

SAR ESSENTIALS OPENCL INSTALLATION

October 2025

© 2025 NV5 Geospatial Solutions, Inc. All rights reserved.

This information is not subject to the controls of the International Traffic in Arms Regulations (ITAR) or the Export Administration Regulations (EAR).

CONTENTS

1	Introduction	3
2	OpenCL runtimes	3
2.1	CPU.....	3
2.2	GPU	5
2.3	Accelerator.....	6
3	Support and troubleshooting	6
3.1	OpenCL configuration scanner (sarxinfo)	6
3.2	OpenCL configuration tester (sarxtest).....	8
3.3	Known issues and limitations	9
4	Frequently Asked Questions (FAQ).....	11

1 INTRODUCTION

We progressively adopted general-purpose computing on Graphics Processing Unit (GPU) technologies to increase the processing performance through improved parallelism. We based our solution on the Open Computing Language (OpenCL) framework, which is an open standard widely supported by most hardware manufacturers and software developers.

To use SAR Essentials, a working **OpenCL 1.2 runtime must be available** on the target machine.

An OpenCL runtime is a kind of third-party software package that allows applications running on the target machine to execute OpenCL-related operations through the selected OpenCL platform/device. A platform is a specific OpenCL implementation (e.g., provided by NVIDIA, Intel, or AMD), while devices are the actual processing hardware (e.g., GPU or CPU) used for carrying out calculations.

According to the hardware characteristics of the machine that you plan to use for SAR Essentials, the following sections might help you in finding out how to properly install and configure OpenCL.

2 OPENCL RUNTIMES

There are three main categories of OpenCL runtimes designed to work with specific devices: CPU, GPU, and accelerator. Depending on the choices made by the implementer of the OpenCL runtime, one single platform might support multiple, heterogeneous devices, or multiple OpenCL platforms are required for devices of different type.

2.1 CPU

One key advantage of OpenCL over other computing frameworks is that it does not require a specific type of device nor brand of GPU to start using it. Even better: one **does not need a GPU at all to use OpenCL** since CPU-only runtimes are available. These CPU-only implementations are capable of exploiting the computational power provided by your CPU at the service of OpenCL.

Although not really required when a powerful OpenCL-compatible GPU is available, we recommend to always have a CPU-only OpenCL runtime installed. In this way, most OpenCL-related problems due to a corrupted installation or a driver issue can be easily identified by switching from an OpenCL runtime to another.

To the best of our knowledge, there are currently three CPU-only runtimes: one provided by Intel (which we recommend), and two open-source implementations with different levels of maturity (PoCL and another one as part of the MESA project). AMD also provided a CPU-only runtime in the past, but it has been discontinued.

2.1.1 Intel CPU-only runtime

Intel CPU-only runtime provides a highly optimised CPU-only implementation of OpenCL that takes advantage of the most recent multicore architectures and CPU instruction sets (such as SSE4.2 and AVX). Although these OpenCL CPU runtimes are provided by Intel, they **also work on AMD CPUs**.

The OpenCL runtime installation package for **Windows** is provided as a stand-alone executable, while the **Linux** packages are made available via Linux-distribution specific repositories.

Windows installation

For Windows, the OpenCL runtime is provided within the SAR Essentials installation package: the user can choose to install it during the installation procedure. The same or a more recent runtime version can be also downloaded from the following link:

<https://www.intel.com/content/www/us/en/developer/articles/technical/intel-cpu-runtime-for-opengl-applications-with-sycl-support.html>.

Linux installation

For Linux, it is therefore necessary to follow the instructions provided at this link:

<https://www.intel.com/content/www/us/en/developer/tools/oneapi/base-toolkit-download.html>.

The steps to perform the installation to date (this could be changed by Intel) are:

- Select the OS version:
 - choose Linux.
- Select the Linux distribution-specific package manager to use:
 - choose APT for Ubuntu or Debian;
 - choose Zypper for SUSE;
 - choose YUM/DNF for Red Hat, Fedora, or Rocky.
- Add the Intel-specific repositories to the Linux distribution:
 - Follow the instructions found on the “[selected package manager] Prerequisites | Set Up the Repository” section.

At this point, the user could run the command as suggested by the “Install with [selected package manager]” but it will download and install much more software components than the few ones that are really required by SAR Essentials. Instead, the user could opt to manually install only the necessary OpenCL packages:

- Install intel-oneapi-runtime-opengl and intel-oneapi-runtime-libs packages:
 - for APT (Ubuntu/Debian): “sudo apt install intel-oneapi-runtime-opengl intel-oneapi-runtime-libs”;
 - for Zypper (SUSE): “sudo zypper install intel-oneapi-runtime-opengl intel-oneapi-runtime-libs”;
 - for YUM/DNF (Red Hat/Fedora/Rocky): “sudo yum install intel-oneapi-runtime-opengl intel-oneapi-runtime-libs”.

- Verify the installation:

The OpenCL runtime can be verified by running the “./sarxinfo” application from within the SAR Essentials bin directory (e.g.: /usr/local/SAR Essentials/bin/). If the OpenCL installation was performed correctly you should obtain a message saying:
“[OK] This machine seems to be SAR Essentials compatible”.

Notes:

1. A default FPGA emulator may be installed along with the runtime. SAR Essentials will ignore this by default. If it causes issues or prevents sarxinfo from completing properly, remove it by deleting the file intel64-fpgaemu.icd by running the following command:

```
sudo rm /etc/OpenCL/vendors/intel64-fpgaemu.icd
```

2. The SAR Essentials installer includes the latest compatible version of the Intel CPU-only OpenCL runtime available at the time of release. If this version does not work on your system, consider using an earlier version:

- A previous runtime version is located under the config_file directory in SAR Essentials
- Additional versions are available on Intel website under the “Intel CPU Runtime for OpenCL Applications” section:
<https://www.intel.com/content/www/us/en/developer/articles/tool/opencl-drivers.html>

2.1.2 Portable Computing Language (PoCL)

PoCL is an open-source (MIT-licensed) implementation of the OpenCL standard. At the time of writing, it supports OpenCL 1.2 with some additional features. Its installation and usage are less immediate than with the Intel CPU-only package: for this reason, it is only recommended to experienced users who have specific reasons to use it.

It is available at this link:

<http://portablecl.org/>

2.2 GPU

OpenCL runtimes for a specific GPU are usually part of the GPU drivers provided by each hardware manufacturer. Please be aware that OpenCL support is often only included in the official full version of the graphic drivers that can be downloaded from each hardware manufacturer: default/generic drivers installed by the operating system might only provide a minimal set of the features supported by a specific model.

If your computer has one single GPU and if you plan to use this GPU for OpenCL acceleration, pay attention to the fact that the same hardware resources will be shared among SAR Essentials and the operating system. In such a case, you might consider using a larger, more powerful GPU with more video memory than the minimal 2 GB, or using two GPUs (one for graphics, one for computing), or switching to a CPU-only OpenCL runtime.

2.3 ACCELERATOR

Accelerators are another (rare) category of devices suitable for OpenCL. Intel Xeon Phi coprocessors and some FPGA-based processing boards belong to this group.

As long as such devices are OpenCL 1.2-compatible, follow the same standards of the CPU and GPU runtimes, and satisfy our basic requirements, they can be used for SAR Essentials.

3 SUPPORT AND TROUBLESHOOTING

Since OpenCL runtimes are third-party software components out of our control, we can offer only a limited range of support in this regard. However, we provide a series of tools and recommendations to help users in the identification of OpenCL-related issues and for guiding them towards a solution.

3.1 OPENCL CONFIGURATION SCANNER (SARXINFO)

The sarxinfo application is the core software instrument used by SAR Essentials to inspect the underlying machine configuration for hardware specs and available OpenCL runtimes. If sarxinfo fails in finding a working OpenCL 1.2-compatible runtime, SAR Essentials cannot be used until this requirement is fulfilled.

```
-----
OpenCL platforms:
-----
```

```
Nr. of platforms . : 2
Platform 0 . . . : Intel(R) OpenCL
  Available devices: 2 (1 GPUs, 1 CPUs, 0 accelerators)
    Device 0 . . . : Intel(R) HD Graphics 4600 (def)
      Device type . : GPU
      Device version : OpenCL 1.2
      Driver version : 20.19.15.4835
      FP64 support . : no [!] FP64 required
      SPIR support . : yes
      Global mem size: 1629 MB
      Local mem size : 64 KB
      Max mem object : 407 MB
      Address bits . : 64
      Compute units . : 20
      Global cache . : 256 KB
      Max clock freq : 1300 MHz
      Max workgroup . : 512
      Image support . : yes
      Context check . : passed
      Device ready . : no
    Device 1 . . . : Intel(R) Core(TM) i7-4810MQ CPU @ 2.80GHz
      Device type . : CPU
      Device version : OpenCL 1.2 (Build 10094)
      Driver version : 5.2.0.10094
      FP64 support . : yes
      SPIR support . : yes
      Global mem size: 16288 MB
      Local mem size : 32 KB
```

```

Max mem object : 4072 MB
Address bits . : 64
Compute units  : 8
Global cache . : 256 KB
Max clock freq : 2800 MHz
Max workgroup  : 8192
Image support  : yes
Context check  : passed
Device ready . : yes

Platform 1 . . . : AMD Accelerated Parallel Processing
Available devices: 2 (1 GPUs, 1 CPUs, 0 accelerators)
Device 0 . . . : Capeverde (def)
Device type . . : GPU
Device version : OpenCL 1.2 AMD-APP (2348.3)
Driver version : 2348.3
FP64 support . : yes
SPIR support . : yes
Global mem size: 2048 MB
Local mem size : 32 KB
Max mem object : 1344 MB
Address bits . : 64
Compute units  : 10
Global cache . : 16 KB
Max clock freq : 775 MHz
Max workgroup  : 256
Image support  : yes
Context check  : passed
Device ready . : yes
Device 1 . . . : Intel(R) Core(TM) i7-4810MQ CPU @ 2.80GHz
Device type . . : CPU
Device version : OpenCL 1.2 AMD-APP (2348.3)
Driver version : 2348.3 (sse2,avx)
FP64 support . : yes
SPIR support . : yes
Global mem size: 16288 MB
Local mem size : 32 KB
Max mem object : 4072 MB
Address bits . : 64
Compute units  : 8
Global cache . : 32 KB
Max clock freq : 2794 MHz
Max workgroup  : 1024
Image support  : yes
Context check  : passed
Device ready . : yes

```

[OK] This machine seems to be SAR Essentials-compatible

The sarxinfo output contains three main blocks: hardware topology, supported types, and OpenCL platforms.

The “hardware topology” block is a short summary on the detected hardware configurations in terms of CPU architecture and memory (1). The “supported types” block is a technical report to make sure

that specific memory allocations have the expected size. This information can be simply ignored by a standard user.

The “OpenCL platforms” block contains most of the information required to make sure that a compatible OpenCL runtime is available.

If no OpenCL platform is listed in this block, this means that either no OpenCL runtime is installed or that its installation is probably broken. In such a case, make sure that a CPU or GPU runtime has been installed without errors, and double-check that the current user is allowed to access such a runtime.

If one or more OpenCL runtimes are found, they are listed as platforms and devices. In the previous example, two OpenCL platforms are available: one provided by AMD and one by Intel. The first platform (with ID 0) is the AMD implementation of OpenCL, installed alongside with the graphic driver, that enables using an AMD GPU with OpenCL. Such runtime has one single device of type GPU, and its OpenCL-related specifications are listed after. If some specifications are not in line with the minimum requirements, the problem is logged in the report. For each device, sarxinfo also performs a simple OpenCL initialization and compute test (under the field “Context check”): if this test fails, the OpenCL runtime is probably broken or not compatible with the underlying hardware (e.g., a CPU-only runtime has been installed on a virtual machine with a very limited virtual processor, or the wrong GPU driver has been installed). If the specifications satisfy the minimum requirements and the context test is successfully passed, the “Device ready” field is flagged with “yes” and this means that this platform/device combination can be used for SAR Essentials.

The second platform (with ID 1) is the CPU-only implementation of OpenCL provided by Intel. The only relevant difference with regards to the previous platform is the type of OpenCL device, which will appear as “CPU” instead of “GPU”.

If at least one working OpenCL platform/device combination is found, sarxinfo will terminate with a confirmation that SAR Essentials can be used on this machine. If no CPU-only runtime is found, an additional warning message will recommend to also install a CPU-only implementation of OpenCL. This warning is purely informative and does not affect the usage of SAR Essentials.

The application can be executed directly from the command-line. To do so, simply open a terminal, navigate under the “bin” directory where SAR Essentials has been installed and run “sarxinfo”.

The sarxinfo output is a useful attachment when you contact our support service for OpenCL-related issues to provide a better understanding of the system configuration.

¹ Under Windows, the reported amount of RAM is related to the currently available free memory. As this field is only informative, the different semantics has no impact on the execution of SAR Essentials.

3.2 OPENCL CONFIGURATION TESTER (SARXTEST)

A positive sarxinfo scan with at least one OpenCL device reported as ready for SAR Essentials is usually enough to make sure that the machine meets SAR Essentials requirements. However, OpenCL runtimes might have problems that only emerge when a deeper testing of the OpenCL functionality is made or have performance-related issues that are not easy to identify.

For these reasons, we provide an additional tool called `sarxtest`, which performs a series of tests and benchmarks on a specific OpenCL platform and device to make sure that it is fully operational and compatible with SAR Essentials.

This application can be started directly from the command-line: type “`sarxtest -h`” for a list of options to explicitly select a specific OpenCL platform and device.

```
SarXEngine test suite v1.6.3r, A. Peternier (C) sarmap SA 2014-2021
using SarX Accelerated Engine v1.3r
using SarX Algorithms v1.1r

Syntax . . : sarxtest [-param] {value} ...
Example . . : sarxtest -p 1 -d 0
Params . . : -p use explicit OpenCL platform ID (as reported by sarxinfo)
              -d use explicit OpenCL device ID (as reported by sarxinfo)
              -f load config from a .xml file
              -h read this help and exit
              -r redirect test results to a log file
              -b run the CPU benchmark test
              -s run the disk benchmark test
              -w wait for keypress after termination
```

Platform and device ID numbers are the ones returned in the `sarxinfo` report previously mentioned

3.3 KNOWN ISSUES AND LIMITATIONS

3.3.1 Timeout Detection Recovery (TDR) triggered under Windows

Under Windows, the Timeout Detection Recovery (TDR) is triggered when GPU is used as OpenCL device.

Context

In most cases, computers have one single GPU that is connected to a display and used for graphics rendering by the operating system. When such GPUs are OpenCL-enabled and meet SAR Essentials's minimum requirements, they can be used also to speed up the processing.

When the same GPU is used both for graphics and as OpenCL device for intensive SAR Essentials processing, the GPU might become so heavily loaded that graphical operations stutter or freeze. After a specific threshold (which is of 2 seconds, by default) the operating system considers the driver to be unresponsive and resets it.

Solution

To get rid or at least limit the impact of this problem, please consider the following workarounds:

1. Increase the TDR timeout. This is done by editing a key into the Windows registry. The key name is “`TdrDelay`”, of type “`DWORD (32-bit)`”, which is located under the “`HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\GraphicsDrivers`” section of the registry. Set its value to something higher than 10 (seconds), which provides

more flexibility over the default value of 2 seconds. If the key does not exist, simply create it.

An even simpler way to increase this value is by using one of the registry scripts provided by SAR Essentials under the “config_file” directory where SAR Essentials has been installed. The two “update_tdr_delay_to_XXs.reg” scripts will directly update the TDR timeout to 10 or 30 seconds, depending on which script is executed. Administrator rights are required to do so.

To make sure that the updated TDR value is considered by the operating system, reboot the machine after each modification.

2. Use a CPU-only OpenCL runtime for the processing of large or problematic datasets that trigger the TDR problem. This is one of the reasons why we recommend to also have a CPU-only runtime installed.
3. If possible, use two different GPUs: one for rendering, one for processing. If you have Nvidia hardware, see whether the compute-only GPU can be switched to Tesla Computer Cluster (TCC) mode. More information at this link:

<https://docs.nvidia.com/nsight-visual-studio-edition/reference/index.html#tesla-compute-cluster>

3.3.2 OpenCL runtimes get broken

Context

When a previously operational OpenCL runtime suddenly stops working, most of the times this is due to a corrupted driver or operating system configuration. Since OpenCL drivers for GPUs are embedded in the graphic drivers, when such drivers are updated (sometimes even automatically by the operating system itself), they might introduce regressions or change/reset some custom parameters (e.g., resetting the TDR value mentioned above).

In addition, Intel used to include a CPU-only OpenCL runtime alongside their GPU drivers (e.g., the Intel HD graphics series), which was different from the stand-alone CPU-only runtimes they also provide, and the two implementations were conflicting. This choice created many confusions in the past, but recent GPU drivers no longer include such CPU-only runtime anymore. However, if you are using an older version of the Intel GPU drivers, you might still have this issue.

Solution

Unfortunately, the variety of things that can go wrong with graphic drivers is too large to address all the possible issues in this regard.

In general, when a graphic driver-related issue is detected, a clean reinstallation of the GPU driver might fix the problem.

Updating the driver to a more recent version can also help, but in some cases a more recent version could introduce regressions at the OpenCL level and a downgrade to a previous driver version be required.

If the Intel HD graphics driver is problematic, some laptops with both an integrated and discrete GPU allow disabling the integrated GPU from the BIOS. But this option would drain the computer battery much faster.

3.3.3 No OpenCL found after the installation of an OpenCL runtime

Context

Although an OpenCL runtime has been freshly installed, it is not detected by sarxinfo.

Solution

Even if not explicitly required, it is often necessary to reboot the machine before a freshly installed OpenCL platform/device is correctly recognized by the system.

Also double-check that the installed runtime is compatible with the CPU/GPU on the target machine and with the specific operating system version it is running on.

3.3.4 sarxtest FFT fails under Nvidia

sarxtest fails on Nvidia hardware during the testing of the Fast Fourier Transform (FFT).

Context

During the execution of sarxtest on an Nvidia GPU, the testing of OpenCL-accelerated Fast Fourier Transform (FFT) fails.

Solution

We noticed a driver regression in Nvidia drivers between version 370 to 392. Either rolling back the driver to a pre-370 version or updating to a post-392 release fixed the issue.

4 FREQUENTLY ASKED QUESTIONS (FAQ)

What is OpenCL?

The Open Computing Language (OpenCL) is a framework for writing applications capable of taking advantage of the parallel power provided by modern computational devices like multicore processors, graphics processing units (GPUs), field/programmable gate arrays (FPGAs), etc. OpenCL is an open standard adopted by most of today's hardware manufacturers. Unlike other similar frameworks such as Nvidia CUDA and Intel oneAPI, OpenCL is not tight to a specific vendor, operating system, or hardware platform.

We are using OpenCL in SAR Essentials to improve the performance of some specific algorithms.

Why using OpenCL in SAR Essentials?

Modern processors are providing additional computational power through improved parallelism rather than higher clock frequencies. For this reason, we already parallelized several portions of the SAR Essentials processing pipelines to cope with this shift in the programming paradigm.

GPUs and other modern massively parallel devices provide a significantly higher level of parallelism within a reasonable cost/benefit ratio. By writing our core units in OpenCL, we allow SAR Essentials to take advantage not only of multicore CPUs, but also (when available) of the parallel computational power provided by those dedicated devices.

GPUs are a cost-efficient solution to deal with the increasing big data issue of today's SAR processing.

Do I need a recent graphics card to run SAR Essentials?

No, although it really depends on the kind of processing you want to carry out with SAR Essentials. Unlike CUDA, where an Nvidia GPU is required to run any CUDA-related application, OpenCL provides some CPU-only implementations that can be used to run SAR Essentials on almost any computer. These CPU-only implementations are much more than a simple fallback for machines not equipped with a recent GPU since they are optimized for exploiting the hardware resources of the CPU (like multiple cores and advanced instruction sets).

If you plan to buy some hardware for explicit usage with SAR Essentials, feel free to contact us for some updated recommendation depending on your needs.

I don't have any GPU in my computer: can I run SAR Essentials?

Yes: through the SAR Essentials installer for Windows you can directly check the option to install the Intel CPU-only OpenCL runtime. The Intel CPU-only OpenCL runtime works on almost any recent Intel or AMD CPU.

If your CPU is relatively old or not supported by the Intel CPU-only runtime, you can always download and install a previous, older version of the Intel CPU-only runtime.

Will SAR Essentials run faster on my computer with two (or more) GPUs connected through Nvidia SLI or AMD CrossFire?

Nvidia SLI and AMD CrossFire are technologies exploited only when the GPUs are used for graphics rendering: they don't have any impact on OpenCL. Only one single OpenCL device can be used by SAR Essentials for a given processing: if several devices are available, it is up to the end-user to specify which one to use.

If performance is a critical factor, you might consider our SAR Essentials Cluster module, which splits the workload over a series of distributed nodes that can run on multiple computers, each one with one or more GPUs. Feel free to contact us for additional information.