this data in the registry key *HKEY_CURRENT_USER\SOFTWARE\HWiNFO32* and HWiNFO64 in *HKEY_CURRENT_USER\SOFTWARE\HWiNFO64*.

In the Settings window, these machine-specific settings can be exported from the Windows registry (via "Backup User Settings") and then imported to another target system by double-clicking the export file. Since these settings are machine-specific (for example depend on system configuration and sensors available), it's not recommended to port them to a different system.



This documentation will not describe every program setting in detail, because this is too extensive and HWiNFO already includes tooltips (shown when mouse is hovered over a particular setting) for each option, explaining the respective option in sufficient detail.

6.1. General / User Interface

This category contains settings that can be classified as general and user interface-specific. These include for example options to define startup appearance like automatically display the system summary and/or sensors at program launch, to automatically start HWiNFO with Windows (Auto Start), to enable automatic updates and shared memory support, set the language of the user interface and whether to permanently remember the preferences.

Automatic startup of HWiNFO is realized by adding a new task in the Windows Task Scheduler. This task can be modified by the user if different options are required or a startup delay is needed to work around some issues with too early startup.
There are buttons to "Backup User Settings", which will export all system- (or user-) specific settings into a registry file. "Reset Preferences" will reset all settings to default values. Furthermore, checking for updates can be disabled or color mode/theme can be changed.

When a "Default" color mode is selected, the mode used by HWiNFO depends on the Windows color mode. If Windows is set to use the Dark Mode, HWiNFO will also work in its own Dark Mode which can either use a Black or Grey color scheme. When Windows color mode is set to Light mode, HWiNFO will not be able to apply either of the Dark Modes available and will always work in Light mode.

6.2. Safety

The Safety category contains settings that are important for stability.

These include for example the control of scan for IDE/(S)ATA drives, whether S.M.A.R.T. and the ATA Statistics will be accessed, TPM should be checked, or certain technologies should be used/enumerated (such as CorsairLink, Intel Management Engine and oneAPI). These options are meant for advanced users and otherwise it's recommended to keep their default values. Disabling some of the following options can be useful in case of problems:

- EC Support
- CorsairLink and Asetek Support
- SW SMI
- Intel ME Support

It's not recommended to enable the "Low-level PCI Access" option especially on later Windows 10/11 systems that might restrict such type of access.
Changing the "IDE/(S)ATA Drive Scan" option from "Safe Mode" might be useful only on some legacy systems with IDE controllers. Otherwise, this option should only be changed to "Disable Scan" in case of problems during disk/drive scan.

Important and necessary in case of problems or technical inquiries is the Debug Mode. When enabled, HWiNFO will create an additional dump file with the extension DBG during runtime. This file contains detailed technical system data which are useful for analyzing problems when provided to HWiNFO's support. This file is not meant to be analyzed by end-users and doesn't contain any personal data. It's not recommended to run with Debug Mode enabled during normal operation as it increases the system load due to retrieving and dumping of additional data.

6.3. SMBus / I2C

The SMBus / I2C category contains expert settings for the System Management Bus, on which some sensor devices and the SPD-EEPROMs of memory modules are located. Individual SMBus devices can be excluded from access, which should only be used in rare cases.

GPU I2C support can be used to detect sensors from graphic cards, and exclude specific GPU I2C bus addresses, as well as control interfaces such as NVAPI and/or AMD ADL.

6.4. Driver Management

The Driver Management category contains control options for the HWiNFO kernel mode driver and displays its current status.

This driver is automatically installed at program launch and uninstalled during the program end when the "Persistent Driver" option is disabled. In this category, a manual installation/uninstallation can be performed, and the driver can be installed as persistent – it will remain installed also after closing HWiNFO.

6.5. License Management

The category for license management contains details about the currently used HWiNFO license, including the status, licensee, type, the maintenance expiry and the location of the license file.

Furthermore, the user can import a license by pasting the unmodified license string or the license file.
Online management of license information, subscriptions and retrieval of license files can be performed via the 2Checkout (Verifone) myAccount:

https://secure.2co.com/myaccount

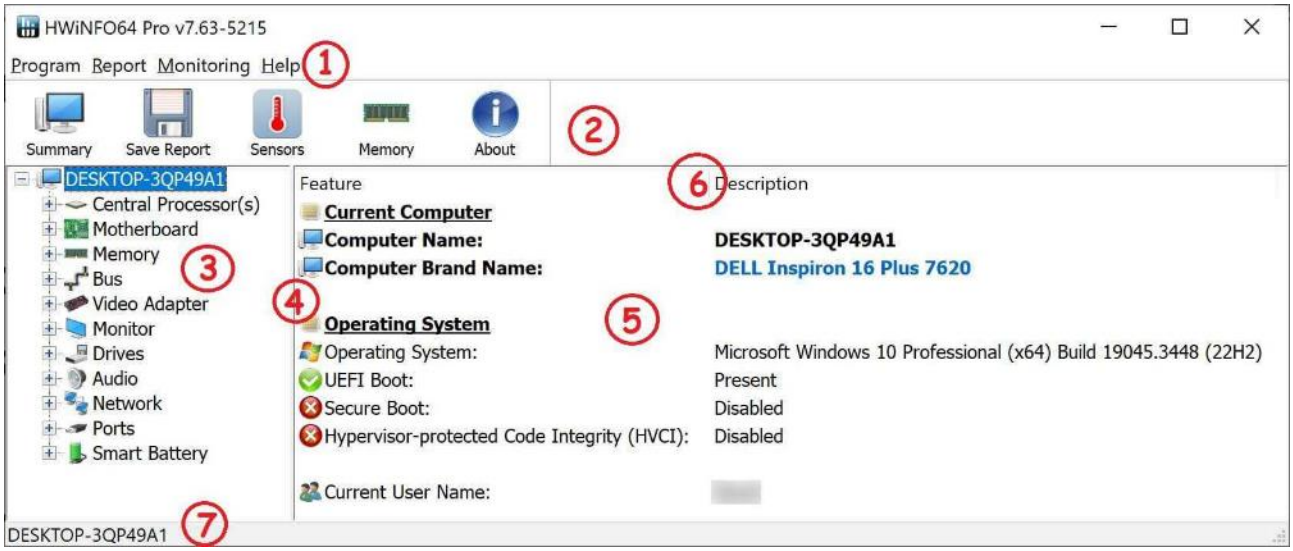
7. Program Interface

The program interface is clearly and intuitively designed into 3 main windows (categories):

- **Main Window**: Contains static information useful for inventory or system snapshot
- **System Summary**: Contains the most important hardware information presented in a compact form
- **Sensors**: Lists all hardware-related parameters and sensors that are dynamic and might require constant monitoring.

7.1. Main Window

The main window consists of several elements, which allow a clear navigation. This is because HWiNFO detects countless system details, which must be sorted, grouped and displayed thematically. The following sample image contains the compact display of the program interface, and each component is marked with a number, that is discussed directly after the image:

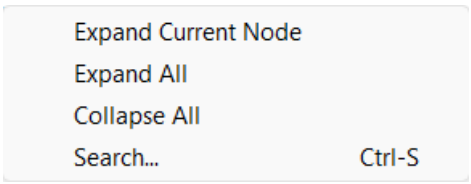


At the top is the menu bar (1), which contains all program functions. Note that the menu might be hidden by default in Dark Mode. Pressing the Alt key will make a hidden menu bar visible.

Below the menu is the toolbar with icons for the most important core functions (2) like Summary, Save Report, Sensors, Memory and About.

The central area of the window represents the main navigation, which contains a tree on the left (3), divided into topic groups - these can be expanded and collapsed. In the middle there is a vertical line (4), called splitter, that can be moved with the mouse to the left or right. This allows the display to be adjusted.

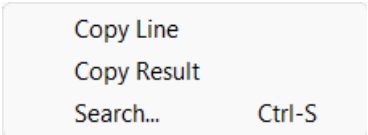
Right-clicking an item in the left (tree) area will display a context-menu allowing additional actions over an item (expanding or collapsing the current or all nodes) or performing a filtered search:



Depending on which category and which entry is selected on the left, HWiNFO reports the corresponding results in the right window area (5), which is again divided into two columns - *Feature* and *Description*.

These two columns also contain a splitter (6) by pressing with the mouse on the vertical line in the column header and shifting the column widths to the left or right while holding the mouse button down.

Right-clicking an item in the right pane area will display a context-menu allowing additional actions like copying the entire line or only the Description part into Windows Clipboard, or performing a filtered search:



At the very bottom of the HWiNFO window is the status bar (7) which displays the level within the tree that is currently activated and displayed.

7.2. System Details (Main Window)

The following subchapters describe which system details are detected in the individual topic groups. Since these system internals are highly dependent on several factors such as system configuration and Windows installation, every single detail can't be described, but instead a sufficient overview is given.

7.2.1. Central Processor(s)

This section lists each central processor (CPU) package that is physically present and detects extensive details for each processor. This includes not only the processor name and the original processor frequency, but also further details like stepping, code name, manufacturing technology, platform/socket, microcode update revision (MCU), dozens of feature flags, overclocking features, core statistics. The latter describes how many physical and logical cores are present, for later Intel processors also the Performance (P-Cores) and Efficient cores (E-Cores).

The overall details are divided into subsections in order to achieve a clear grouping and to be able to find certain details more quickly. A distinction is made between *General Information*, *Operation Points*, *Overclocking*, *Cache and TLB* as well as extensive *Feature Flags*.

The list and display differ greatly from processor to processor and thus system configuration.

7.2.2. Motherboard

The motherboard section contains information about the motherboard including chipset (North-, South- Bridge, PCH, FCH), slots (PCI, A.G.P., PCI-X, PCI Express), current BIOS as well as information about the LPC (Super-IO) chip and Trusted Platform Module (TPM). Some systems might also list some chipset-specific details here.

Furthermore, the three subcategories *ACPI Devices*, *SMBIOS DMI* and *Intel ME* (if available) exist in this section so that further system details are grouped thematically.

ACPI Devices:

ACPI devices are shown here with the assigned resources such as I/O port, IRQ and memory location.

SMBIOS DMI:

The system tables found in the System Management BIOS are detected and displayed here, whereby this data is defined by the BIOS and motherboard manufacturer and is not always complete or meaningful, especially on older systems. SMBIOS was originally established to better manage the system configuration in networks and to provide a uniform standard for the storage and evaluation of this. A distinction is made between various system tables for BIOS, mainboard, system enclosure, processor(s), caches, cooling/temperature devices, onboard devices, TPM, slots, ports and memory devices.
The scope and accuracy vary greatly depending on system configuration and as this data is entered with varying degrees of completeness by the manufacturer.

Intel ME:

Here additional details about the Intel Management Engine and its various sub-services are displayed, if this technology is available. These are subsections for the host state, capabilities, firmware capabilities and firmware feature state.

7.2.3. Memory

The Memory section contains general memory details such as the total memory size, current performance settings and various timing parameters. A subsection is also created for each memory module found, and the data of the SPD-EEPROM (Serial Presence Detect) is evaluated within it.

SPD has been an established technology for over 20 years, in which the BIOS reads the configuration parameters from a separate EEPROM chip on the memory module during POST (Power-On Self-Test) and configures the system accordingly. These configuration parameters are stored in a standardized form and displayed by HWiNFO in this section.

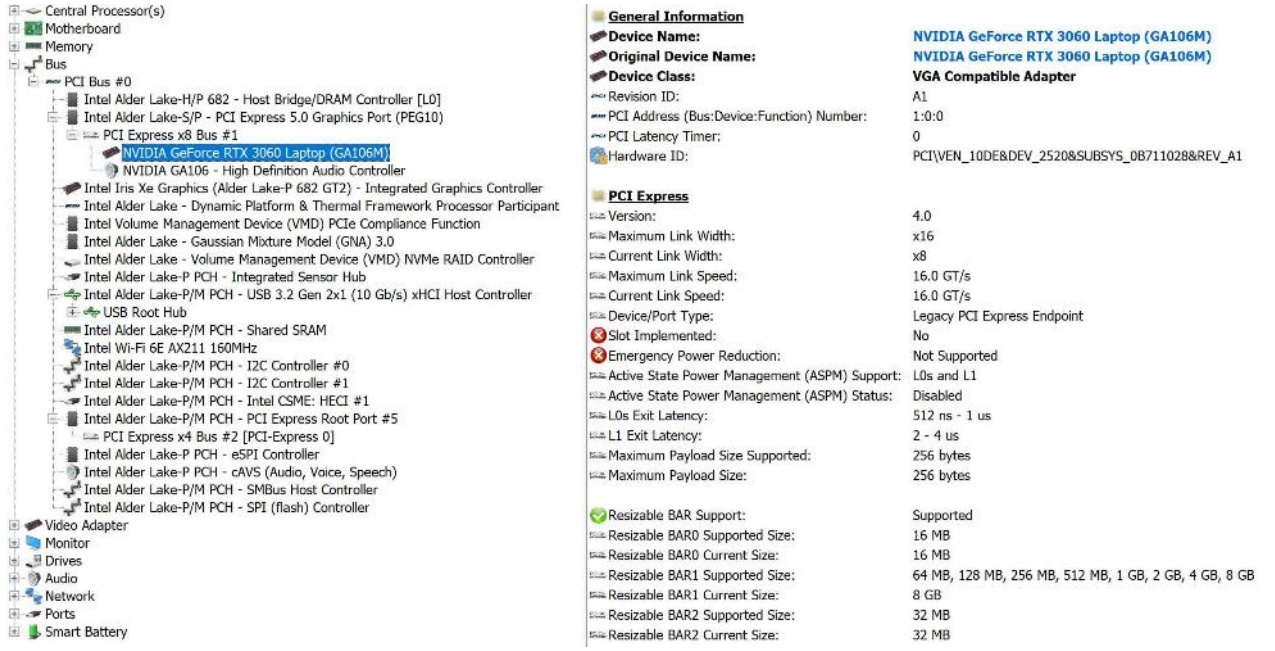
Depending on the memory technology and generation (DDR3, DDR4, DDR5, …), there are various subsections such as general module information, module characteristics, voltage levels and timing parameters. Each memory module comes with standardized (JEDEC) set of timings and some might also include additional sets like XMP (Intel), AMP or EXPO (AMD). These optional sets provide additional profiles with timing parameters for higher performance.

Note, that on some systems (notebooks or servers) it might not be possible to read all or any DIMM SPDs due to proprietary locking mechanisms. Also, LPDDR4/LPDDR5 systems don't feature SPDs as the memory chips are soldered-down on mainboard (DIMM modules are not used).

7.2.4. Bus

The bus section detects all buses and subsystems belonging to the overall topic category of the PCI bus and displays them in a hierarchical structure.

The PCI bus as the original version has been extended over the years with various subsystems, such as Mini PCI, PCI-X, CompactPCI, AGP and PCI Express - all of which are based on the same communication protocol. The PCI bus has a tree-like structure where a device on bus (port) can split the lanes and form a parent of a new sub-bus. This hierarchy is accurately shown by HWiNFO and might be useful to understand the relations on bus and to determine whether devices/ports share available lanes.



HWiNFO enumerates all devices on the bus and displays different topic groups for each PCI device. These are *General Information*, *System Resources*, *Features* and *Driver Information*.

The general information section contains various basic details such as device name and class, revision ID, PCI address number (Bus:Device:Function), PCI latency timer and the hardware ID. The system resources section deals with the resources used by the respective device. These can be interrupt lines and pins, as well as memory and I/O base addresses.

The features section contains details about whether the device has Bus Mastering enabled, operates at 66 MHz (legacy) and supports fast back-to-back transactions.

The PCI device details are completed by the corresponding driver information including manufacturer, version and date.

For PCI Express devices there is additional information displayed that describes the maximum and current link width and speed. The maximum parameters describe the device capability, and current values the actual negotiated parameters between the device and its bridge. Another important parameter especially for GPUs is the Resizable BAR Support which is shown in detail for each of the Base Addresses (BARs) for devices that support this feature. This parameter is also shown under the GPU node for the appropriate framebuffer BAR.

If the device is an USB host controller, this controller is enumerated and connected USB root hubs are shown with the USB devices connected to them. More details about the USB device display can be found further on in the subchapter for ports.

7.2.5. Video Adapter

The video adapter section (legacy name) contains a list of graphic cards (GPUs) or integrated graphics adapters (iGPUs) and reports extensive details for each card.

Various topic sections are available and named as *Video Chipset*, *Video Card*, *Performance* and *Driver Information*.

The subsection for video chipset describes the graphic chipset with its graphic memory and the subsection for video card contains details about the video bus, GPU type and video BIOS (VBIOS).

The subsection for performance contains clock frequencies for various parts of the GPU (engines), as well as configuration information such as the number of shaders, cores, ROPs, TMUs. Some GPUs might show additional interesting details here.

Details about the used graphic driver with version and date complete this section.

7.2.6. Monitor

The monitor section contains a list of the connected monitors and extensive details about each monitor.

The details are grouped into topic sections, which include *General Information*, *Advanced parameters*, *DPMS Modes*, *Supported Video Modes* and *Driver Information*.

The general information section contains the monitor's name, its serial number, as well as the date of manufacture and the vertical/horizontal dimensions. The advanced parameters section contains information about the input signal, video interface (like for example DisplayPort, HDMI, DVI...) and the gamma factor.
Power saving capabilities are detected in the section for DPMS mode, as well as video modes supported by the monitor in the following section.

All monitor details are completed by the corresponding driver information including manufacturer, version and date.

7.2.7. Drives

The Drive section contains all detected hard disk drives with extensive details and fitness information about S.M.A.R.T. (Self-Monitoring, Analysis and Reporting Technology).

The details are grouped by the drive type *(S)ATA/ATAPI* and *NVMe*. Within a recognized drive there are the topic sections for *General Information*, *Capabilities*, *S.M.A.R.T.* and *Device Statistics*.

The general information section contains the drive/host controller, drive model, serial number and the capacity. The capabilities section contains device properties like dataset management.

In further sections for S.M.A.R.T. and Device Statistics many other reliable and interesting drive details are detected, which are described in more detail in the following two subchapters.

7.2.7.1. ATA S.M.A.R.T.

S.M.A.R.T. stands for *Self-Monitoring, Analysis and Reporting Technology* and allows a fitness evaluation of drives, which is performed by the drive controller/firmware and HWiNFO reports all these details. This includes whether the drive has exceeded a critical temperature threshold, or the device reliability has been degraded.

Each drive parameter that is monitored is called an "attribute" and each attribute is assigned a unique number. It is important to note that **S.M.A.R.T. was never fully standardized**, which means that each vendor is allowed to define its own attributes and assign them a specific meaning. Only a handful of attributes are more-less used by all vendors. This fact results in a difficulty of correctly understanding and reporting the meaning of many attributes, because each vendor is using different meanings even for its different drive series. Moreover, the interpretation of additional data provided by attributes (for example Total Reads/Writes) can vary between drive models, hence it's difficult to accurately report in all cases. HWiNFO tries to match this as best as it can by using model-specific adjustments for attribute meanings, but a 100% accurate coverage cannot be expected here.

HWiNFO reports the S.M.A.R.T attribute values in the following format:
[nn] Attribute name: vv / tt, Worst: ww

- nn = Attribute number in hexadecimal format.
- vv = Actual Parameter value.
- tt = Threshold value. "Always OK" means there's no threshold defined.
- ww = Worst value observed during lifetime.

The logic of S.M.A.R.T. is that if any value falls below a threshold defined (by vendor), it means a failure.

Of interest might be parameters like "Number of Reallocated Sectors" or "Off-Line Uncorrectable Sector Count" which indicate the number of so called "bad" sectors on drive. If these parameters start to show non-zero values and the count increases over time, it is highly recommended to perform regular backups and consider drive replacement.

7.2.7.2. ATA Device Statistics

The ATA Device Statistics are available for some (S)ATA drives and describe drive details of a statistical nature in several topic blocks.

These include for example fitness values of the device history such as the power-on hours, logical sectors written/read during lifetime, the number of write/read commands, total amount of reallocated sectors, mechanical start errors and reported uncorrectable errors.

Temperature statistics include values like the current temperature, average and lifetime short-term/long-term temperatures, and elapsed time during under- and over- temperature.

Unlike S.M.A.R.T. (which is prone to misinterpretation of attribute meanings), the ATA Device Statistics is a **fully standardized protocol** which is reporting accurate information and thus is preferred by HWiNFO.

7.2.7.3. NVMe Health Status

NVMe drives do not support a S.M.A.R.T. feature like ATA drives, instead they report a different set of health information which is not based on custom attributes and is fully standardized. This provides a slightly smaller set of health information, but this set is fully reliable.

7.2.8. Audio

The Audio section detects all existing audio devices with extensive details.

All details are grouped into the categories *Audio Adapter*, *High-Definition Audio Codec* and *Driver Information*.

The audio adapter section contains the name of the audio device (controller) including its hardware ID. The AC'97 or High-Definition Audio Codec section detects the name of the audio codec(s).

All audio details are completed by the corresponding driver information including manufacturer, version and date.

7.2.9. Network

The Network section detects all existing network devices, which includes hardware-only devices such as network cards (NICs) and Wi-Fi controllers.

The details are grouped into *General Information*, *Capabilities* and *Driver Information*.

The general information section contains the name of the network device as well as the manufacturer and the MAC address. The capabilities section detects the maximum link speed, transmit/receive buffer size and the hardware ID.

All network details are completed by the corresponding driver information including manufacturer, version and date.

7.2.10. Ports

The Port section detects various existing interfaces with details about them.

While for serial and parallel ports their presence is simply displayed, HWiNFO groups USB ports according to the detected USB host controllers. These USB controllers are hierarchically subdivided with the corresponding root hubs, and the ports per USB hub.

For detected USB devices on a port the details are grouped into *Device Information* and *Driver Information*.

The device information section contains the most important device information like manufacturer and product name, serial number, supported USB version and device speed as well as the hardware ID.

All USB device details are completed by the corresponding driver information including manufacturer, version and date.

7.2.11. Smart Battery

The Smart Battery section detects existing batteries on portable devices with various details about them.

The details are grouped into *General Properties* and *Current Power Status*.

In the general properties section, the basic details like device / manufacturer name, serial number / unique ID, chemical composition, capacity and the wear level are detected. The latter represents the difference between the maximum factory capacity of the battery and the current maximum possible charge capacity (which decreases over time).

The current power status section contains the current power source, current capacity and voltage as well as the discharge rate.

7.2.12. ACPI System

When the option *ACPI AML Enumeration* is activated in the main settings, this topic group appears at the bottom of the topic list.

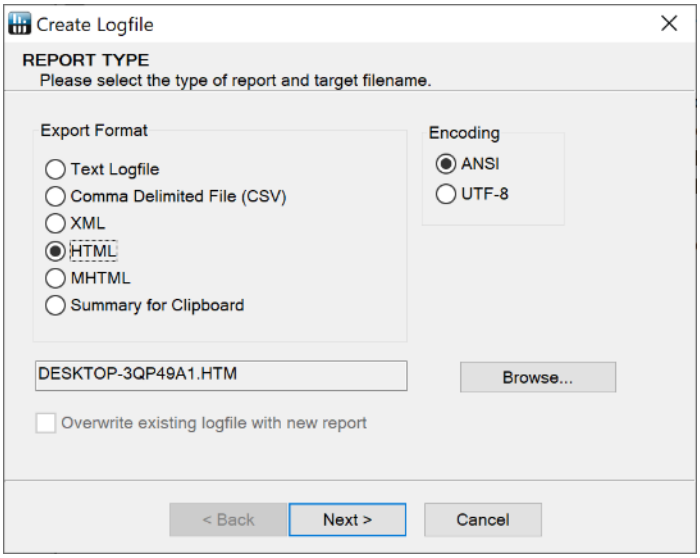
The list enumerates the ACPI AML namespace, starting with the system root, which can contain various structures, objects, names, methods, mutexes, states, events and aliases.

These details are primarily relevant for developers because they require a deeper knowledge of the ACPI specification.

7.3. Reports

Since HWiNFO recognizes a lot of system details and displays them on the screen, a powerful report function exists for further exporting, archiving or post-processing.

This function can be started either from the main menu or by clicking the toolbar icon called *Save Report*.



As first, the export format needs to be selected from a list and the following formats are supported:

- Text Logfile
- Comma Delimited File (CSV)
- XML
- HTML
- MHTML and
- Summary for Clipboard

While the text logfile describes a simple text format, a CSV file allows further processing, for example with spreadsheet programs, because the columns are divided by a separator. XML as Extensible Markup Language allows the display in a hierarchically structured form, and HTML as base of the Internet a comfortable display in the browser - the latter is preselected as the default format. An extended MHTML variant allows the embedding of graphics and symbols in a single HTML file, which would normally have to be supplied additionally. Therefore, with MHTML the user has a single archive file per report and achieves a true-to-original report archiving.

For the encoding the two standards ANSI and UTF-8 are available. ANSI is an 8-bit encoding, which is very limited to encode other languages and its interpretation depends on the option *Current language for non-Unicode programs* in the Windows Regional Options. UTF-8 is a variable-length universal and widely adopted encoding.

The final step is to specify the file name to which the report is written. By default, HWiNFO uses the current computer name with a file extension depending on selected export format. The file name can be entered manually or specified in the file selection dialog using the Browse button.

The Next button takes the user to the Report Filter selection, where the detectable system details are listed in a hierarchical structure, each subdivided into groups with the sensors at the very bottom. Here the user can either keep the settings and the most important information are exported comprehensively - or can adjust the individual options by fine-tuning.

The report is exported and saved with the Finish button.

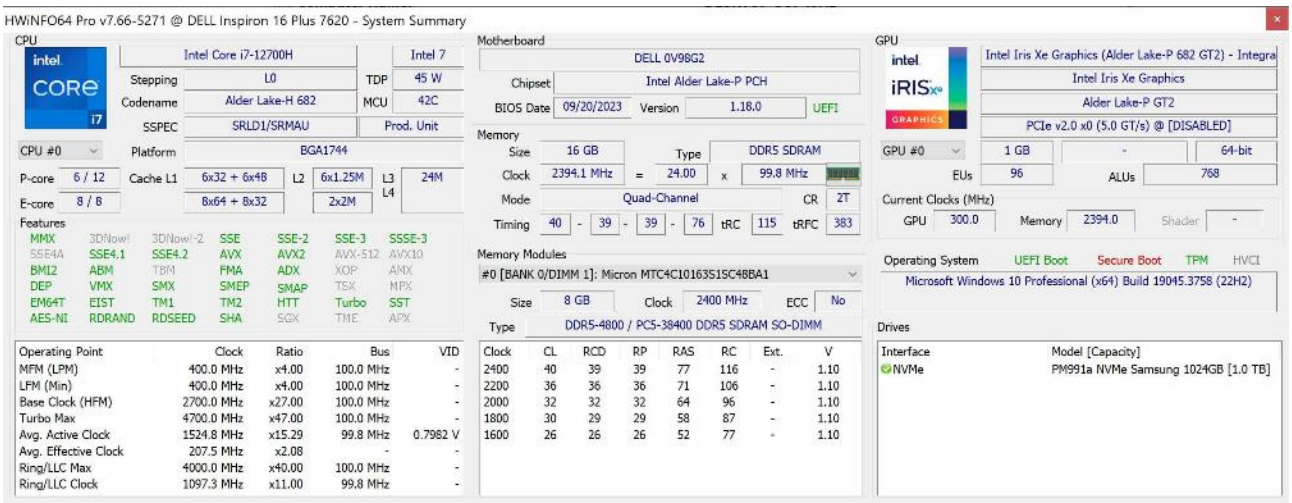
A special feature here is the export format selection "Summary for Clipboard", which creates an extremely compact report displayed in a text dialog. This special report is suitable for the need of a very compact summary of the system configuration and can be copied into the Windows Clipboard.

8. System Summary

The system summary is a clearly structured and compact overview of the system configuration, which is shown by default after program launch.

This window can be opened either from the main menu (*Summary icon*) or from the system tray icon. The System Summary is not available in the Sensors-only mode as it requires detecting components that are excluded in the mode optimized for Sensors-only.

The most important system parameters are displayed in real-time using additional visual components such as processor and graphics card logos.



The window is divided into three columns and an additional window containing the active clocks for each physical processor.

The left column of the system summary contains details about the selected central processor package ("CPU #n" combo box), which include the name/codename, Thermal Design Power (TDP), available cores (divided into performance and efficiency cores for later Intel processors), processor capabilities as well as various clock frequencies (minimum, base and maximum turbo clock). The clocks are divided into columns for Operating Point, Clock, Bus ratio, Bus clock (BCLK) and Voltage ID (VID). Individual CPU features are displayed using the following colors:

- Green – Feature present and enabled
- Red – Feature present but disabled (in BIOS or OS)
- Grey – Feature not present

The middle part contains information about the motherboard including chipset and BIOS. Below is the memory configuration displayed which consists of two parts – dynamic information like current clocks and timings which can change during run time, and static configuration showing available per-DIMM timings as defined by DIMM SPD. The DIMM combo box allows selecting the module for which to report this information.

The right column contains details about the selected (via "GPU #n" combo box) GPU - including chipset, bus interface, graphic memory and clocks, number of shaders, cores, ROPs (Raster Operation Pipelines) and TMUs (Texture Mapping Units).

Beneath the GPU is the operating system displayed including build number, and a list of drives including model, interface capability (not actual rate) and capacity. Next to each drive is a green icon shown if there are no errors/failures reported, or a red icon if any of the S.M.A.R.T., Device Statistics of NVMe health status reports a failure. When no icon is displayed it means the health status cannot be determined.

8.1 Clock Window

While the System Summary windows is active, the Clock Window for selected physical processor (package) is also displayed. Here the individual cores (divided into Performance and Efficiency cores for later Intel processors) are displayed graphically in real time with their actual clock. The additional information about clock and the ratio based on the bus clock belongs to each core. A bar indicates the current clock with respect to clock limits. The color of the bar doesn't indicate a problem but the actual clock level and has the following meaning:

- Green – Clock is below Base Clock (HFM)
- Yellow – Clock is above Base Clock but below Turbo Clock
- Red – Clock is above Turbo Clock

For Intel processors with per-core Digital Thermal Sensors (DTS), the core number can have a different color depending on its current thermal status:

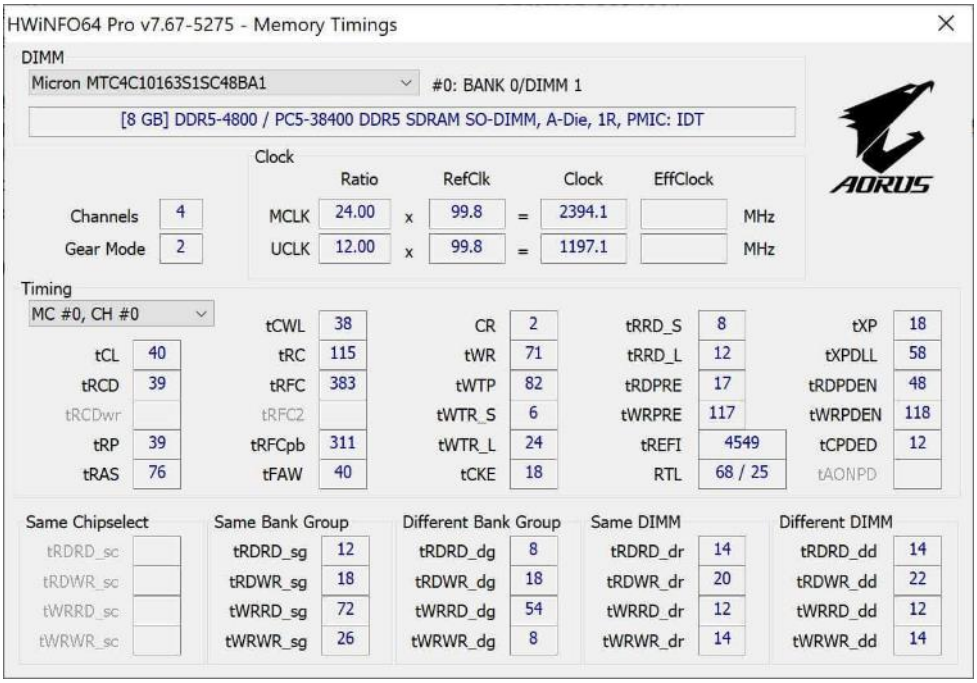
- Green – the core's temperature is currently below thermal throttling limit
- Red – the core is currently throttled due to temperature exceeding the limit

Another feature of the Clock Window is to display either the Active or Effective Clock. This mode can be toggled by clicking inside the window. The current mode of clock reporting is displayed in the window title. For further details see chapter 10.4 for the Effective clock vs. instant (discrete) clock.

9. Memory Timings

The Memory Timings window is a clearly structured and compact display displaying various timing parameters for installed memory modules.

This function can be started either from the main menu, the Memory icon in toolbar, from System Summary or from the system tray icon.



While the respective memory module is selected via the selection list at the top, the basic details such as DIMM size, type, speed, form factor, die revision (if available), number of ranks and PMIC vendor (for DDR5) are shown in the box below.

This is followed by clocks and timing details, whose definition is explained in detail in the respective memory specification. Some of the most important parameters include:

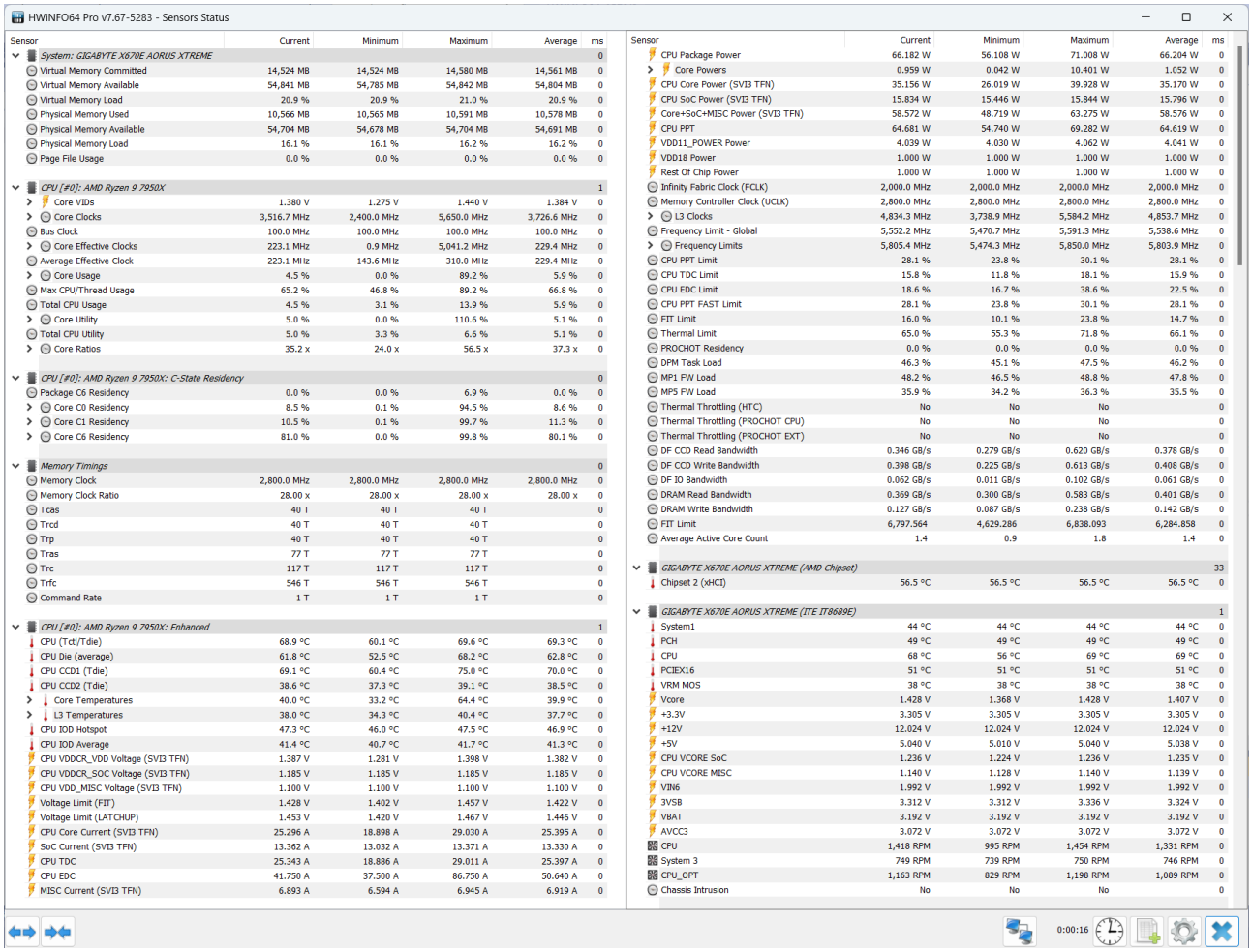
- CL: CAS Latency - is the number of cycles between sending a column address to the memory and the beginning of the data in response
- RCD: RAS to CAS Delay - is the minimum number of clock cycles required between opening a row of memory and accessing columns within it
- RP: Row Precharge Time - is the minimum number of clock cycles required between issuing the precharge command and opening the next row
- RAS: Row Active Time - is the minimum number of clock cycles required between a row active command and issuing the precharge command
- RC: Row Cycle Time - is the sum of RAS + RP
- RFC: Row Refresh Cycle Timing - is the number of cycles to refresh a row on a memory bank
- CR: Command Rate - is the number of cycles between, when a DRAM chip is selected and a command is performed
- Gear Mode: is technology used for overclocking and can often be enabled within the BIOS. The gearing defines the ratio between memory controller (IMC) and main memory in the form of Gear 1 (1:1) and Gear 2 (1:2)

10. Sensor Status

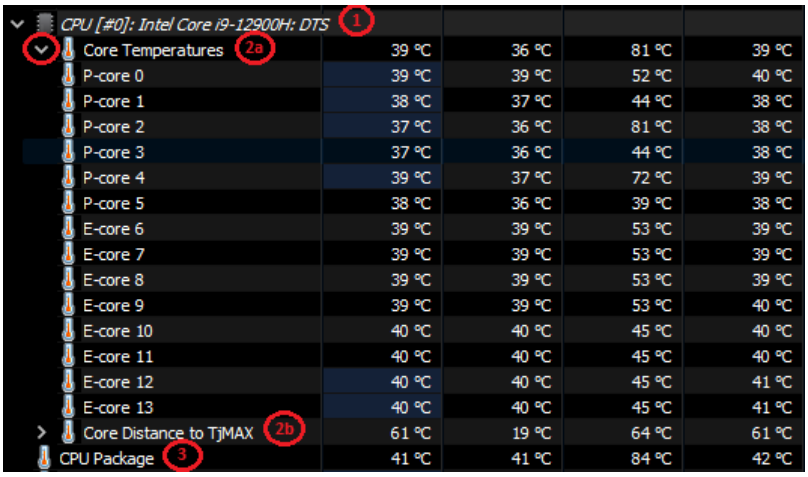
The sensor engine of HWiNFO is one of the most important and complex functions and combines many unique capabilities that make HWiNFO so sophisticated and powerful.

A lot of different sensor data are detected and displayed, additionally a very flexible configuration allows customization to own requirements.

The sensor status can be started from the welcome screen, with the setting *Sensors-only* directly at program launch, or during the program runtime from the menu bar or the icon *Sensors* located in the toolbar.



In the default layout, all detected sensors are displayed and arranged into topic groups. These groups contain thematically summarized sensors, for example for categories like System, CPU, Memory Timings, Memory DIMMs, S.M.A.R.T., Drives, GPU, Battery, Network and Windows Hardware Errors (WHEA).



The above picture shows the different types of sensors grouping:

1. **Sensor group**: Can contain multiple items (sub-groups; temperatures, voltages, fans, etc.). Each sensor can be folded into a single row (showing the heading only) or unfolded into a full list by clicking the arrow to the left of its icon.

2. **Sub-group**: Lists values of the same type (for example core temperatures). Each sub-group can be folded (2b) into a single row or unfolded (2a) into a full list by clicking the arrow to the left of its icon.
   Note, that the meaning of values in columns (Current, Minimum, Maximum, Average) is slightly different for sub-groups. The heading of a sub-group shows the following values in each column:
   a. Current: Actual average value of all aggregated current values (not all-time average)
   b. Min/Max: Run-time minimum/maximum value of all aggregated values
   c. Average: Run-time average of aggregated current values

3. **Sensor Value**: A single sensor value

Each sensor is by default displayed with the columns *Sensor*, *Current*, *Minimum*, *Maximum* and *Average*. The Current value shows the actual sensor value of the last readout, the Minimum/Maximum values show the run-time minimum/maximum values read since the HWiNFO sensors window was opened (or since the Min/Max values were reset).

Sensor data (readouts) are updated periodically at every tick of the global polling period which is set to 2000 milliseconds by default.
If an actual sensor value cannot be read (due to a sensor failure), the value is displayed in grey color and shows the last successful value read.

For each sensor, a ToolTip is displayed when the mouse is hovered over an item. This is shown to help understand the vast amount of available sensor values and their meanings. It can be switched off in the sensor settings.

At bottom of the window there are several buttons that lead to further sensor functions:

- Expand window by adding a new table:
  Create a new table with sensor data to the right of the current table.
- Shrink window by removing rightmost table:
  Deletes the rightmost table and thus reduces the size of the window.
- Manage network connections for remote monitoring:
  Opens the Remote Center for network monitoring, please see separate subchapter in this documentation.
- Reset min/max/average values and timer:
  Resets all minimum, maximum and average values of the sensor table to the initial state as well as the sensor run time indicator at the bottom.
- Logging start:
  Allows to define a CSV log file and start sensor logging. Once activated, the same button allows to stop the logging session.
- Configure sensors:
  Opens the sensor settings window for configuring the sensors, please see separate subchapter in this documentation.
- Save settings and close:
  Saves all settings and closes the sensor window (or completely exits HWiNFO, if started in the *Sensors-only* mode*)*.

The bottom window area contains a run time counter, which starts counting at the moment the sensor status was opened.

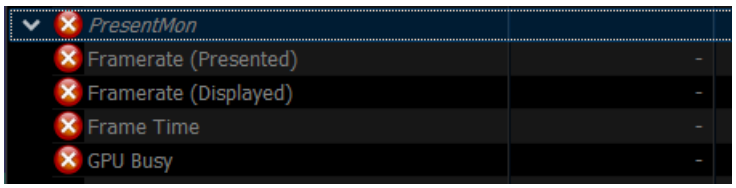
Clicking a sensor item will select it, holding the Ctrl or Shift key during selection allows selecting multiple items. Several actions can be performed with such selection. The following keyboard shortcuts are defined:

- Del - disable monitoring of all selected sensors
- Ins - enable monitoring of all selected sensors
- Shift-Del - hide all selected sensors
- F2 - rename selected sensor

Additionally, clicking and holding the mouse button down while moving allows placing a sensor item into a different position, similar to drag-and-drop. This requires disabling the Fixed Order setting which is described in more detail in the sensor settings section.

Note, that disabling the monitoring of a sensor group (its heading) will also disable monitoring of all its associated sensor items. In some cases when there is a need to disable monitoring of a sensor (for example to avoid unnecessary or undesired polling), it might not be sufficient to disable monitoring of all sensor subitems but also to disable the heading (sensor name).

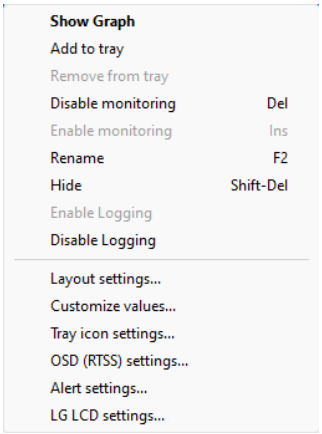
The picture below shows a completely disabled monitoring of sensor including its heading ("PresentMon") and its subitems ("Framerate", …):



The red X icon indicates that monitoring of the sensor item is disabled.

When a sensor is hidden, it won't be shown in the list. Such hidden sensors are listed in sensor settings. The *Layout* tab contains a list of *Hidden items* and those items can be made visible again from there.

By selecting a sensor or a sensor group, a context menu is shown via the right mouse button, which can be used as a shortcut to further actions. These include sensor or sensor group-specific functions as well as the general sensor settings, which can also be accessed in sensor settings.



For sensors, a context menu appears with the following entries:

- Show Graph: Opens a new window with a graphical display of the sensor. This is also the default action when double-clicking an item.
- Add to tray: adds the sensor to the tray (Notification Area)
- Remove from tray: removes a previously added sensor from the tray
- Disable monitoring: disables monitoring for the sensor
- Enable monitoring: enables monitoring for the sensor
- Rename: renames the sensor
- Hide: hides the sensor
- Enable logging: enables including of the sensor in sensor (CSV) log
- Disable logging: disables including of a sensor in sensor (CSV) log

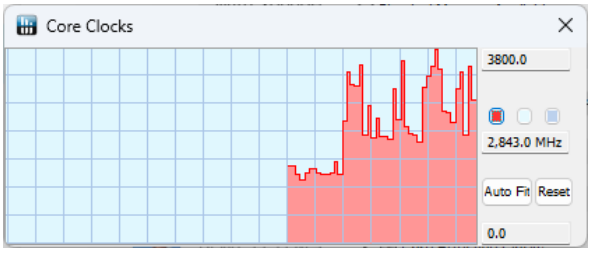
For sensor groups, a context menu appears with the following entries:

- Disable monitoring: disables monitoring for the whole sensor group
- Enable monitoring: enables monitoring for the whole sensor group
- Rename: renames the sensor group
- Hide: hides the whole sensor group
- Enable logging: enables including of the sensor group in sensor (CSV) log
- Disable Logging: disables including of a sensor group in sensor (CSV) log

For each sensor and sensor group there are also entries below a separator line to display the corresponding tab of the Sensor Settings with a preselected corresponding sensor/sensor group:

- Layout settings
- Customized values
- Tray icon settings
- OSD (RTSS) settings
- Alert settings
- LG LCD settings

Any sensor value can be displayed in its own graph by double-clicking the respective item:



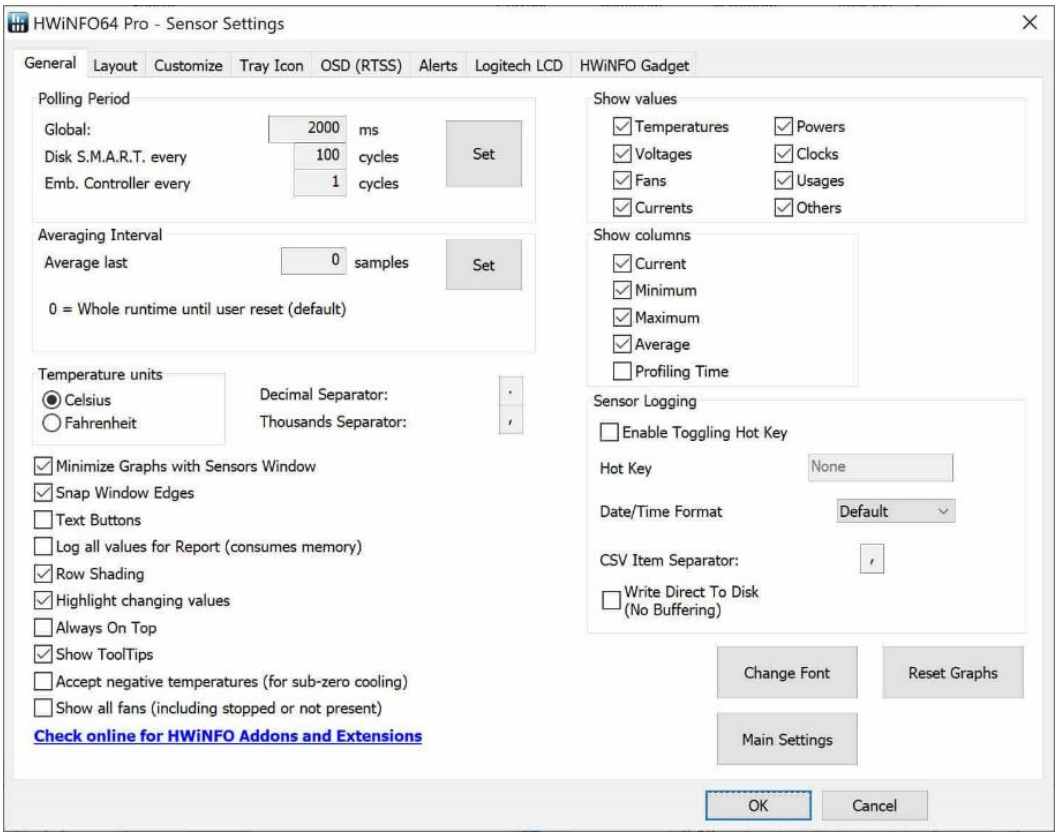
Each graph window can be resized and snapped to other graph or window corners. The right side of graph window contains the following controls:

- Minimum (at bottom) or Maximum (at top) value of the scale. These values can be changed, and the graph will automatically adjust its scale after a change
- Current value (in the middle)
- 3 buttons to change: series color, graph background color, grid color
- "Auto fit" button: When enabled, the graph will automatically adjust the min/max values and scale based on values read during run time
- "Reset" button: to reset the graph

The area with controls on right side of the graph window can be hidden by clicking inside the graph area. This will extend the graph area to the entire window. This can be reverted by clicking again inside the graph area.

10.1. Configure Sensors

This subchapter describes which configuration options are available for sensors, how remote sensor monitoring and custom user sensors works and which features are important.



The sensor status is extensively configurable via the sensor settings and allows a fine tuning of the many sensor values and their appearance.

Due to the large number of settings, these are grouped into tabs at the top of the window:

- General
- Layout
- Customize
- Tray Icon
- OSD (RTSS)
- Alerts
- Logitech LCD and
- HWiNFO Gadget.

General:

This category contains the control of polling periods, filter values to show (Temperatures, Voltages, Fans, Currents, Powers, Clocks, Usages and Others), which columns to show (Current, Minimum, Maximum, Average and Profiling Time), change temperature units (Celsius or Fahrenheit), sensors logging and other general settings.

Enabling the Profiling Time column will show a new column in the sensors window that displays the time (in milliseconds) needed to read (poll) the entire sensor group and its particular values. The sensor group row shows the total time needed to read the entire group. This might be useful when diagnosing sluggish sensor appearance which might be caused by some sensor being blocked.

HWiNFO reads data from all sensors at each tick of the Global Polling Period. This is done for all sensors except a few specific ones where polling the sensor can have a negative impact on system performance or other side effects. In such case, a less frequent polling might be desired. Such sensors are:

- Disk S.M.A.R.T. sensors
- Embedded Controller (EC) sensors

For the above sensors it's possible to set the polling at each n-cycles. For example, if the Global Polling period is set to 2000 ms and Disk S.M.A.R.T. polling set to every 100 cycles, it means that the S.M.A.R.T. sensor will be polled every 2000*100 ms (or 200 seconds or 3.3 minutes).
Note, that the Disk S.M.A.R.T. sensor polling period is valid for ATA drives only, it doesn't apply to NVMe because there were no adverse effects observed related to polling of NVMe drives.

Other settings include for example whether all values should be logged in the report, if values that changed from previous readout should be highlighted (different background color in the Current column), should be highlighted, if the sensor window should always remain on top, or if ToolTips should be shown. Some settings that might require more detailed description:

- "Accept negative temperatures (for sub-zero cooling)": Some sensors might return negative temperatures when they are not connected. By default, HWiNFO considers such values invalid and won't show them. If this setting is enabled, such values will be accepted and shown as a negative temperature.
- "Show all fans (including stopped or not present)": In many cases it's not possible to determine if a fan is connected to a sensor input and the fan speed sensor returns 0 RPM when the fan is currently not spinning or not connected at all. To avoid reporting several "dummy" 0 RPM values for fans not connected, HWiNFO uses the following logic when this option is disabled (default):
  - When a fan speed sensor initially reports 0 RPM, the fan is not displayed in the sensor window.
  - As soon as the sensor reports a valid non-zero value, it will be shown in the list and remain shown even if it reports again a zero value.
  - This logic will be reset when the sensor window is closed.

  When this option is enabled, all fan speed sensors will always be shown regardless of whether they report 0 RPM.

When using the sensor logging function (CSV log), the same decimal and CSV separator should not be used as that will cause mismatch in the CSV logfile. A correct CSV separator depending on post-processing tool used (for example Microsoft Excel) should be chosen. In most cases this is either a comma "," or a semi-colon ";".

Three different buttons can be used to change the sensor font, reset the graph positions and reach the main settings window.

Layout:

Here the layout of sensor items can be customized and for each sensor it can be configured whether monitoring should be enabled and if it should be displayed. If the setting of a sensor is changed so that it is no longer shown (same to hide sensor), HWiNFO moves it to the list of hidden items located at the bottom. The *Restore Original Order* button will restore positions and appearance of all sensor values to the default layout. This can be useful when there's a mismatch in layout.

HWiNFO by default displays the sensors in a fixed order, which can be overridden by disabling the *Fixed Order* option. Disabling the fixed item order will give full control over sensor item placement. If disabled, HWiNFO will no longer automatically maintain the order all sensor items, which allows moving sensor values to any position or under a different sensor heading. If a new sensor/value appears during runtime, it will be placed at the end of the list instead of the respective sensor, because in this mode HWiNFO doesn't maintain binding between sensor item and its group.

Buttons with arrow keys are available for changing the item order, which can be used to move a sensor up/down (one line at a time) and all the way up/ down (to list top / bottom). The button with an underline dash to the right can be used to insert a blank line.

Customize:

In this section, each sensor or value can be customized - changing the label name, font color and properties (bold or italic), displaying a value in red color or treating a value invalid if a specific threshold is undershot/overshot. The unit (for example MHz, %, °C, V, RPM etc.) can also be renamed. Per-sensor thousands separator or the number of decimal digits can be changed. Any value can also be adjusted using a custom multiplier or offset. To subtract a value, enter a negative value in the "Add (+)" field. To divide a value, enter an inverse number in the "Multiply" field (for example to divide by 1000 enter 0.001).

With the *Logged* setting at the bottom right the user can include or exclude individual sensors for logging.

Tray Icon:

The tray icon function is mainly used to keep important and frequently used sensors permanently visible in the tray. Any sensor can be displayed in the tray (Windows Notification Area) as an icon, and these settings can be defined in the *Tray Icon* tab.

Once a sensor has been enabled with the setting *Show in Tray*, various display and behavior settings are available. These include the foreground and background color (or transparency), the font used to render the icon text. Further options allow setting a custom divisor, or truncation of digits in icon, and which type of sensor value should be displayed (Current, Minimum, Maximum or Average).

The *"Add All Items To Tray"* button will put all sensors into tray, but should be used with caution because too many sensors might exceed the maximum number of icons that can be displayed and Windows might then display an error message.

OSD (RTSS):

In the tab *OSD (RTTS)* sensors can be displayed to an on-screen display of the RivaTuner Statistics Server.

When RTSS is installed and running, HWiNFO will show the current OSD status as *"Running"* and for later RTSS versions that support graphs, this will also be indicated in the status. In this case, the user can define for each sensor whether the value and/or label should be displayed in OSD. Units can be displayed in superscript, as well as only the raw value (without units), or alignment for digits can be specified.

Each value must be assigned to a specific position in OSD (line and column). If multiple items are assigned on the same line, they will be separated by a column. For graphs the size and margin of the graph in OSD can be specified.

The OSD output can be switched on/off via a keyboard shortcut for which a hotkey can be specified.

Alerts:

In the *Alerts* tab different alarms can be enabled for each sensor which will be triggered by predefined conditions.

For this purpose, the alarm function needs to be first enabled by selecting the corresponding sensor and enabling the option *Enable Alerting*.

An alarm can be configured to be triggered when the sensor falls below or exceeds a custom threshold value.

Several actions can be combined with triggering of an alarm. These include for example playing of a sound, launch of a certain application (with optional arguments), displaying of a warning window or showing the item's value in the sensors window in red color. An alarm condition can also be logged into a text file, which might be useful for later analysis of when/which alarm happened.

To avoid too frequent triggering of alarms, it's also possible to limit the frequency of successive alarms triggered using a specific minimum notification distance. This setting determines the minimum time that needs to elapse before the next alarm will be triggered.

It's also possible to specify a minimum number of successive samples necessary to trigger an alarm. This might be useful for cases when a sensor might occasionally raise an alarm which is not critical. For example, specifying 3 samples means that at least 3 successive below/above threshold conditions need to happen for the alarm to be triggered.

Another option allows limiting the alarm to be triggered only once, so that when successive sensor reads exceed the threshold, the alarm won't be triggered again.

Logitech LCD:

HWiNFO directly support showing sensors on a Logitech keyboard featuring an LCD (Black-white or QVGA). This can be configured in the *Logitech LCD* tab.

When a corresponding keyboard is available and its drivers are installed, HWiNFO shows the status when the connection has been established. In this case the user can define for each sensor if the value should be displayed on the LCD.

It's possible to specify on which screen, line and column the output should be shown, and which font size and color should be used.

It's also possible to specify which type of value should be used: Current, Minimum, Maximum or Average.

HWiNFO Gadget:

In the *HWiNFO Gadget* tab, sensors can be displayed in the Windows Sidebar. This was part of the operating system from Windows Vista and was renamed to *Windows Desktop Gadgets* in Windows 7, but then no longer included in future Windows.

There are some 3rd party solutions like the *8GadgetPack* that emulate the sidebar, so this option can be used also with later Windows versions without native sidebar support.

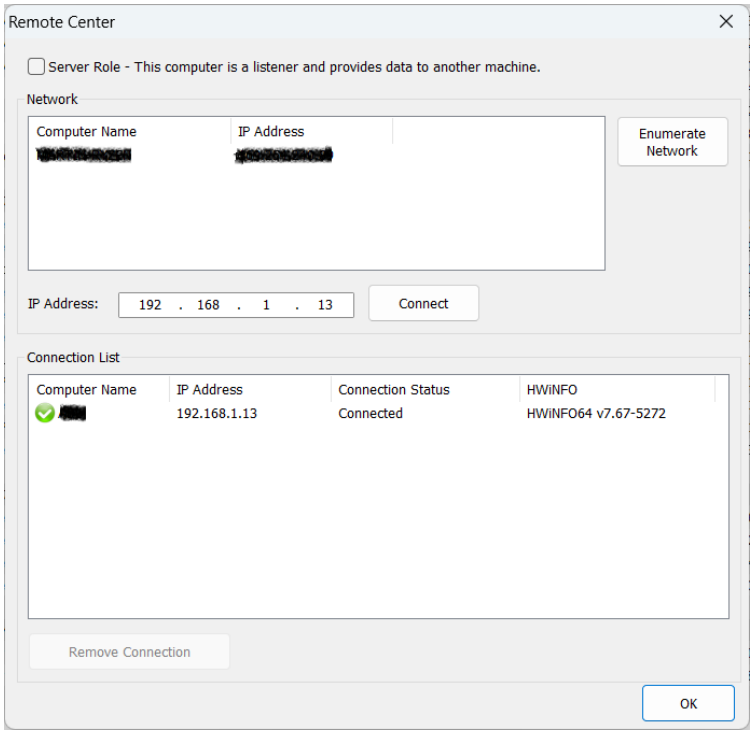
As first, this functionality needs to be globally enabled, then any sensor can be individually selected for reporting in the gadget.

This feature outputs all sensor values into the registry from where the respective gadget can read them. For this purpose, HWiNFO also reports the index of each value exported.

10.2. Remote Sensor Monitoring

HWiNFO is also able to display sensors from remote computers via a feature called *Remote Sensor Monitoring*.

Remote sensor monitoring for more than 12 hours (same as the Shared Memory Limit) requires the Pro version of HWiNFO64 with appropriate number of licenses. HWiNFO needs be started on each computer to be monitored (hereinafter referred to as slave computers). The button *Manage Network Connections for Remote Monitoring* opens the *Remote Center*, where each slave computer can be enabled for a server role (will act as a listener and provide data to another machine). Enabling this option might trigger a Windows Firewall message where the connection needs to be allowed.



On a computer from which the monitoring is to be performed (hereinafter referred to as master computer), the user can also switch to the *Remote Center* and enumerate computers on the network (button *Enumerate Network*) or enter the target IP address of the slave computers manually.

The connection is then established by pressing the *Connect* button and the *Connection List* below will populate all connected slave computers and also display respective HWiNFO versions. Any established connection can be disconnected by pressing the *Remove Connection* button.

For remote monitoring, the *Sensor Status* window on slave computers must be opened (can be minimized), then sensor values of those slave computers will be displayed inside the *Sensor Status* of the master computer. Sensors from remote computers appear in the sensor list of master computer with the machine name in prefix of each sensor group.

When the *Sensor Status* window on a slave computer is closed while the connection is established, the HWiNFO instance of the master computer adjusts the display and sets all sensor entries to gray - this means that the last status is displayed, but it's not possible to read actual data.

With an appropriate configuration, the sensors from a maximum of 300 slave computers can be monitored on the network.

10.3. Custom User Sensors

Since version 6.10, a unique feature has been introduced in HWiNFO with ability to display custom user-defined sensors in the sensor status window. This allows users with basic programming skills to display various sensor values, which is especially useful when using custom devices (for example to push data from other sources into HWiNFO Sensors like Arduino devices) or when certain sensor values are not shown in HWiNFO (for example a custom set of combined values).

The interface used to specify such sensors is the Windows Registry. Users wishing to publish data into sensors will need to create a few specific keys and values in the registry. The keys are predefined as:

HKEY_CURRENT_USER\Software\HWiNFO64\Sensors\Custom

and from HWiNFO version 7.65 additionally

HKEY_LOCAL_MACHINE\Software\HWiNFO64\Sensors\Custom

All sensors must be contained inside the tree for HKEY_CURRENT_USER or HKEY_LOCAL_MACHINE. Under this node, up to 64 custom keys can be placed, each of them representing a unique sensor instance. The name of the key is used as the sensor name, for an example:

HKEY_CURRENT_USER\Software\HWiNFO64\Sensors\Custom\My Device1

The exact format is described in the HWiNFO forum:

www.hwinfo.com/forum/threads/custom-user-sensors-in-hwinfo.5817

Starting with HWiNFO version 6.21-4055 Beta, there is an additional option to use simple formulas. This new feature allows to create custom sensor values using simple math operations including combinations of existing sensor values. The entire formula has to be specified in the *Value* registry field, which needs to be of string (REG_SZ) type.

The formula can contain simple math operations with an integer or floating-point number or the value of an existing sensor. Currently the following operations are supported:

- + Addition
- - Subtraction
- * Multiplication
- / Division
- min(a, b, ...) minimum value
- max(a, b, ...) maximum value

Basically, the custom user sensors are a very effective feature, which improves the functionality of the HWiNFO sensor engine and is worth a closer look for every user, who wants to deal more with his sensor values and their interpretation.

10.4. Effective clock vs. instant (discrete) clock

It has become a common practice for several years to report **instant (discrete) clock** values for processors. This method is based on knowledge of the actual bus clock (BCLK) and sampling of core ratios at specific time points. The resulting clock is then a simple result of *ratio * BCLK*. Such approach worked quite well in the past but is no longer sufficient.

That is because over the years processors have become very dynamic components that can change their operating parameters hundreds of times per second depending on several factors including workload amount, temperature limits, thermal/VR current and power limits, turbo ratios, dynamic TDPs, etc.

While this method still represents actual clock values and ratios reported match defined P-States, it has become insufficient to provide a good overview of processor dynamics especially when parameters are fluctuating with a much higher frequency than any software is able to capture. Another disadvantage is that cores in modern processors that have no workload are being suspended (lower C-States). In such case when software attempts to poll their status, it will wake them up briefly and thus the clock obtained doesn't respect the sleeping state.

Hence a new approach needs to be used called the **Effective clock**. This method relies on hardware's capability to sample the actual clock state (all its levels) across a certain interval, including sleeping (halted) states. The software then queries the counter over a specific polling period, which provides the **average** value of all clock states that occurred in the given interval. HWiNFO version 6.13-3955 Beta introduced reporting of this clock.

Many users might be surprised how different this clock is in comparison to the traditional clock values reported. Please note that this effective value is the average clock across the polling interval used in HWiNFO.

This new method has been tested on several processors and has shown to provide more accurate results especially in scenarios with extremely fluctuating values.

More information can be found on the HWiNFO forum here:

www.hwinfo.com/forum/threads/effective-clock-vs-instant-discrete-clock.5958

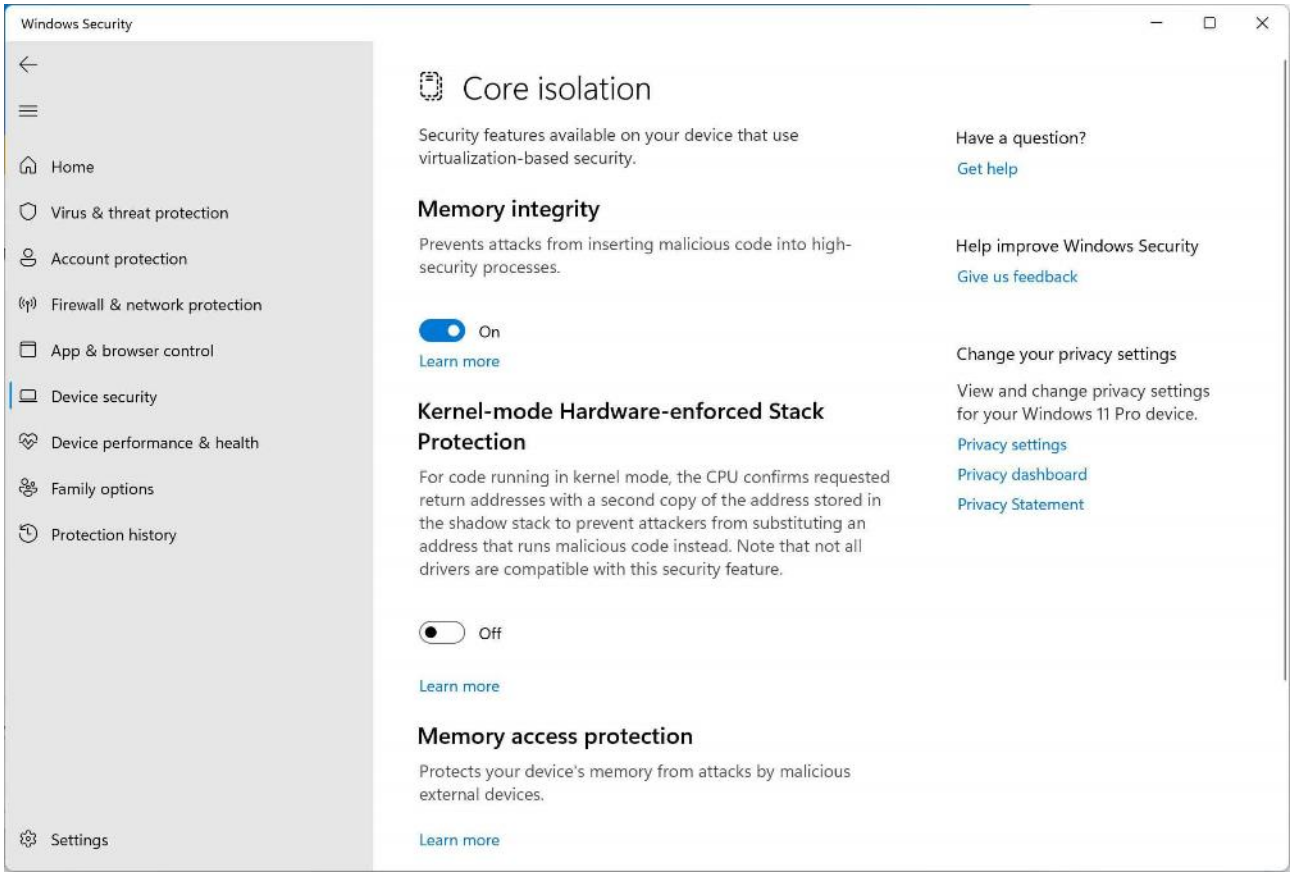
10.5. Core Frequency reading in Windows 11

On some systems running Windows 11 the core frequencies detected by HWiNFO might deviate strongly from the expected values. In such case, the bus clock (BCLK) reported by HWiNFO will be wrong, resulting in all active clocks (based on BCLK) and memory clock to be wrong too.

This is related to Virtualization Based Security (VBS) feature also called Hypervisor-protected Code Integrity (HVCI) which is activated when Core isolation/Memory integrity is enabled in Windows.

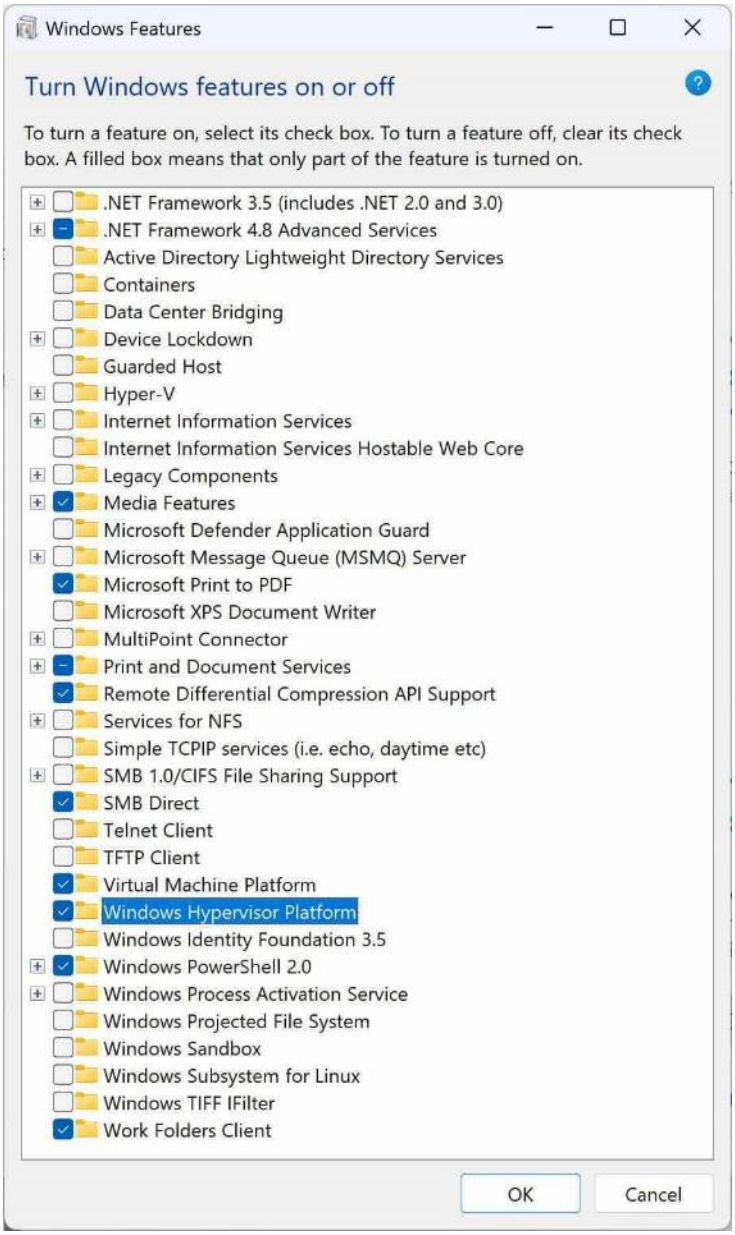
The problem with reporting correct clocks is because HVCI blocks certain interfaces that are needed to determine the accurate value of BCLK and it usually affects Intel 10th generation or earlier CPUs. Some AMD systems might be affected too.

Since it is not possible to bypass the virtualization, the general advice is to disable the "Memory integrity" feature if the system is affected by this problem. This can be performed via *Windows Security* settings. Within the Windows Security app navigate to *Device Security -> Core isolation -> Memory integrity*. Deactivating this option requires Administrator rights (Windows User Access Control prompt is displayed) and a Windows restart is necessary to make the changes effective.



Disabling *Memory integrity* might sometimes not be sufficient and still leave the Hyper-V active, so it might be needed to disable the entire Virtual Machine and Windows Hypervisor as well. This can be done via the *Turn Windows features on or off* control panel. The respective options are:

- Virtual Machine Platform
- Windows Hypervisor Platform.


Windows Features
Turn Windows features on or off
To turn a feature on, select its check box. To turn a feature off, clear its check box. A filled box means that only part of the feature is turned on.
.NET Framework 3.5 (includes .NET 2.0 and 3.0)
.NET Framework 4.8 Advanced Services
Active Directory Lightweight Directory Services
Containers
Data Center Bridging
Device Lockdown
Guarded Host
Hyper-V
Internet Information Services
Internet Information Services Hostable Web Core
Legacy Components
Media Features
Microsoft Defender Application Guard
Microsoft Message Queue (MSMQ) Server
Microsoft Print to PDF
Microsoft XPS Document Writer
MultiPoint Connector
Print and Document Services
Remote Differential Compression API Support
Services for NFS
Simple TCPIP services (i.e. echo, daytime etc)
SMB 1.0/CIFS File Sharing Support
SMB Direct
Telnet Client
TFTP Client
Virtual Machine Platform
Windows Hypervisor Platform
Windows Identity Foundation 3.5
Windows PowerShell 2.0
Windows Process Activation Service
Windows Projected File System
Windows Sandbox
Windows Subsystem for Linux
Windows TIFF IFilter
Work Folders Client
OK
Cancel

11. Shared Memory Support (Pro Version)

The Shared Memory function exposes the entire set of sensor data shown in HWiNFO to external applications over a defined interface. This feature has been available since version 3.83 of HWiNFO32/64 and. From HWiNFO32/64 version 4.30 onwards, a substantially revamped version 2 was added.

In practice, other applications can retrieve and process the sensor data in real time, with HWiNFO serving as a data provider with its advanced sensor engine. For this purpose, HWiNFO can be configured to load the sensors at program launch and to minimize the main window at the same time - thus opening the sensor status directly after program launch.

Within the Main Settings the option *Shared Memory Support* needs to be enabled. This feature can be used without any restrictions in the Pro version of HWiNFO, but in the free (non-Pro) version this option is automatically disabled after 12 hours of run time and has to be manually enabled again.

The layout of the Shared Memory Interface and additional examples in various programming languages are described in the following forum section:

www.hwinfo.com/forum/forums/shared-memory-interface.13

To access this section, an account on the forum is required.

A helpful tool to view the exported sensor data is called *HWiNFO Shared Memory Viewer*. When the *Shared Memory Support* option is enabled in HWiNFO and the sensor status is active, this tool will display all exported data including additional details useful for understanding the layout.

Each attribute is exported to predefined fields, including the sensor type, its unique ID, original/displayed name, original/displayed unit, and the values (current, minimum, maximum and average). A new version of this interface added UTF-8 encoded strings.

12. Add-ons

Over the years, various 3rd party applications have been created for HWiNFO that extend the functionality and experience in many ways. Moreover, some other applications have also added interfaces to HWiNFO (usually via the Shared Memory interface). A list of such add-ons or applications can be found in various threads in the HWiNFO forum at:

www.hwinfo.com/forum/forums/3rd-party-extensions-plug-ins-gadgets.8

or listed on the official HWiNFO homepage:

www.hwinfo.com/add-ons

Below is a list of the most important add-ons with a short description:

RivaTuner/MSI Afterburner/EVGA Precision On-Screen Display:

- On-Screen Display of RivaTuner Statistics Server (RTSS OSD), MSI Afterburner or EVGA Precision
- Displays any of the HWiNFO32/64 sensor values in DirectX applications/games via OSD
- RTSS OSD renders values straight into the full-screen graphics application
- Flexible and configurable
- Directly supported by HWiNFO, or
- Dedicated plugin in MSI Afterburner which provides even more options for layout customization

Rainmeter plug-in:

- A plug-in for the Rainmeter gadget
- Displays any HWiNFO32/64 sensor values in the Rainmeter gadget

PromDapter:

- A customizable Prometheus adapter for HWiNFO with Grafana Dashboard

Elgato Stream Deck plug-in:

- Display any HWiNFO sensor values on Elgato Stream Deck with this plug-in
- Flexible, configurable, open-source

GenericLogViewer:
- Analyze and compare HWiNFO sensor log files
- A tool to display HWiNFO sensor log data in different graphs and compare multiple log files side by side in one graph

Aquasuite:
- Visualize or use as control source any HWiNFO sensor values with Aquasuite
- Aqua Computer Aquasuite supports reading HWiNFO sensor values

Samurize plug-in:
- A plug-in for Samurize
- Get all HWiNFO32/64 sensor values in Samurize (system monitoring and desktop enhancement engine)

HWiNFOMonitor:
- An advanced, fully customizable sidebar gadget with charts, bars, etc.
- SideShow support
- Based on the popular GPU Monitor

Online Report Converter/Viewer:
- Upload HWiNFO report files to web
- Use to upload HWiNFO XML reports and view them back with a unique URL as a HTML page

InfoPanel - Desktop Visualisation Software:
- Display HWiNFO sensors on a USB only LCD (like BeadaPanel)

Home Assistant (MQTT):
- Attach HWiNFO sensor data in Home Assistant via the MQTT protocol

PromDapter - Prometheus Adapter (+ Grafana Dashboard):
- A customizable open-source dashboard for HWiNFO sensors via Prometheus Adapter and Grafana.

13. HWiNFO SDK

The HWiNFO Software Development Kit (SDK) allows integration of the proven HWiNFO engine for hardware detection/inventory and sensor monitoring into other software. This product is available for larger projects with individual pricing and customization per project.

More details are located at:

www.hwinfo.com/sdk

14. Support

The HWiNFO team always strives for highest accuracy and quality of the software which also requires extensive testing of a vast amount of hardware components, respective drivers and their combinations. Even through such enormous effort, sometimes problems can happen which might require users to seek active support. This can be for example: missing information, misreporting of parameters, insufficient support of some (early) platforms or problems during runtime like freezing or crashing.

The HWiNFO team offers various support channels with a fast turn-around time. Depending on the problem, following are the recommended options:

- Billing problems: Direct e-mail
- Pro License retrieval, subscription management:
  2Checkout MyAccount - https://secure.2co.com/myaccount
- Technical problems or questions:
  HWiNFO Forum - www.hwinfo.com/forum

Reporting a technical problem providing detailed information, especially:

- Detailed problem description.
  A screenshot of the affected area might be helpful in many cases
- Description of observed vs expected behavior
- A HWiNFO Report File
- The HWiNFO Debug File (see below)

14.1. HWiNFO Debug File

The Debug Mode in HWiNFO can be understood as developer mode, where HWiNFO stores additional internal program and system details in a separate file. This file is called HWiNFO32.DBG or HWiNFO64.DBG respectively for the corresponding version of HWiNFO and is located in the directory of the launched HWiNFO executable. The file size varies and depends on system configuration or run time length as it constantly adds data during the whole run time.

The Debug Mode should be enabled before the initial system scan to contain these data too. With a later activation - for example after the system analysis - important system data might not be present.

Therefore, as first the main settings in the welcome screen should be opened, and in the Safety tab the *Debug Mode* option activated. The second option called *Debug Write Direct* ensures that the debug data is written directly to the debug file without caching or buffering. In case of entire system crash, this option ensures that all debug data will be written to the debug file up to that point and can be retrieved after new system start.

HWiNFO denotes active Debug Mode by an additional identifier "[D]" in the window title of the welcome dialog or system summary. The system analysis performed at program launch takes a little longer with active Debug Mode, because additional data are gathered and written into the debug file.

After program launch, it is recommended to open the sensor window, because only then the whole sensor data is logged in the debug file.

During program run time the debug file cannot be accessed by other applications because it is locked by HWiNFO and is unlocked only after closing HWiNFO. Sometimes (especially during longer runtime) the file can grow quite large, so it's recommended to compress it using any common compression method (for example ZIP, RAR, 7-Zip). This archive should then be attached to the complete report.

14.2. Forum & Community

In the many years of HWiNFO's market presence, an extensive and unique and complex community has formed that exchanges and compares program results in various forums. In the overclocking and gaming scene, sensor details are often in the foreground to provide a proof of system state, achieved frequencies, cooling capabilities, etc.

The HWiNFO team also hosts its own forum which can be reached at:

www.hwinfo.com/forum

This forum serves as a primary portal for interaction with users and contains an extensive knowledge base of countless program functions, various discussions, feature clarifications and advisories. It's the main source of further information, update announcements and can also be used to provide feedback and problem reports to the team. When seeking for an advice or more information, it's recommended to first make a search on this forum, as it's quite likely that a similar question was already raised (and answered).

The HWiNFO team would like to thank all users for the active community and hope for many more years of technical exchange and appreciation - thank you.

15. Glossary

This glossary is intended to provide an overview of the most important abbreviations with their description. Due to the large number of possible abbreviations of technologies and products that go beyond the intended use of HWiNFO, a complete listing is almost impossible.

Nevertheless, a compact overview of the most important terms is collected below:

Abbreviation	Meaning
AES	Advanced Encryption Standard
AGC	Adaptive Gain Control
ASP	AMD Secure Processor
ATAPI	AT Attachment Packet Interface
BCLK	Bus Clock
D3D	Direct3D
DDC	Digital Display Channel
DDI	Digital Display Interface for DP or HDMI/DVI
DIMM	Dual In-line Memory Module
DMA	Direct Memory Access
DMI	Direct Media Interface
DPC	DIMMs Per Channel
DPPM	Dynamic Power Performance Management
DTS	Digital Thermal Sensor
EC	Embedded Controller
ECC	Error Correction Code
GT Cores	Graphic (iGPU) Cores
HFM	High-Frequency Mode
HPET	High Precision Event Timer
IA Cores	x86 Execution Cores
IMC	Integrated Memory Controller
IOMMU	I/O Memory Management Unit
IPU	Image Processing Unit
LFM	Low-Frequency Mode
LLC	Last-Level Cache
LPC	Low-Pin Count Interface
LTR	Latency-Tolerant Reporting
MCLK	Memory Clock
MCP	Multi-Chip Package
MCU	Microcode Update
MFM	Minimum Frequency Mode
MTB	Maximum Turbo Power

NVMe	Non-Volatile Memory Express
NVML	NVIDIA Management Library
OC	Overclocking
PBP	Processor Base Power
PCH	Intel Platform Controller Hub
PECI	Platform Environment Control Interface
PL	Power Limit
PMIC	Power Management Integrated Circuit
PTT	Intel Platform Trust Technology
SAS	Serial Attached SCSI
SATA	Serial AT Attachment
SDK	Software Development Kit
SDP	Scenario Design Power
SIO	Super I/O chip
SMART	Self-Monitoring, Analysis and Reporting Technology
SMBus	System Management Bus
SPD	Serial Presence Detect
SPI	Serial Peripheral Interconnect
SSC	Spread Spectrum Clock
SST	Intel Speed Shift Technology / Speed Select Technology
TBMT	Intel Turbo Boost Max Technology 3.0
TBT	Intel Turbo Boost Technology
TCC	Thermal Control Circuit
TDC	Thermal Design Current
TDP	Thermal Design Power
TLB	Translation Lookaside Buffer
TPM	Trusted Platform Module
TVB	Intel Thermal Velocity Boost
TXT	Intel Trusted Execution Technology
Uncore	Denotes all parts except the execution core
UCLK	Uncore Clock
VID	Voltage Identification
VT	Intel Virtualization Technology