



Intel® SoC Watch for Linux*

Release Notes

October 2020

Version 2020.4

Intel Corporation

www.intel.com

[Legal Information](#)

Contents

Legal Information	3
Version History	4
Customer Support.....	5
Chapter 1: Introduction	
Chapter 2: New in This Release	
Chapter 3: System Requirements	
Supported Architectures	8
Dependencies	8
Chapter 4: Where to Find the Release	
Chapter 5: Installation Notes	
Extracting the Intel SoC Watch Package.....	10
Intel SoC Watch Prerequisites for Yocto and Wind River* Linux	10
Build the Kernel Modules	11
Building Linux* Kernel Modules	11
Install Intel SoC Watch.....	11
Intel SoC Watch for Linux* Installation.....	12
Prerequisites for FPGA metric collection using Intel SoC Watch	12
Key Files	13
Remove the Intel SoC Watch Drivers	13
Chapter 6: Fixed Issues	
Chapter 8: Related Documentation	

Legal Information

Intel technologies may require enabled hardware, software or service activation.

No product or component can be absolutely secure.

Your costs and results may vary.

© Intel Corporation. Intel, the Intel logo, and other Intel marks are trademarks of Intel Corporation or its subsidiaries. Other names and brands may be claimed as the property of others.

Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice.

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

The products described may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.

Intel, the Intel logo, Intel Atom, Intel Core, Intel Xeon Phi, VTune and Xeon are trademarks of Intel Corporation in the U.S. and/or other countries.

Microsoft, Windows, and the Windows logo are trademarks, or registered trademarks of Microsoft Corporation in the United States and/or other countries.

Java is a registered trademark of Oracle and/or its affiliates.

OpenCL and the OpenCL logo are trademarks of Apple Inc. used by permission by Khronos.

© **Intel Corporation**

Version History

These are the main releases of Intel® SoC Watch:

Date	Revision	Description
June, 2019	2.11	Improves handling of unrecognized CPUs, reporting S-state when hibernation occurs, and other bug fixes.
September, 2019	2019.12	Added support for Intel platform code named Ice Lake. Modified hw-cpu-pstate reporting.
October, 2019	2019.13	Fixed issue in hw-cpu-pstate for Intel platform code named Ice Lake.
November, 2019	2020.1	Added support for Intel platform code named Comet Lake.
February, 2020	2020.2	Added collection of tool usage analytics. Added new features pch-slps0, pch-slps0-dbg. Improved error messages and help output. Enhanced driver security.
June, 2020	2020.3	Bug fix release.
July, 2020	2020.3.1	Bug fixes
September, 2020	2020.3.2	Bug fixes .
October, 2020	2020.4	Added support for Intel platform code named Tiger Lake. Added support dgfx-pwr support for discrete graphics card code named DG1. Added non-root user support. Added topology label in reports for some metrics. Re-named feature cpu-gpu-concurrency to cpu-igpu-concurrency. Removed support for older platforms.

Customer Support

For technical support, including answers to questions not addressed in this product, Intel® oneAPI IoT Toolkit forum (<https://community.intel.com/t5/Intel-oneAPI-IoT-Toolkit/bd-p/oneapi-iot-toolkit>).

Introduction

Intel® SoC Watch is a data collector for power-related data that can help identify issues on a platform that prevent entry to power-saving states. Captured metrics include:

- System sleep states
- CPU and GPU sleep states
- Processor frequencies
- Temperature data
- Device sleep states

You can correlate the collected data and visualize over time using Intel®VTune Profiler.

This document provides system requirements, installation instructions, issues and limitations, and legal information.

To learn more about this product, see:

- New features listed in the [New in This Release](#) section below, or in the help.
- Reference documentation listed in the [Related Documentation](#) section below
- Installation instructions can be found in the [Installation Notes](#) section below.
- For a detailed quick start guide to running the tool, see the *Intel SoC Watch User's Guide* in your installed documentation.

Optimization Notice

Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice.

Notice revision #20110804

2

New in This Release

The 2020.4 release (driver v2.13.0) contains these changes:

- Intel® SoC Watch reports now include additional topology information in column headers and row entries for some features. A change was required to differentiate data reported for multiple discrete graphics cards of the same type and discrete graphics card with multiple tiles. This change extended into the reporting of CPU and integrated graphics metrics due to the common underlying tool framework. The topology components are separated by "/" and the number of components in the topology string will vary depending on the topology of the target platform. For example, you will see "CPU/Package_0/Core_0" shown where in prior releases it was simply "Core_0". For integrated graphics, you now see "iGPU/Graphics" rather than "Graphics". For discrete graphics, you see "dGPU/DG1_173:0:0/Tile_0" where the Bus:Device:Function for the card is appended to uniquely identify its data. Reports for metrics that are system level or from PCH are unchanged.
- Changed feature name `-f cpu-gpu-concurrency` to `-f cpu-igpu-concurrency` to make it clear this refers to integrated GPU vs. discrete GPU. The table name was changed accordingly to "CPU-iGPU Concurrency Summary".
- Changed table names for features `-f hw-igfx-cstate` and `-f hw-igfx-pstate` to include "Integrated". Previously, these output tables did not identify the results as Integrated Graphics vs. Discrete Graphics.
- Added support for Intel platform code named Tiger Lake.
- Added support for collecting package power data on Intel discrete graphics card code named DG1 (`-f dgfx-pwr`). Collection of this feature requires the presence of a driver that is available when running on platforms with DG1 in the Intel DevCloud.
- Intel SoC Watch collections can now be executed by users in a specific group that do not have root privilege, however inserting the SoC Watch driver kernel module still requires root privilege. To insert the SoC Watch kernel module and allow users in the default group 'vtune' (defined for use with Intel® VTune™ Profiler) to run collections, change to the SoC Watch install directory and run: `./drivers/insmod-socwatch`. Use command `rmmmod-socwatch` to remove permission for the default group. To give non-root collection access to users of a different specific group, use the following : `./drivers/insmod-socwatch -g <permitted-group>`. Note: The group should already exist on the target system for the command to succeed.
- Support for the following older platforms has been removed in this release: Intel platforms code named Anniedale, Valleyview, Cherry Trail, Haswell, and Broadwell.

System Requirements

Supported Architectures

Intel SoC Watch supports these Intel microarchitecture or platform code names:

- Denverton
- Apollo Lake
- Gemini Lake
- Skylake
- Kaby Lake
- Coffee Lake
- Whiskey Lake
- Amber Lake
- Comet Lake
- Ice Lake
- Tiger Lake
- Skylake-Xeon
- Cascade Lake-Xeon

Intel SoC Watch supports these Intel discrete graphics code names:

- DG1

Dependencies

Intel SoC Watch depends on specific OS configurations and hardware capabilities. If these are not present on the target system, Intel SoC Watch may fail to work properly.

- Linux Kernel version needs to be 2.6.32 or later.
- GNU C Library version must be GLIBC_2.17 or later.
- `KERNEL_CONFIG_TRACEPOINTS` must be enabled.
- Kernel should be compiled with "`CONFIG_MODULES`" enabled.
- P States
 - Kernel config `CONFIG_X86_SFI_CPUFREQ` or `CONFIG_X86_ACPI_CPUFREQ` must be enabled (i.e. set to 'y' or 'm').
 - One of these pstate drivers must be utilized: `sfi-cpufreq`, `acpi-cpufreq`, or `intel_pstate`. To determine which driver is loaded, check the `sysfs /sys/devices/system/cpu/cpu0/cpufreq/scaling_driver` file.
 - If one of these pstate drivers is not loaded, the kernel needs to be reconfigured and recompiled.
- C States
 - Kernel config `CONFIG_TIMER_STATS` must be enabled.
 - Kernel config `CONFIG_INTEL_IDLE` must be enabled and the `intel_idle` kernel module has to support the core of the target platform.
 - To determine if the `intel_idle` kernel module is loaded, check the `sysfs /sys/devices/system/cpu/cpuidle/current_driver` file. It must equal `intel_idle`. If it equals `acpi_idle`, only C0 and C1 will be used by the core.

Where to Find the Release



Intel SoC Watch is available in Intel® oneAPI Base Toolkit as a part of Intel® VTune™ Profiler.

Installation Notes

Intel SoC Watch for Linux* OS is installed as part of the Intel® VTune™ Profiler, which is included in the Intel® oneAPI Base Toolkit. Intel SoC Watch is used by Intel® VTune™ Profiler for collecting certain analysis, but it is also a standalone tool. Following are instructions for setting up Intel SoC Watch for standalone use.

Extracting the Intel SoC Watch Package

Intel SoC Watch must be installed on the target system (the system to be analyzed). To extract and install the package, the user should have sudo (root) privilege.

When Intel® oneAPI Base Toolkit is used to install Intel® VTune™ Profiler and it is installed directly on the target system:

Intel SoC Watch will already be extracted and is located in `/opt/intel/oneapi/vtune/<oneAPI-release-version>/socwatch`.

The Intel SoC Watch release notes and user's guides are located in `/opt/intel/oneapi/vtune/<oneAPI-release-version>/documentation/en/socwatch`.

When Intel® oneAPI Base Toolkit is used to install Intel® VTune™ Profiler and it is installed on a host system:

Intel SoC Watch will need to be copied to the target system and extracted. The install package is located on the host system at `/opt/intel/oneapi/vtune/<oneAPI-release-version>/target/linux/vtune_profiler_target_x86_64.tgz`. Copy `vtune_profiler_target_x86_64.tgz` to the target system and extract it into directory `/opt/intel` using command `tar zxvf vtune_profiler_target_x86_64.tgz -C /opt/intel/`.

The release notes and user's guide are not included in this package. They are located on the host system in `/opt/intel/oneapi/vtune/<oneAPI-release-version>/documentation/en/socwatch`.

You will need to build the kernel modules and install the SoC Watch driver to enable collection of all metrics. See instructions below.

Intel SoC Watch Prerequisites for Yocto and Wind River* Linux

The Intel SoC Watch binary is a C++ program that requires `libstdc++` to be present on the target system in order to function. Although most Linux distributions provide this by default, there are some minimal distributions of Yocto and Wind River Linux that may not.

To check if your OS image contains `libstdc++`, run this command:

```
find / -name "libstdc++"
```

If the library is present, you should see an output similar to:

```
/usr/lib64/libstdc++.so.6
```

If you see no output, the required library is missing. To fix this, do one of these steps:

- Rebuild Yocto or Wind River Linux with an option to include `libstdc++.so` file in the image.
- Provide the library file directly to Intel SoC Watch.

Rebuild OS image to include `libstdc++.so` file

1. Open the `projectDir/local.conf` file in a text editor.
2. Add the library with the `IMAGE_INSTALL_append` option and save the file:

```
IMAGE_INSTALL_append = " libstdc++"
```

NOTE Make sure that you include the leading space as it will be concatenated with any libraries previously added in a list.

3. Rebuild the platform project by running the following command from the projectDir:

```
make
```

Once the build completes, the library will be part of the project image rootfs. Proceed with reflashing the OS image to the target system and subsequently installing Intel SoC Watch.

Provide libstdc++.so library file directly to Intel SoC Watch

If you are unable to rebuild the OS image, but are able to obtain a valid libstdc++.so file (you can copy it from another system or build it yourself):

1. Unpack the SoC Watch package.
2. Copy the file into the /libs subfolder where it will be picked up at runtime.

libgcc_s.so Library

Some minimal Yocto distributions may also lack another library: libgcc_s.so. In this case, follow either of the aforementioned solutions.

Build the OS and include the image:

Use this command when building the OS.

```
IMAGE_INSTALL_append = " libgcc_s"
```

Copy file into Intel SoC Watch package:

Copy the library libgcc_s.so file into the /libs folder of the Intel SoC Watch package.

Build the Kernel Modules

If the Intel SoC Watch kernel modules (i.e. device drivers) are not present in the OS image of the target system, you will need to build and possibly sign them. Building and signing device drivers requires access to the kernel build directory for the OS image running on your target device. A kernel build directory is generated while building the OS image of the target system.

When building the kernel modules, do not open (or unzip) the Intel SoC Watch package (i.e. tar.gz file) on a Windows* based system then copy to a Linux system. The package must be extracted on the Linux build system using the unzip command to make sure the build scripts and make files are unmodified.

If a kernel is built with the CONFIG_MODULE_SIG kernel config enabled, any device driver loaded into that kernel must be signed with the same keys used to build the kernel. In general, drivers built for Linux targets do not need to be signed and the following description assumes the drivers do not need to be signed. But, if an end user tries to load an unsigned driver into a kernel that requires signed drivers, the `insmod` command will fail with the error "Required key" not available. If a signed driver is loaded into a kernel that does not require signed drivers, the load will succeed.

Building Linux* Kernel Modules

Linux kernel modules may only be built after the Intel SoC Watch package and kernel headers are copied to and installed on the target. See the section [Intel SoC Watch for Linux Installation](#) below for instructions on how to build the kernel modules for a target device running Linux.

Install Intel SoC Watch

Host: laptop, desktop, or server used to communicate with target device.

Target: device to be analyzed with Intel SoC Watch.

Intel SoC Watch for Linux* Installation

If Intel SoC Watch was previously installed on the target, delete the `socwatch_linux_*` directory before installing a new version. Then, perform the following steps on the target device.

1. Login to the device as root:

```
ssh root@<your_target_IP>
```

2. Identify the `<install-dir>`. To run SoC Watch on
 - a. *Host system:* `/opt/intel/oneapi/vtune/<oneAPI-release-version>/`
 - b. *Target system:* `/opt/intel/vtune_profiler_target`

3. Navigate to the Intel SoC Watch directory:

```
cd <install-dir>/sepdk/src
```

4. Use the following commands to build the SoC Watch driver:
 - a. To build the driver.

```
./build-driver -ni (switch "ni" refers to "No User Interface")
```

- b. Reload the module to load the SoC Watch driver.

```
./insmod-sep -r
```

- c. To verify if SoC Watch driver is loaded, use the below command to query.

```
./insmod-sep q
```

5. Use the below command to configure SoC Watch collection environment . To run on
 - a. *Host system:* `source <install-dir>/vtune-vars.sh`
 - b. *Target system:* `source <install-dir>/sep_vars.sh`
 - c. To set the environment variable, use the following command (The `<install-dir>` should be based on the host/target system.)

```
export PATH=<install-dir>/socwatch/x64:$PATH
```

NOTE Sourcing of `vtune-vars.sh` or `sep_vars.sh` is required if `/opt/intel/oneapi/setvarsh.sh` is not already sourced as a part of Intel® oneAPI Base Toolkit installation.

Prerequisites for FPGA metric collection using Intel SoC Watch

Intel SoC Watch leverages Open Programmable Acceleration Engine (OPAE) drivers to collect data on supported FPGA platforms. OPAE drivers must be loaded on the system containing one of the supported FPGA architectures (see [System Requirements](#)) prior to running Intel SoC Watch collections. Source code for OPAE drivers may be downloaded from the OPAE GIT repository located at the following link: <https://github.com/OPAE/opae-sdk/releases>

The OPAE installation guide provides more information on the installation requirements for OPAE drivers. The guide may be downloaded from the following location: https://opae.github.io/latest/docs/install_guide/installation_guide.html.

Follow these instructions to build the OPAE drivers for your system:

1. Download the kernel headers to your system.
2. Download and extract the file starting with `opae-intel-fpga-driver*.tar.gz` to your system.
3. Go to the extracted directory, locate the file named `Makefile`, and run the `make` command in that directory.

This process will generate several files with a `.ko` extension. These are the OPAE drivers that need to be loaded to enable Intel SoC Watch collections. Load these driver files using the following commands:

```
sudo insmod fpga-mgr-mod.ko
sudo insmod intel-fpga-pci.ko
sudo insmod intel-fpga-fme.ko
sudo insmod intel-fpga-afu.ko
```

Key Files

The following table describes the key files.

File	Description
<code>build_drivers.sh</code>	The build script used to build all of the device drivers utilized by Intel SoC Watch.
<code>socwatch2_x.ko</code>	The Intel SoC Watch kernel module used to collect both hardware and kernel data at runtime.
<code>setup_socwatch_env.sh</code>	The script used to setup the Intel SoC Watch runtime environment.
<code>socwatch</code>	The Intel SoC Watch executable built as a native application. Use this file to collect data and generate additional results from a raw SW2 file.
<code>SOCWatchConfig.txt</code>	The Intel SoC Watch configuration file. The configuration file is read by Intel SoC Watch immediately before each collection. It contains hardware addresses utilized by the device driver during the collection.
<code>EULA.txt</code>	End User License Agreement file.
<code>third-party-programs.txt</code>	List of third party programs included in the package.
<code>plugins/libSWCore.so</code>	A library providing Intel SoC Watch functionality.

Remove the Intel SoC Watch Drivers

Remove Intel SoC Watch drivers using the `rmmmod` command. For example: `rmmmod socwatch2_10`.

Fixed Issues

The 2020.4 release has a fix for these issues.

- Re-enabled features -f acpi-sstate, -f pch-slps0-dbg, -f pkg-pwr, and -f dram-pwr.

Related Documentation



The release contains these documents:

- Intel® SoC Watch for Android* OS and Linux* OS User's Guide
- Energy Analysis help (<https://software.intel.com/en-us/energy-analysis-user-guide>)