

Intel® System Studio 2014 for Linux* Beta Installation Guide and Release Notes

Installation Guide and Release Notes for Linux* Host

Document number: 329487-001US

2 October 2013

Contents

1	Introduction	4
	Change History.....	4
	Intel® System Studio 2013 Update 1 for Linux*	4
	Intel® System Studio 2013 Update 2 for Linux*	4
	Intel® System Studio 2014 For Linux* Beta	5
	Intel® Software Manager	5
	Product Contents.....	6
2	Technical Support and Documentation	7
	Release Notes Location	7
	Article & Whitepaper Locations.....	9
	Support.....	9
3	What's New.....	10
	Cross-Development Support	10
	Operating System and Hardware Platform Support	10
	Tool Suite Component Features	11
	3.1.1 Optimizing Compiler.....	11
	3.1.2 Performance and Parallelism Tuning	11
	3.1.3 Power and Core Frequency Tuning	11
	3.1.4 Intel® Sampling Collector for Intel® VTune™ Amplifier 2014	11
	3.1.5 Remote Intel® VTune Amplifier Collector for Embedded Systems	12
	3.1.6 Memory Checking and Threading Analysis	12
	3.1.7 The GNU* Project Debugger GDB	13
	3.1.8 Intel® JTAG Debugger for System Software Debug	13

3.1.9	Software Visible Event Nexus (SVEN)	13
3.1.10	Intel® Math Kernel Library	14
3.1.11	Intel® Integrated Performance Primitives for Signal and Media Processing	14
4	System Requirements.....	15
4.1.1	Supported Host Platforms	15
4.1.2	Host Prerequisites and Resource Requirements.....	16
4.1.3	Target Software Requirements	17
4.1.4	Target Prerequisites and Resource Requirements.....	17
4.1.5	Hardware Requirements	18
5	Installation Notes	20
	Installing the Tool Suite	20
	Using the online installer	21
5.1.1	Online Installer Failure Reasons	22
	Installing the full distribution package	23
	Notes.....	27
5.1.2	Silent Install	27
5.1.3	Selecting the provided GDB.....	27
	Development target package installation	27
5.1.4	Intel® Inspector Command line interface installation.....	28
5.1.5	Intel® VTune™ Collectors Installation on Remote Systems	28
5.1.6	Intel® VTune™ Amplifier Sampling Enabling Product Installation on Remote Systems	29
5.1.7	Intel® Integrated Performance Primitives redistributable shared object installation	29
5.1.8	Intel® Math Kernel Library redistributable shared object installation	29
	Eclipse* IDE Integration.....	30
5.1.9	Installation	30
5.1.10	Launching Eclipse for Development with the Intel C++ Compiler	30
5.1.11	Editing Compiler Cross-Build Environment Files	30
5.1.12	Cheat Sheets	31
5.1.13	SVEN Trace Viewer integration into Eclipse*	31
5.1.14	Integrating the provided GDB into Eclipse* for remote debug	31
	Installing Intel® XDP3 JTAG Probe	32

Installing Macraigor Systems* usb2Demon* Support.....	32
Removing the Product.....	33
6 Virtualization	34
6.1.1 Using Intel® VTune™ Amplifier with Virtualization	34
6.1.2 Using GDB to debug applications running inside a virtual machine.....	34
7 Issues and Limitations.....	37
Known Issues and Limitations	37
7.1.1 Offline installation with non-root access	37
7.1.2 Running online-installer behind proxy server fails	37
7.1.3 The online-installer has to be run with sudo or root access.....	37
7.1.4 Yocto Project* Application Development Toolkit 1.2 (ADT) has only limited support for static linking	37
Intel® VTune™ Amplifier Power and Frequency Analysis	37
7.1.5 /boot/config-‘uname –r’ file must be present on platform	37
7.1.6 Power and Frequency Analysis only supported on Intel® Core™ Processor Family 37	37
Sampling Collector (SEP) for Intel® VTune™ Amplifier Usage with Yocto Project*	38
7.1.7 Building Sampling Collector driver on host Linux* system.....	38
Software Visible Event Nexus (SVEN).....	38
7.1.8 Eclipse* IDE Trace Viewer add-on is not signed Error! Bookmark not defined.	
7.1.9 SVEN SDK is not targeting Intel® Atom™ Processor CE series	38
8 Attributions.....	39
9 Disclaimer and Legal Information.....	40

1 Introduction

This document provides a brief overview of the Intel® System Studio 2014 For Linux* Beta and provides pointers to where you can find additional product information, technical support, articles and whitepapers.

It also explains how to install the Intel® System Studio product. Installation is a multi-step process. Please read this document in its entirety before beginning and follow the steps in sequence.

The Intel® System Studio consists of multiple components for developing, debugging, tuning and deploying system and application code targeted towards embedded designs.

The tool suite covers several different use cases targeting development for embedded intelligent system platforms ranging from Intel® Atom™ Processor based low-power embedded platforms to 3rd and 4th generation Intel® Core™ microarchitecture based designs. Please refer to the Intel® System Studio Product Guide for guidance on how to apply Intel® System Studio to the various use case scenarios that are available with this versatile product.

In addition the file embedded-usage.pdf will provide additional pointers to whitepapers and documents for Intel® Atom™ Processor targeted embedded software development. The file sp-usage.pdf provides additional pointers to whitepapers and documents for Intel® Core™ processor based development with focus on data throughput.

Due to the nature of this comprehensive integrated software development tools solution, different Intel® System Studio components may be covered by different licenses. Please see the licenses included in the distribution as well as the [Disclaimer and Legal Information](#) section of these release notes for details.

Change History

This section highlights important from the previous product version and changes in product updates. For information on what is new in each component, please read the individual component release notes.

Intel® System Studio 2013 Update 1 for Linux*

- Intel® C++ Compiler Support for cross-development targeting Wind River* Linux* 4

Intel® System Studio 2013 Update 2 for Linux*

- Intel® VTune™ Amplifier 2014 Update 1 for Systems with various improvements and fixes for remote event based sampling. Please see the Intel® VTune™ Amplifier 2014 for Systems release notes for details:

```
<install-dir>/system_studio_2014.0.xxx/vtune_amplifier_2014_for_systems/
documentation/en/release_notes_amplifier_linux.pdf

<install-dir>/
vtune_amplifier_2014_for_systems/documentation/en/remote-target-
collection.pdf
```

Intel® System Studio 2014 For Linux* Beta

1. Windows* host support for Linux* targeted development
2. Yocto Project* 1.2 through 1.5 support
3. Wind River* Linux* 5 native build support
4. Integration into Eclipse* build environment on both Linux* and Windows* OS
5. Support for 4th generation Intel® Core™ processor
6. Intel® C++ Compiler optimizations for next generation Intel® Atom™ processor
7. Improved sysroot cross-build integration support
8. Intel® JTAG Debugger support for 4th generation Intel® Core™ processor and Intel® Xeon™ processor as well as next generation Intel® Atom™ processor (requires NDA) a
9. System-wide Intel® VTune™ Amplifier performance analysis support

Intel® Software Manager

The installation now provides an Intel® Software Manager to provide a simplified delivery mechanism for product updates and provide current license status and news on all installed Intel software products.

You can also volunteer to provide Intel anonymous usage information about these products to help guide future product design. This option, the Intel® Software Improvement Program, is not enabled by default – you can opt-in during installation or at a later time, and may opt-out at any time. For more information please see <http://intel.ly/SoftwareImprovementProgram>.

Product Contents

The product contains the following components

1. Intel® C++ Compiler 14.0 for Embedded OS Linux*
2. Intel® Integrated Performance Primitives 8.0 Update 1 for Linux*
3. Intel® Math Kernel Library 11.1 for Linux* (Intel® 64 only)
4. Intel® VTune™ Amplifier 2014 for Systems
 - 4.1. Intel® VTune™ Amplifier Sampling Enabling Product (SEP) 3.10 Update 12
5. Intel® Inspector 2014 for Systems
6. The GNU* Project Debugger – GDB v7.5 (Provided under GNU General Public License v3)
7. Intel® JTAG Debugger 2014 for Linux*
 - 7.1. Intel® JTAG Debugger notification module xdbntf.ko (Provided under GNU General Public License v2)
8. SVEN Technology 1.0 (SDK provided under GNU General Public License v2)

2 Technical Support and Documentation

Release Notes Location

The release notes and getting started guide for the tools components making up the Intel® System Studio product can be found at the following locations after unpacking `I_cemdb_b_2014.0.xxx.tgz` and running the `install.sh` installation script.

The paths are given relative to the installation directory `<install-dir>`. The default installation directory is `/opt/intel` unless indicated differently.

Intel® System Studio Product Guide

- `<install-dir>/system_studio_2014.0.xxx/documentation/embedded_compsupdoc_1.pdf`

Intel® System Studio Getting Started Guide

- `<install-dir>/system_studio_2014.0.xxx/documentation/embedded_gsddoc_1.htm`

Intel® System Studio Release Notes

- `<install-dir>/system_studio_2014.0.xxx/documentation/all-release-install.pdf`

Embedded Use Case Whitepapers

- `<install-dir>/system_studio_2014.0.xxx/documentation/embedded-usage.pdf`

Signal Processing Use Case Whitepapers

- `<install-dir>/system_studio_2014.0.xxx/documentation/sp-usage.pdf`

Intel® C++ Compiler

- `<install-dir>/system_studio_2014.0.xxx/documentation/compiler-release.pdf`

Intel® Integrated Performance Primitives

- `<install-dir>/system_studio_2014.0.xxx/documentation/en_US/ipp/ReleaseNotes.htm`

Intel® Math Kernel Libraries

- `<install-dir>/system_studio_2014.0.xxx/documentation/en_US/mkl/ReleaseNotes.htm`

Intel® JTAG Debugger

- `<install-dir>/system_studio_2014.0.xxx/documentation/en_US/debugger/xdb/jtag-release-install.pdf`

SVEN SDK

- `<install-dir>/system_studio_2014.0.xxx/documentation/en_US/debugger/sven/sdk/SVEN_SDK.pdf`

SVEN Trace Viewer

- `<install-dir>/system_studio_2014.0.xxx/documentation/en_US/debugger/sven/viewer/SVEN_VIEWER.pdf`

GDB

- `<install-dir>/system_studio_2014.0.xxx/documentation/en_US/debugger/gdb/GDB_Release_notes.pdf`
- `<install-dir>/system_studio_2014.0.xxx/documentation/en_US/debugger/gdb/GDB.pdf`

Intel® VTune™ Amplifier

- `<install-dir>/system_studio_2014.0.xxx/vtune_amplifier_2014_for_systems/documentation/en/release_notes_amplifier_linux.pdf`
- `<install-dir>/vtune_amplifier_2014_for_systems/documentation/en/remote-target-collection.pdf`

Intel® Inspector

- `<install-dir>/system_studio_2014.0.xxx/inspector_2014_for_systems/documentation/en/Release_Notes_InspectorXE_Linux.pdf`

Intel® VTune™ Amplifier Data Collector

- `<install-dir>/system_studio_2014.0.xxx/vtune_amplifier_2014_for_systems/documentation/en/remote-target-collection.pdf`

Intel® VTune™ Amplifier Sampling Enabling Product

- The user's guide explaining the usage of the SEP command line tool for hardware event-based sampling collection on embedded devices can be found at `<install-dir>/system_studio_target/sep_3.10_axeu12_linux_<arch>/SEP_User_Guide.pdf` after unpacking the `system_studio_target.tgz` archive.

The installation directory `<install-dir>` for the Sampling Collector for Intel® VTune™ Amplifier 2014 is the directory into which the installation tar archive file `system_studio_target.tgz` has been extracted.

The architecture designation `<arch>` can take the values `ia32` or `x64`.

Article & Whitepaper Locations

For a list of all available articles, whitepapers and related resources please visit the Intel® System Studio product page at <http://software.intel.com/en-us/intel-system-studio> and look at the Support tab.

Support

If you did not register your compiler during installation, please do so at the [Intel® Software Development Products Registration Center](#). Registration entitles you to free technical support, product updates and upgrades for the duration of the support term.

To submit issues related to this product please visit the [Intel Premier Support](#) webpage and submit issues under the product **Intel(R) System Studio for Linux* Beta**.

Additionally you may submit questions and browse issues in the [Intel® System Studio User Forum](#).

For information about how to find Technical Support, product documentation and samples, please visit <http://software.intel.com/en-us/intel-system-studio>.

Note: If your distributor provides technical support for this product, please contact them for support rather than Intel.

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 optimizations 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 User and Reference Guides for more information regarding the specific instruction sets covered by this notice.

Notice revision #20110804

3 What's New

The Intel® System Studio has a set of new features specifically targeting cross-development needs and also adds support for new Intel® Architecture based hardware platforms.

10. Windows* host support for Linux* targeted development
11. Yocto Project* 1.2 through 1.5 support
12. Wind River* Linux* 5 native build support
13. Integration into Eclipse* build environment on both Linux* and Windows* OS
14. Support for 4th generation Intel® Core™ processor
15. Intel® C++ Compiler optimizations for next generation Intel® Atom™ processor
16. Improved sysroot cross-build integration support
17. Intel® JTAG Debugger support for 4th generation Intel® Core™ processor and Intel® Xeon™ processor as well as next generation Intel® Atom™ processor (requires NDA) a
18. System-wide Intel® VTune™ Amplifier performance analysis support

Cross-Development Support

- Improved support for –sysroot and compiler integration into cross-build environments
- Intel® System Studio is Yocto* Compatible approved.
- Support for OpenEmbedded* and Yocot Project* 3rd party toolchain layer recipes

Operating System and Hardware Platform Support

- Windows* host support for Linux* targeted development
- Yocto Project* 1.2 through 1.5 support
- Wind River* Linux* 4,5 native build support (standard LibC required)
- Support for 4th generation Intel® Core™ processor
- Intel® C++ Compiler optimizations for next generation Intel® Atom™ processor
- Intel® JTAG Debugger support for 4th generation Intel® Core™ processor and Intel® Xeon™ processor as well as next generation Intel® Atom™ processor (requires NDA)

Tool Suite Component Features

3.1.1 Optimizing Compiler

- Improved support for `--sysroot` and compiler integration into cross-build environments
- Intel® System Studio is Yocto* Compatible approved.
- Support for OpenEmbedded* and Yocto Project* 3rd party toolchain layer recipes

Please refer to

`<install-dir>/system_studio_2014.0.xxx/documentation/compiler-release.pdf`
for additional new features.

3.1.2 Performance and Parallelism Tuning

- Support for target systems based on Wind River* Linux* 5, Tizen* IVI 2.0, CE Linux* PR32, Yocto Project* 1.2 through 1.4.
- Remote data collection via standard command line interface (`amplxe-cl`)

Please refer to

`<install-dir>/system_studio_2014.0.xxx/
vtune_amplifier_2014_for_systems/documentation/en`
for additional new features.

3.1.3 Power and Core Frequency Tuning

- Power analysis on 4th generation Intel® Core™ processors based on the Intel microarchitecture code name Haswell.

Please refer to

`<install-dir>/system_studio_2014.0.xxx/
vtune_amplifier_2014_for_systems/documentation/en`
for additional new features.

3.1.4 Intel® Sampling Collector for Intel® VTune™ Amplifier 2014

- Support for 4th generation Intel® Core™ processor
- Support for latest generation Intel® Atom™ processor

Please refer to

`<install-dir>/system_studio_2014.0.xxx/
vtune_amplifier_2014_for_systems/documentation/en`

for additional new features.

3.1.5 Remote Intel® VTune Amplifier Collector for Embedded Systems

- Support for 4th generation Intel® Core™ processor
- Support for latest generation Intel® Atom™ processor

Please refer to

<install-dir>/system_studio_2014.0.xxx/

vtune_amplifier_2014_for_systems/documentation/en/remote-target-collection.pdf

for additional new features.

3.1.6 Memory Checking and Threading Analysis

- Support for 4th generation Intel® Core™ processor
- Support for latest generation Intel® Atom™ processor
- Windows* host support for Linux* targeted development
- Yocto Project* 1.2 through 1.5 support
- Wind River* Linux* 5 native build support

Please refer to

<install-dir>/ system_studio_2014.0.xxx/

inspector_2014_for_systems/documentation/en

for additional new features.

3.1.7 The GNU* Project Debugger GDB

- Windows* host support for Linux* targeted development
- Yocto Project* 1.2 through 1.5 support
- Wind River* Linux* 5 native build support

Please refer to

<install-dir>/system_studio_2014.0.xxx/documentation/en_US/debugger/gdb/GDB.pdf

and

<install-

dir>/system_studio_2014.0.xxx/documentation/en_US/debugger/gdb/GDB_Release_Notes.pdf
for details on these new features.

3.1.8 Intel® JTAG Debugger for System Software Debug

- Support for 4th generation Intel® Core™ processor
- Support for latest generation Intel® Atom™ processor

Please refer to

<install-dir>/system_studio_2014.0.xxx/documentation/en_US/debugger/xdp
for additional new features.

3.1.9 Software Visible Event Nexus (SVEN)

- Support for 4th generation Intel® Core™ processor
- Support for latest generation Intel® Atom™ processor
- Windows* host support for Linux* targeted development
- Yocto Project* 1.2 through 1.5 support
- Wind River* Linux* 5 native build support

Please refer to

<install-dir>/system_studio_2014.0.xxx/documentation/en_US/debugger/sven

For additional new features

3.1.10 Intel® Math Kernel Library

- Support for 4th generation Intel® Core™ processor
- Windows* host support for Linux* targeted development
- Yocto Project* 1.2 through 1.5 support
- Wind River* Linux* 5 native build support

Please refer to

<install-dir>/system_studio_2014.0.xxx/documentation/en_US/mkl
for additional new features.

3.1.11 Intel® Integrated Performance Primitives for Signal and Media Processing

- Support for 4th generation Intel® Core™ processor
- Support for latest generation Intel® Atom™ processor
- Windows* host support for Linux* targeted development
- Yocto Project* 1.2 through 1.5 support
- Wind River* Linux* 5 native build support
- Support for MMSE MIMO algorithm

Please refer to

<install-dir>/system_studio_2014.0.xxx/documentation/en_US/ipp
for additional new features.

4 System Requirements

4.1.1 Supported Host Platforms

One of the following Linux distributions (this is the list of distributions supported by all components; other distributions may or may not work and are not recommended - please refer to Technical Support if you have questions).

In most cases Intel® System Studio for Linux* will install and work on a standard Linux* OS distribution based on current Linux* kernel versions without problems, even if they are not listed below. You will however receive a warning during installation for Linux* distributions that are not listed

- Red Hat Enterprise* Linux* 5, 6
- Ubuntu* 10.04 LTS, 12.04 LTS, 13.04
- Fedora* 17, 18
- Wind River* Linux* 5
- openSUSE 12.1
- SUSE LINUX Enterprise Server* 11 SP2

Individual Intel® System Studio 2014 for Linux* components may support additional distributions. See the [individual component's installation guide and release notes](#) after you unpacked and ran the installer for the tool suite distribution

```
> tar -zxvf l_cemdb_b_2014.0.xxx.tgz
```

for details.

Sudo or Root Access Right Requirements

- Integration of the Intel® C++ Compiler into a Yocto Project* Application Development Toolkit installed to /opt/poky/ requires the launch of the tool suite installation script install.sh as root or sudo user.
- Installation of the hardware drivers for the Intel® ITP-XDP3 probe to be used with the Intel® JTAG Debugger requires the launch of the tool suite installation script install.sh as root or sudo user.

Environment Setup

To setup the environment for the Intel® C++ Compiler and integrate it correctly with the build environment on your Linux host, execute the following command:

```
> source <install-dir>/system-studio-vars.sh <arg>
```

where `<install-dir>` is the top-level Intel® System Studio installation directory, and `<arg>` is one of the following architecture arguments:

`ia32`: Compilers and libraries for IA32 architectures only

`intel64`: Compilers and libraries for Intel® 64 architectures only

4.1.2 Host Prerequisites and Resource Requirements

4.1.2.1 Host Space Requirements by Component

	Minimum RAM	Recommended RAM	Disk Space
Intel® System Studio for Linux*	2Gb	4Gb	7Gb
Intel® System Studio for Linux* including JTAG Debugger	2Gb	4Gb	7Gb
Intel® C++ Compiler	1Gb	2Gb	2.5Gb
Intel® Integrated Performance Primitives	1Gb	4Gb	1-2Gb
Intel® Math Kernel Library	1Gb	4Gb	2.3Gb
Intel® VTune™ Amplifier for Systems	2Gb	4Gb	650Mb
Intel® Inspector for Systems	2Gb	4Gb	350Mb
GDB	1Gb	2Gb	200Mb
Intel® JTAG Debugger	1Gb	2Gb	300Mb
SVEN	2Gb	4Gb	350Mb

4.1.2.2 Intel® IPP Details

Intel IPP for IA-32 Hardware Requirements:

- 1800MB of free hard disk space, plus an additional 400MB during installation for download and temporary files.

Intel IPP for Intel® 64 Hardware Requirements:

- 1900MB of free hard disk space, plus an additional 700MB during installation for download and temporary files.

4.1.3 Target Software Requirements

The target platform should be based on one of the following environments:

- Yocto Project* 1.3, 1.4, 1.5 based environment
- CE Linux* PR32 based environment
- Tizen* IVI 1.0, 2.0
- Wind River* Linux* 4, 5 based environment

Note:

The level of target OS support by a specific Intel® System Studio component may vary.

4.1.4 Target Prerequisites and Resource Requirements

4.1.4.1 Target Space Requirement by Component

	Minimum RAM	Dependencies	Disk Space
Intel® C++ Compiler	application dependent	Linux kernel 1.26.18 or newer glibs-2.5 or compatible libgcc-4.1.2 or compatible libstdc++-3.4.7 or compatible	13Mb (IA-32) 15Mb (Intel® 64)
Intel® VTune™ Amplifier CLI	4Gb	Specific kernel configuration reqs. Details below.	200Mb
Intel® VTune™ Amplifier SEP	(# logical cores+2) Mb	specific kernel configuration reqs. Details below.	8Mb
Intel® VTune™ Amplifier amplex-runss	64Mb	specific kernel configuration reqs. Details below.	25Mb
Intel® Inspector for Systems CLI	2Gb	4Gb	350Mb
gdbserver	negligable	none	1.5Mb
xdbntf.ko	<1Mb	kernel build environment	<1Mb
SVEN	<1Mb	kernel build environment	sampling dependent

4.1.4.2 Intel® VTune™ Amplifier target OS kernel configuration

For Intel® VTune™ Amplifier performance and power analysis there are minimum kernel configuration requirements. The settings below are required for different analysis features.

- For event-based sampling (EBS) sep3_x.ko and pax.ko require the following settings:
CONFIG_PROFILING=y

CONFIG_OPROFILE=m (or CONFIG_OPROFILE=y)
CONFIG_HAVE_OPROFILE=y

- For EBS with callstack information vtsspp.ko additionally needs the following settings:
CONFIG_MODULES=y
CONFIG_SMP=y
CONFIG_MODULE_UNLOAD=y
CONFIG_KPROBES=y
CONFIG_TRACEPOINTS=y (optional but recommended)
- For power analysis, required by apwr3_x.ko
CONFIG_MODULES=y
CONFIG_MODULE_UNLOAD=y
CONFIG_TRACEPOINTS=y
CONFIG_FRAME_POINTER=y
CONFIG_COMPAT=y
CONFIG_TIMER_STATS=y
CONFIG_X86_ACPI_CPUFREQ=m (or CONFIG_X86_ACPI_CPUFREQ=y)
CONFIG_INTEL_IDLE=y

4.1.4.3 Intel® VTune™ Amplifier Feature vs. Resource Matrix

	Event based sampling (EBS) analysis	EBS analysis with stacks	Algorithmic analysis (PIN- based)	Power analysis	Remote collection from host	Result view on target	Requirements:
SEP "VTune Amplifier hardware event-based sampling collector for performance analysis"	X						~8 MB disk space (Number of logical cores +2) Mb RAM
amplxe-runss "VTune Amplifier collector for power and performance analysis on Embedded Linux systems"	X	X		X	X		~25 MB disk space ~64 Mb RAM
CLI "VTune Amplifier command line interface for text-based power and performance analysis"	X	X	X	X	X	X	~200MB disk space >= 4Gb RAM

4.1.5 Hardware Requirements

- IA32 or Intel® 64 architecture based host computer

- Development platform based on the Intel® Atom™ processor Z5xx, N4xx, N5xx, D5xx, E6xx, N2xxx, D2xxx, E3xxx, or Intel® Atom™ processor CE4xxx, CE53xx and the Intel® Puma6™ Media Gateway
- Alternatively development platform based on 2nd, 3rd or 4th generation Intel® Core™ processor.
- Alternatively development platform based on 2nd, 3rd or 4th generation Intel® Xeon® processor.

5 Installation Notes

Installing the Tool Suite

The default installation directories are:

- `/opt/intel/system_studio_2014.0.xxx/`
- `/opt/intel/system_studio_2014.0.xxx/ipp`
- `/opt/intel/system_studio_2014.0.xxx/mkl`
- `/opt/intel/system_studio_2014.0.xxx/debugger/xdb`
- `/opt/intel/system_studio_2014.0.xxx/debugger/gdb`
- `/opt/intel/system_studio_2014.0.xxx/debugger/sven`
- `/opt/intel/system_studio_2014.0.xxx/`
 `vtune_amplifier_2014_for_systems/`
- `/opt/intel/system_studio_2014.0.xxx/inspector_2014_for_systems/`

for Intel® C++ Compiler, Intel® Integrated Performance Primitives, Intel® Math Kernel Library, Intel® JTAG Debugger, Intel® VTune™ Amplifier, and Intel® Inspector respectively.

The Intel® System Studio contains components under GNU* Public License (GPL) in addition to commercially licensed components. This includes the GNU* Project Debugger – GDB, the SVEN SDK and the kernel module used by the Intel® JTAG Debugger to export Linux* dynamically kernel module memory load information to host.

The Intel® VTune™ Amplifier and Intel® Inspector are available for power (Intel® Core™ Processor Family only) and performance tuning as well as memory and thread checking on the installation host. There is the capability to remotely collect performance and power data for embedded cross-development use cases as described in `remote-target-collection.pdf`.

For additional installation of command-line only versions of Intel® VTune™ Amplifier and Intel® Inspector on the development target, please follow the sub-chapter on the command line interface (CLI) installations below.

Furthermore a target package `system_studio_target.tgz` contains Intel® C++ Compiler runtime libraries, the Intel® VTune™ Amplifier Sampling Enabling Product (SEP), target components for the Intel® VTune™ Amplifier Data Collector and the kernel module used by the Intel® JTAG Debugger to export Linux* dynamically kernel module memory load information to host.

The Intel® JTAG Debugger at this point is intended for Intel® Atom™ processor targeted system software cross-debug.

Sudo or Root Access Right Requirements

- Integration of the Intel® C++ Compiler into the Yocto Project* Application Development Toolkit requires the launch of the tool suite installation script `install.sh` as root or sudo user.
- Installation of the hardware drivers for the Intel® ITP-XDP3 probe to be used with the Intel® JTAG Debugger requires the launch of the tool suite installation script `install.sh` as root or sudo user.

Using the online installer

For installation of the tool suite on the development host please follow the steps below:

1. Ensure that you are connected to the internet and that https protocol based component downloads are permitted by your firewall.
2. Copy `online-installer.sh` into a directory of your choice.
3. Execute the install script in the directory where the tar file was extracted.
`>./online-installer.sh`
4. After launching the online installer you will see the following output:

```
Checking pre-requisites...
Checking for 'wget'...
Checking for 'awk'...
Checking for 'tar'...
Checking for 'cksum'...
Extracting data...
Starting installer...
This is an online-installer for Intel(R) System Studio 2014
Temporary files will be downloaded to: /tmp/root/.
Started downloading of installer. Please, wait for several
minutes... 100%
Installer have been downloaded successfully.
Extracting installer...
Starting the PSET client...
```
5. The target package `system_studio_target.tgz` will be located in `/tmp/<username>` after the host installation is complete. The actual location can be verified by looking at the online installer output above.
6. Follow the same steps as outlined in “Installing the full distribution package” steps 5 through 18 below.

5.1.1 Online Installer Failure Reasons

5.1.1.1 Symptom 1: Online install bootstrapper cannot connect to IRC.

```
~$ ./online-installer.sh
Checking pre-requisites...
Checking for 'wget'...
Checking for 'awk'...
Checking for 'tar'...
Checking for 'cksum'...
Extracting data...
Starting installer...
This is an online-installer for Intel(R) System Studio 2014
Connection to the IRC site cannot be established.
~$
```

Root Cause:

- Port 443 not opened in firewall
- Port 443 not opened in ACL for specific server (can be verified using network trace).
- The Intel® System Studio online installer currently does not fully support proxy servers.

5.1.1.2 Symptom 2: Download stops in an infinite loop

Download stops in an infinite loop.

```
:~$ ./online-installer.sh
Checking pre-requisites...
Checking for 'wget'...
Checking for 'awk'...
Checking for 'tar'...
Checking for 'cksum'...
Extracting data...
Starting installer...
This is an online-installer for Intel(R) System Studio 2014
Temporary files will be downloaded to:
/tmp/<user>/l_cemdb_b_2014.0.xxx.
```

Root Cause:

- Proxy server not specified on calling shell

5.1.1.3 Symptom 3: The online-installer hangs during step 6 of the installation

The online-installer hangs during step 6 of the installation

```
Step no: 6 of 7 | Installation
```

```
-----  
Each component will be installed individually. If you cancel the installation,  
components that have been completely installed will remain on your system. This  
installation may take several minutes, depending on your system and the options  
you selected.  
-----
```

Root Cause:

- The Intel® System Studio online installer hangs because the user's proxy environment is not inherited after changing user to root account during installation. Please run online installer under root account, make sure that proxy settings were set before.

Installing the full distribution package

For installation of the tool suite on the development host please follow the steps below:

1. Unpack the tool suite package in a directory to which you have write access.

```
> tar -zxvf l_cemdb_b_2014.0.xxx.tgz
```
2. Upon registering for the program you will receive a serial number and email with a license file. You will need either of these two to complete the installation process. If you want to use the license file you can point to it during install, but you can also copy it to `/opt/intel/licenses/` for automatic pickup by the installer.
3. Change into the directory the tar file was extracted to `../l_cemdb_b_2014.0.xxx`
4. Execute one of the installation scripts in the directory where the tar file was extracted.

```
> ./install.sh
```

Or

```
> ./install_GUI.sh
```

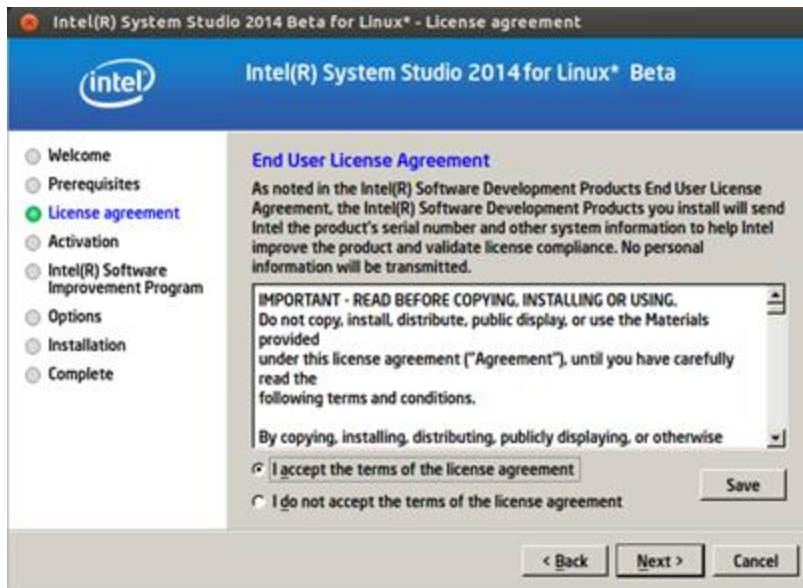
The later one will provide you with a full-GUI assisted installation experience. We will base the rest of this installation outline on the GUI install.

5. If you are not logged in as root, you will be asked if you want to install as root, install as root using sudo, or install without root privileges. Installing as root (using sudo if you have that privilege) is recommended, as that will update the system RPM database. Use the `install as current user` option if you want to install to a private area. To be able to install the Intel® JTAG Debugger however it is necessary to select "install as root" or "install as root using sudo". Without root privileges the option to install the Intel® JTAG Debugger will not be offered during install. Depending on the installation location of the Yocto Project* Application Development Toolkit the integration of the Intel® C++ Compiler with it may require root privileges as well.
6. The welcome message to the Intel® System Studio for Linux* installation process appears.

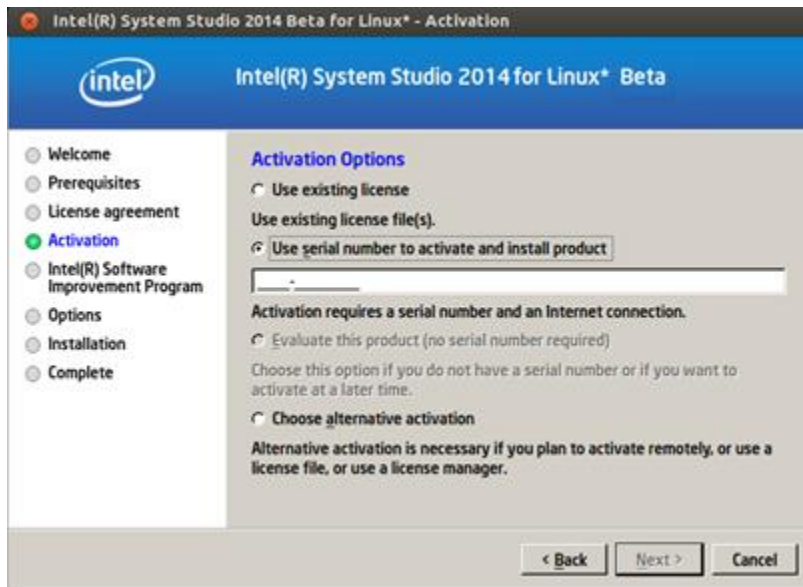


Selecting `Next` will move you to the next step in the installation process.

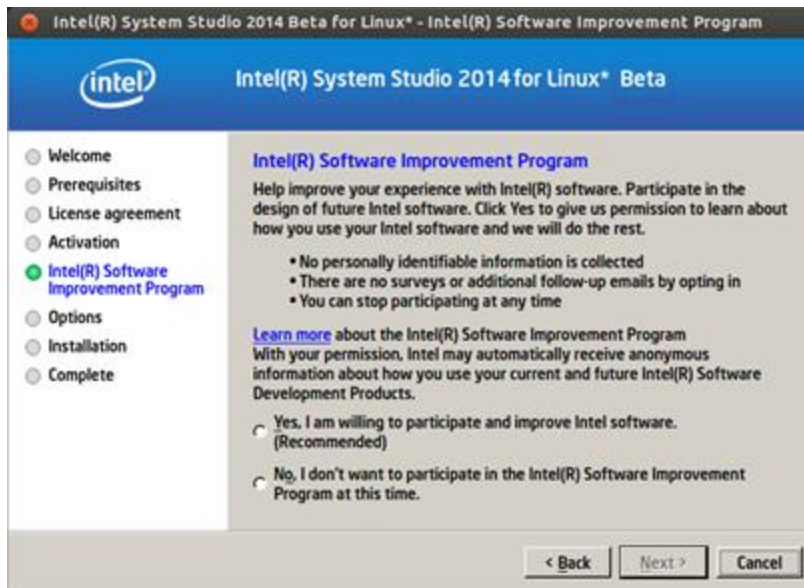
7. The installation routine checks for the availability of all product dependencies. Please address these dependencies, if a warning message appears.
8. Afterwards you will be asked to read the end-user license agreement for the tool suite.



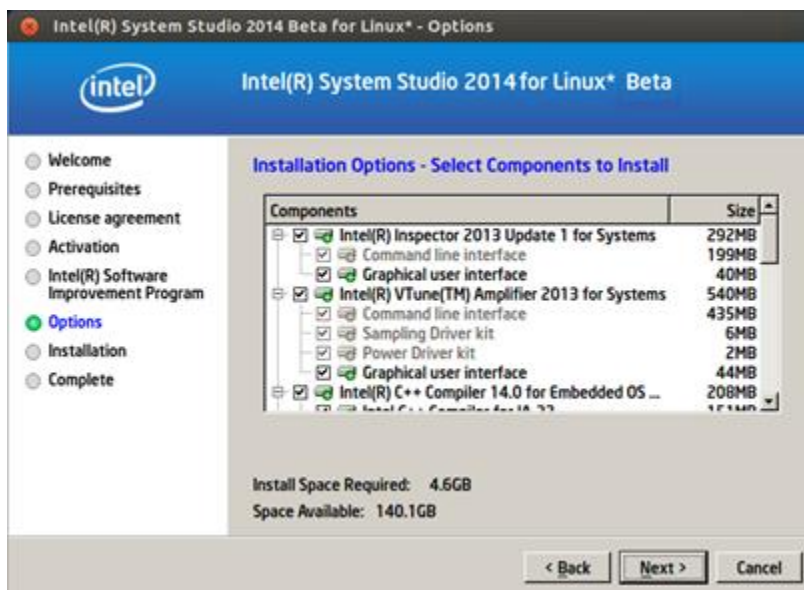
9. When asked whether you would like to activate and install your product select one of the options provided depending on whether you have a license file available or not. If there is already a valid license file available and installed on your system, the installation routine will recommend to simply use the existing license file. If you do not have access to the internet at the time of installation, select the alternative activation option.



10. The next screen will ask you whether you would like to participate in the Intel® software Improvement Program. This will help us identify opportunities for product improvement.



11. The following screen lets you review your installation options, if you would like to only install a specific tool suite component, and change the components settings.



12. Select the installation folder.
13. During the Intel® System Studio 2014 for Linux* installation you will see installation messages for the components you selected.
14. In a final step you can decide to have the installation program do the Eclipse* CDT integration of Intel® System Studio for you.

15. After the installation is complete an installation summary screen will be displayed.

16. To setup the environment for the Intel® C++ Compiler and integrate it correctly with the build environment on your Linux host, execute the following command:

```
> source <install-dir>/bin/iccvars.sh <arg>
```

where <install-dir> is the directory structure containing the compiler /bin directory, and <arg> is one of the following architecture arguments:

ia32: Compilers and libraries for IA32 architectures only

intel64: Compilers and libraries for Intel® 64 architectures only

Notes

5.1.2 Silent Install

For information on automated or “silent” install capability, please see <http://intel.ly/ngVHY8>.

5.1.3 Selecting the provided GDB

To use the provided GNU* Project Debugger GDB instead of the default GDB debugger provided with the default GNU* tools installation of your distribution, please source the following debugger environment setup script:

```
<install-dir>/system_studio_2014.0.xxx/debugger/gdb/bin/  
debuggervars.sh
```

Development target package installation

The target package system_studio_target.tgz contains Intel® C++ Compiler runtime libraries, the Intel® VTune™ Amplifier Sampling Enabling Product (SEP), target components for the Intel® VTune™ Amplifier Data Collector, target components for the Intel® Inspector, and the xdbntf.ko used by the Intel® JTAG Debugger to export Linux* dynamically kernel module memory load information to host.

To install it follow the steps below

1. Unpack the tool suite package in a directory to which you have write access.

```
> tar -zxvf l_cembd_b_2014.0.xxx.tgz
```
2. On the target device unpack the file in a directory to which you have write access

```
> tar -zxvf system_studio_target.tgz
```
3. Add the compiler runtime libraries that you find in
../system_studio_target/compiler/lib/ia32 to your target environment search path.
4. For the dynamic kernel module load export feature follow the instructions found at
../system_studio_target/xdb/kernel-modules/xdbntf/read.me.

This is also detailed in the Intel® JTAG Debugger Installation Guide and Release Notes `jtag-release-install.pdf`.

5. For the Intel® VTune Amplifier Sampling Enabling Product (SEP) follow the instructions in
`../system_studio_target/sep_3.10_axeu12_linux_<arch>_public/docs/SEP_Install_Instructions_Linux.txt`.
6. For the Intel® Inspector for Systems follow the instructions in
`../system_studio_target/inspector_2014_for_systems/documentation/`

5.1.4 Intel® Inspector Command line interface installation

If you would like to install the Intel® Inspector command line interface only for thread checking and memory checking on a development target device, please follow the steps outlined below:

1. Unpack the tool suite package in a directory to which you have write access.
`> tar -zxvf l_cemdb_b_2014.0.xxx.tgz`
2. From the thus created directory `../l_cemdb_b_2014.0.xxx/rpm/`, copy the file `system_studio_target.tgz` to the target device using ftp, sftp or scp.
3. On the target device unpack the file in a directory to which you have write access
`> tar -zxvf system_studio_target.tgz`
4. In the thus created directory `../system_studio_target/`, you will find the Intel® Inspector installation at
`../system_studio_target/inspector_2014_for_systems/`
5. Execute the environment configuration script `inspxe-genvars.sh`.
6. The fully functional command-line Intel® Inspector installation can be found in the `bin32` and `bin64` subdirectories for IA32 and Intel® 64 targets respectively.

5.1.5 Intel® VTune™ Collectors Installation on Remote Systems

If you would like to install the Intel® VTune™ Amplifier data collector for power tuning and performance tuning on a development target device, please follow the steps outlined below:

1. Unpack the tool suite package in a directory to which you have write access.
`> tar -zxvf l_cemdb_b_2014.0.xxx.tgz`
2. From the thus created directory `../l_cemdb_b_2014.0.xxx/rpm/`, copy the file `system_studio_target.tgz` file to the target device using ftp, sftp or scp.
3. On the target device unpack the file in a directory to which you have write access
`> tar -zxvf system_studio_target.tgz`
4. In the thus created directory `../system_studio_target/`, you will find the Intel® Amplifier target data collector installation at
`../system_studio_target/vtune_amplifier_2014_for_systems/target`

5. Data collection on both IA32 and Intel® 64 targets is supported.
6. Follow the instructions in Help document in section “User’s guide->Running analysis remotely” for more details, on how to use this utility.

5.1.6 Intel® VTune™ Amplifier Sampling Enabling Product Installation on Remote Systems

If you would like to install the Intel® VTune™ Amplifier Sampling Enabling Product (SEP), please follow the steps outlined below:

1. Unpack the tool suite package in a directory to which you have write access.
`> tar -zxvf l_cemdb_b_2014.0.xxx.tgz`
2. From the thus created directory `../l_cemdb_b_2014.0.xxx/rpm/`, copy the file `system_studio_target.tgz` to the target device using ftp, sftp or scp.
3. On the target device unpack the file in a directory to which you have write access
`> tar -zxvf system_studio_target.tgz`
4. In the thus created directory `../system_studio_target/`, you will find the Intel® VTune Amplifier Sampling Enabling Product at
`../system_studio_target/sep_3.10_axeu12_linux_x32_public` or
`../system_studio_target/sep_3.10_axeu12_linux_x64_public`
5. Follow the instructions in
`../system_studio_target/sep_3.10_linux_<target>/docs/SEP_Install_Instructions_Linux.txt`
 and
`../system_studio_target/sep_3.10_linux_<target>/docs/README`
6. See `SEP_User_Guide.pdf` in the target package for more details on usage.

5.1.7 Intel® Integrated Performance Primitives redistributable shared object installation

If you are using dynamic linking when using the Intel® Integrated Performance Primitives, you will need to copy the relevant Linux* shared objects to the target device along with the application.

A list of the redistributable shared objects can be found at

`/opt/intel/system_studio_2014.0.xxx/Documentation/en_US/ipp`

5.1.8 Intel® Math Kernel Library redistributable shared object installation

If you are using dynamic linking when using the Intel® Math Kernel Libraries, you will need to copy the relevant Linux* shared objects to the target device along with the application.

A list of the redistributable shared objects can be found at

`/opt/intel/system_studio_2014.0.xxx/Documentation/en_US/mkl`

Eclipse* IDE Integration

5.1.9 Installation

The Intel® C++ Compiler and SVEN SDK can be automatically integrated into a preexisting Eclipse* CDT installation. The Eclipse* CDK, Eclipse* JRE and the Eclipse* CDT integrated development environment are not shipped with this package of the Intel® System Studio. The Eclipse* integration is automatically offered as one of the last steps of the installation process if you run `install.sh` or `install_GUI.sh`. If you decide against integration during an earlier install, simply rerun the Intel® System Studio installer.

When asked point the installer to the installation directory of your Eclipse* install. Usually this would be `/opt/eclipse/`.

The prerequisites for successful Eclipse integration are:

1. Eclipse* 3.7 (Indigo) – Eclipse* 4.3 (Kepler)
2. Eclipse* CDT 8.0 – 8.1
3. Java Runtime Environment (JRE) version 6.0 (also called 1.6) update 11 or later.

5.1.10 Launching Eclipse for Development with the Intel C++ Compiler

Since Eclipse requires a JRE to execute, you must ensure that an appropriate JRE is available to Eclipse prior to its invocation. You can set the `PATH` environment variable to the full path of the folder of the `java` file from the JRE installed on your system or reference the full path of the `java` executable from the JRE installed on your system in the `-vm` parameter of the Eclipse command, e.g.:

```
eclipse -vm /JRE folder/bin/java
```

Invoke the Eclipse executable directly from the directory where it has been installed. For example:

```
<eclipse-install-dir>/eclipse/eclipse
```

5.1.11 Editing Compiler Cross-Build Environment Files

Environment File Support appears under “Intel Tools - Intel® System Studio Tools Environment File” on the menu bar.

For details on the Environment File Editor, please check the

Intel® System Studio Product Guide at

```
<install-dir>/system_studio_2014.0.xxx/documentation/  
embedded_compsupdoc_1.pdf
```


5.1.12 Cheat Sheets

The Intel® C++ Compiler Eclipse* Integration additionally provides Eclipse* style cheat sheets on how to set up a project for embedded use cases using the Intel® C++ Compiler. In the Eclipse* IDE see

Help > Cheat Sheets > Intel C++ Integration

5.1.13 SVEN Trace Viewer integration into Eclipse*

Please refer to the SVEN Trace Viewer Installation chapter in

<install-dir>/system_studio_2014.0.xxx/documentation/en_US/debugger/sven/viewer/
SVEN_VIEWER.pdf

5.1.14 Integrating the provided GDB into Eclipse* for remote debug

Remote debugging with GDB using the Eclipse* IDE requires installation of the C/C++ Development Toolkit (CDT) (<http://www.eclipse.org/downloads/packages/eclipse-ide-cc-linux-developers-includes-incubating-components/indigosr2>) as well as Remote System Explorer (RSE) plugins (<http://download.eclipse.org/tm/downloads/>). In addition RSE has to be configured from within Eclipse* to establish connection with the target hardware.

1. Copy the `gdbserver` provided by the product installation
(`<install-dir>/system_studio_2014.0.xxx/debugger/gdb/
<arch>/<python>/bin/`)
to the target system and add it to the execution PATH environment variable on the target.
2. Configure Eclipse* to point to the correct GDB installation:
 - a. Inside the Eclipse* IDE click on Window>Preferences from the pulldown menu.
 - b. Once the preferences dialogue appears select C++>Debug>GDB from the treeview on the left.
 - c. The GDB executable can be chosen by editing the “GDB debugger” text box. Point to
`<install-dir>/system_studio_2014.0.xxx/debugger/gdb/
<arch>/<python>/bin/`,

where `<arch>` is `ia32` or `intel64` and `<python>` is `py24`, `py26`, or `py27`,
depending on architecture and Python* installation

Installing Intel® XDP3 JTAG Probe

If the `install.sh` installation script is executed using root access, `su` or `sudo` rights, the required drivers will be installed automatically. Root, `su` or `sudo` rights are required for the installation

Installing Macraigor Systems* usb2Demon* Support

To enable support for the Macraigor Systems* `usb2Demon*` device for debugging Intel® Atom™ processor based platforms with the Intel® JTAG Debugger it is necessary to install the Linux* drivers for the Macraigor Systems* `usb2Demon*` device. The driver can be found at http://www.macraigor.com/full_gnu.htm. Please scroll down all the way and download and install the Linux32 `OCDRemote` package recommended at <http://www.macraigor.com/intel/>. Follow the installation instructions posted at the same location closely.

1. For Fedora* host installations download `mcgr-hwsupport-x.x-x.x86_64.rpm` or `mcgr-hwsupport-x.x-x.i386.rpm` from http://www.macraigor.com/full_gnu.htm. For Ubuntu* host installations you can download `mcgr-hwsupport-x.x-x.x86_64.deb` or `mcgr-hwsupport-x.x-x.i386.deb` from the same location. You will find the latest recommended version of the `OCDRemote*` driver set referenced there.

The Intel® JTAG Debugger has been validated for use with the Macraigor Systems* `usb2Demon*` device and `OCDRemote*` 12.00. We recommend using at least version 12.00 of the Macraigor* Systems* driver.

Ensure you have root or `sudo` access rights on your Linux* installation.

2. Install `mcgr-hwsupport-x.x-x.x86_64.rpm` or `mcgr-hwsupport-x.x-x.i386.rpm` using the `rpm -i` command,

or install `mcgr-hwsupport-x.x-x.x86_64.deb` or `mcgr-hwsupport-x.x-x.i386.deb` using the `dpkg -install` command .
3. You should now be able to launch the Intel(R) JTAG Debugger using the respective `start_xdb_MCRG_processorname.sh` startup script. These scripts are being put into the `/opt/intel/system_studio_2014.0.xxx` directory by the tool suite installer as described in the general installation section above.

For further details on how to configure the `OCDRemote*` driver set from Macraigor* Systems, please refer to the full installation instructions at http://www.macraigor.com/eclipse/eclipse_faq.htm.

Removing the Product

To uninstall the Intel® System Studio, change to the `/opt/intel/system_studio_2014.0.xxx` directory and run the `uninstall.sh` uninstall script.

You will be given the following choices:

- 0. All components
- 1. Intel(R) VTune(TM) Amplifier 2014 for Systems
- 2. Intel(R) Inspector 2014 for Systems
- 3. Intel(R) C++ Compiler 14.0 for Linux*
- 4. Intel(R) JTAG Debugger 2014 for Linux*
- 5. SVEN (SDK provided under GNU General Public License v2)
- 6. GNU* GDB 7.5 (Provided under GNU General Public License v3)

- x. Exit [default]

6 Virtualization

6.1.1 Using Intel® VTune™ Amplifier with Virtualization

1. Dynamic binary instrumentation based data collection for hotspot analysis, locks&waits analysis and concurrency analysis have been tested with VMWare* 5.1.
2. Event based Sampling and Sampling with stacks data collection is functional inside a Virtual machine only for architectural events.
Please see the [Performance Monitoring Unit Sharing Guide](http://software.intel.com/file/30388/) at <http://software.intel.com/file/30388/> for more details on the supported architectural events
VMWare* 5.1 and KVM* 1.0.50 have been verified to support this.
3. Power and Frequency data collection is not supported in a virtualized environment.

6.1.2 Using GDB to debug applications running inside a virtual machine

Using GDB for remotely debugging an application running inside a virtual machine follows the same principle as remote debug using the gdbserver debug agent.

The only additional step is to ensure TCP/IP communication forwarding from inside the virtual machine and making the ip address of the virtual machine along with the port used for debug communication visible to the network as a whole.

Details on how to do this setup can be found on [Wikibooks*](http://en.wikibooks.org/wiki/QEMU/Networking) (<http://en.wikibooks.org/wiki/QEMU/Networking>)

The basic steps are as follows

1. Install QEMU, the KQEMU accellerator and bridge-utils

```
$ su -
```

```
$ yum install qemu bridge-utils
```

2. Creating the image for the guest OS

For best performance, you should install your guest OS to an image file. To create one, type:

```
$ qemu-img create filename size[ M | G ]
```

where **filename** is going to be the name of your image, and **size** is the size of your image with the suffix 'M' (MB) or 'G' (GB) right after the number, no spaces.

```
$ qemu-img create Linux.img 10G
```

3. Configuring network for your guest OS

Put the following contents into **/etc/qemu-ifup**:

```
#!/bin/sh
#
# script to bring up the device in QEMU in bridged mode
#
# This script bridges eth0 and tap0. First take eth0 down, then bring it up with IP 0.0.0.0
#
/sbin/ifdown eth0
/sbin/ifconfig eth0 0.0.0.0 up
#
# Bring up tap0 with IP 0.0.0.0, create bridge br0 and add interfaces eth0 and tap0
#
/sbin/ifconfig tap0 0.0.0.0 promisc up
/usr/sbin/brctl addbr br0
/usr/sbin/brctl addif br0 eth0
/usr/sbin/brctl addif br0 tap0
#
# As we have only a single bridge and loops are not possible, turn spanning tree protocol off
#
/usr/sbin/brctl stp br0 off
#
# Bring up the bridge with IP 192.168.1.2 and add the default route
#
/sbin/ifconfig br0 192.168.1.2 up
/sbin/route add default gw 192.168.1.1
#stop firewalls
/sbin/service firestarter stop
/sbin/service iptables stop
```

The bold values can be changed. Please change the IP's to show your setup - The first bold is a comment, so it doesn't really matter. The second bolded value is the IP identical to the one assigned to your computer (means you'll need static IPs so you can predict your IP) and the third and last is your gateway.

Now, put this into `/etc/qemu-ifdown`:

```
#!/bin/sh
#
# Script to bring down and delete bridge br0 when QEMU exits
#
# Bring down eth0 and br0
#
/sbin/ifdown eth0
/sbin/ifdown br0
/sbin/ifconfig br0 down
#
# Delete the bridge
#
/usr/sbin/brctl delbr br0
#
# bring up eth0 in "normal" mode
#
```

```
/sbin/ifup eth0
#start firewalls again
/sbin/service firestarter start
/sbin/service iptables start
```

Make the scripts executable so QEMU can use them:

```
$ su -

$ chmod +x /etc/qemu-if*

$ exit
```

4. Installing the guest OS

Type the following to start the installation:

```
$ su

$ /sbin/modprobe tun

$ qemu -boot d -hda image.img -localtime -net nic -net tap -m 192
-usb -soundhw sb16 -cdrom /dev/hdc;/etc/qemu-ifdown
```

Where **image.img** was the name you gave to your image earlier. I'm also assuming /dev/cdrom is your CD drive - if it's not, then please change it to the correct device. After the install is complete, proceed to step 5.

5. Making the run script & running at will

The last step is to create the QEMU start script and from there on you can run your guest OS. Create this file - called **qemustart** - in the same directory as your image:

```
#!/bin/sh

su -c "/sbin/modprobe tun;qemu -boot c -hda image.img -localtime
-net nic -net tap -m 192 -usb -soundhw sb16;/etc/qemu-ifdown"
```

Where **image.img** was the name given to the image earlier.
Last step - make the startup script executable:

```
$ chmod +x /path/to/qemustart
```

7 Issues and Limitations

Known Issues and Limitations

For known issues of individual Intel® System Studio components please refer to the individual component release notes. Their location in the installed product can be found in chapter 2:

[Technical Support and Documentation](#)

7.1.1 Offline installation with non-root access

When installing as a current non-privileged user, activation option “Use a different computer with internet access” does not work. To activate the product without internet connection enabled, use a license file or use root or sudo privileges instead

7.1.2 Running online-installer behind proxy server fails

Running online-installer behind proxy server produces error: "Connection to the IRC site cannot be established". Please see the [Installation Notes](#) for more details

7.1.3 The online-installer has to be run with sudo or root access

If the online-installer is run as a regular user the installation will hang in step 6 of the installation. Please see the [Installation Notes](#) for more details.

7.1.4 Yocto Project* Application Development Toolkit 1.2 (ADT) has only limited support for static linking

When using the Application Development Toolkit 1.2 and specifying the build option – static in a makefile to enforce a static linking model for libraries, the build may fail with unresolved library dependencies.

Intel® VTune™ Amplifier Power and Frequency Analysis

7.1.5 /boot/config-`uname -r` file must be present on platform

In order to enable CPU power data collection please make sure your environment does have a file named /boot/config-`uname -r` located in your /boot/config directory

If there is no such file you should run the following command:

```
$ cat /proc/config.gz | gunzip - > /boot/config-`uname -r`
```

7.1.6 Power and Frequency Analysis only supported on Intel® Core™ Processor Family

Power and frequency analysis currently requires at least a 2nd generation Intel® Core™ Processor Family based platform.

Sampling Collector (SEP) for Intel® VTune™ Amplifier Usage with Yocto Project*

7.1.7 Building Sampling Collector driver on host Linux* system

Additional kernel utilities are required for building drivers and kernel modules need to be present in the kernel source tree. The following utilities need to be manually added to the standard Yocto Project* 1.2 kernel build tree: viz, recordmcount, fixdep, and modpost.

Software Visible Event Nexus (SVEN)

7.1.8 SVEN SDK is not targeting Intel® Atom™ Processor CE series

The SVEN SDK included in this distribution is not intended for use with the Intel® Atom™ Processor CE41xx, CE42xx, CE53xx, or the Intel® Puma6™ Media Gateway.

To take advantage of all the capabilities provided by SVEN on those platforms, please contact your Intel Field representative.

8 Attributions

This product includes software developed at:

The Apache Software Foundation (<http://www.apache.org/>).

Portions of this software were originally based on the following:

- software copyright (c) 1999, IBM Corporation., <http://www.ibm.com>.
- software copyright (c) 1999, Sun Microsystems., <http://www.sun.com>.
- the W3C consortium (<http://www.w3c.org>) ,
- the SAX project (<http://www.saxproject.org>)
- voluntary contributions made by Paul Eng on behalf of the Apache Software Foundation that were originally developed at iClick, Inc., software copyright (c) 1999.

This product includes updcrc macro,
Satchell Evaluations and Chuck Forsberg.
Copyright (C) 1986 Stephen Satchell.

This product includes software developed by the MX4J project
(<http://mx4j.sourceforge.net>).

This product includes ICU 1.8.1 and later.
Copyright (c) 1995-2006 International Business Machines Corporation and others.

Portions copyright (c) 1997-2007 Cypress Semiconductor Corporation.
All rights reserved.

This product includes XORP.
Copyright (c) 2001-2004 International Computer Science Institute

This product includes software licensed from Macraigor Systems, LLC.
Copyright (c) 2004-2009, Macraigor Systems LLC. All rights reserved.

This product includes software from the book
"Linux Device Drivers" by Alessandro Rubini and Jonathan Corbet,
published by O'Reilly & Associates.

This product includes hashtab.c.
Bob Jenkins, 1996.

9 Disclaimer and Legal Information

The Intel® C++ Compiler, Intel® Integrated Performance Primitives, Intel® Math Kernel Library, Intel® VTune™ Amplifier, and Intel® Inspector for Systems are provided under Intel's End User License Agreement (EULA).

The GNU* Project Debugger, GDB is provided under the General GNU General Public License GPL V3.

The SVEN SDK is provided under the General GNU General Public License GPL V2.

The SVEN Trace Viewer is provided under the Eclipse Public License EPL V1.

The Intel® JTAG Debugger kernel module xdbntf.ko is provided under the General GNU General Public License GPL V2.

Please consult the licenses included in the distribution for details.

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER, AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

UNLESS OTHERWISE AGREED IN WRITING BY INTEL, THE INTEL PRODUCTS ARE NOT DESIGNED NOR INTENDED FOR ANY APPLICATION IN WHICH THE FAILURE OF THE INTEL PRODUCT COULD CREATE A SITUATION WHERE PERSONAL INJURY OR DEATH MAY OCCUR.

Intel may make changes to specifications and product descriptions at any time, without notice. Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The information here is subject to change without notice. Do not finalize a design with this information.

The products described in this document 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.

Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.

Copies of documents which have an order number and are referenced in this document, or other Intel literature, may be obtained by calling 1-800-548-4725, or go to:
<http://www.intel.com/design/literature.htm>

Intel processor numbers are not a measure of performance. Processor numbers differentiate features within each processor family, not across different processor families. Go to:
http://www.intel.com/products/processor_number/

MPEG-1, MPEG-2, MPEG-4, H.261, H.263, H.264, MP3, DV, VC-1, MJPEG, AC3, AAC, G.711, G.722, G.722.1, G.722.2, AMRWB, Extended AMRWB (AMRWB+), G.167, G.168, G.169, G.723.1, G.726, G.728, G.729, G.729.1, GSM AMR, GSM FR are international standards promoted by ISO, IEC, ITU, ETSI, 3GPP and other organizations. Implementations of these standards, or the standard enabled platforms may require licenses from various entities, including Intel Corporation.

BunnyPeople, Celeron, Celeron Inside, Centrino, Centrino Inside, Cilk, Core Inside, i960, Intel, the Intel logo, Intel AppUp, Intel Atom, Intel Atom Inside, Intel Core, Intel Inside, Intel Inside logo, Intel NetBurst, Intel NetMerge, Intel NetStructure, Intel SingleDriver, Intel SpeedStep, Intel Sponsors of Tomorrow., the Intel Sponsors of Tomorrow. logo, Intel StrataFlash, Intel Viiv, Intel vPro, Intel XScale, InTru, the InTru logo, InTru soundmark, Itanium, Itanium Inside, MCS, MMX, Moblin, Pentium, Pentium Inside, skool, the skool logo, Sound Mark, The Journey Inside, vPro Inside, VTune, Xeon, and Xeon Inside are trademarks of Intel Corporation in the U.S. and other countries.

* Other names and brands may be claimed as the property of others.

Microsoft, Windows, Visual Studio, Visual C++, 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.

Copyright (C) 2008–2013, Intel Corporation. All rights reserved.