Allied Vision

APPLICATION NOTE

Installing Vimba under Linux

V2.0.0

Scope of this document

This document describes the following topics:

- Installing Vimba under Linux
- Changing the IP configuration if your GigE camera is in a foreign subnet
- Finding and running Vimba's programming examples
- Compiling the C++ API

Prerequisites

To install Vimba, you need tar and the C runtime library glibc6 (PC: version 2.11 or higher, ARM: version 2.15 or higher).

Installing Vimba

Vimba comes as a tarball. To install it, follow these steps:

1. Uncompress the archive with the command `tar -xf ./Vimba.tgz` to a directory you have writing privileges for, e.g.: /opt: `tar -xzf ./Vimba.tgz -C /opt`
   Under this directory, Vimba will be installed in its own folder. In this document, we refer to this path as [InstallDir].
2. **GigE camera users**: Go to [InstallDir]/Vimba_x_x/VimbaGigETL.
   USB camera users: Go to [InstallDir]/Vimba_x_x/VimbaUSBTL.
3. Execute the shell script Install.sh with root privileges (e.g., `sudo ./Install.sh` or `su -c.Install.sh`). If you use GigE and USB cameras, perform this step for both TLs (transport layers).

Install.sh and GenICam environment variables

Executing Install.sh automatically registers the GENICAM_GENTL32_PATH and/or the GENICAM_GENTL64_PATH environment variable in /etc/profile.d, so that every GenICam GenTL consumer can access the Vimba transport layers. If multiple users work with the system, make sure all users can access /etc/profile.d.

4. Log off and log on again.

Now the changes have been applied to the system. You can start to configure Allied Vision cameras and capture images with Vimba Viewer, which can be found in, e.g., [InstallDir]/Vimba_x_x/Tools/Viewer/Bin/x86_32bit/.

Changing the IP configuration in a foreign subnet

To change the IP configuration of a GigE camera in a foreign subnet, run Vimba Viewer with root privileges (e.g., sudo -E [InstallDir]/Vimba_2_1/Tools/Viewer/Bin/x86_32bit/VimbaViewer). Note that running it as root instead of using sudo -E requires that GENICAM_GENTL32_PATH and/or GENICAM_GENTL64_PATH are set for the root as well.

Compiling the programming examples and the C++ API

Vimba includes many programming examples that can be found in, e.g.,

[InstallDir]/Vimba_x_x/VimbaC/Examples/Bin/x86_32bit and

[InstallDir]/Vimba_x_x/VimbaCPP/Examples/Bin/x86_32bit.

Vimba for ARM comes with compiled programming examples. To compile the precompiled programming examples on a PC or to compile the open source Vimba C++ API, you additionally need the packages listed below. Very likely, most of them are already part of your system:

- make
- ffmpeg
- g++ (PC: Version 4.4.5 or higher / ARM: Version 4.7.3 or higher)
- Qt (PC: Version 4.8.4 / ARM: 4.8.5)
- TinyXML (Version 2.5.3 or higher)

Vimba provides all necessary runtime libraries for executing the examples including the Vimba Viewer example.

Exception

The Vimba C++ programming example AsynchronousOpenCVRecorder requires OpenCV 3.0. The example includes a script for compiling and installing OpenCV on Debian-based distributions.

Download OpenCV

http://opencv.org/

Compiling the examples

To compile the examples (not required on ARM systems), go to Build/Make in the VimbaC and VimbaCPP example folders and type ***make*** in your shell.

Disclaimer

For the latest version of this document, please visit our website. All trademarks are acknowledged as property of their respective owners.