

EVE Software Getting Started Document



October 6, 2015

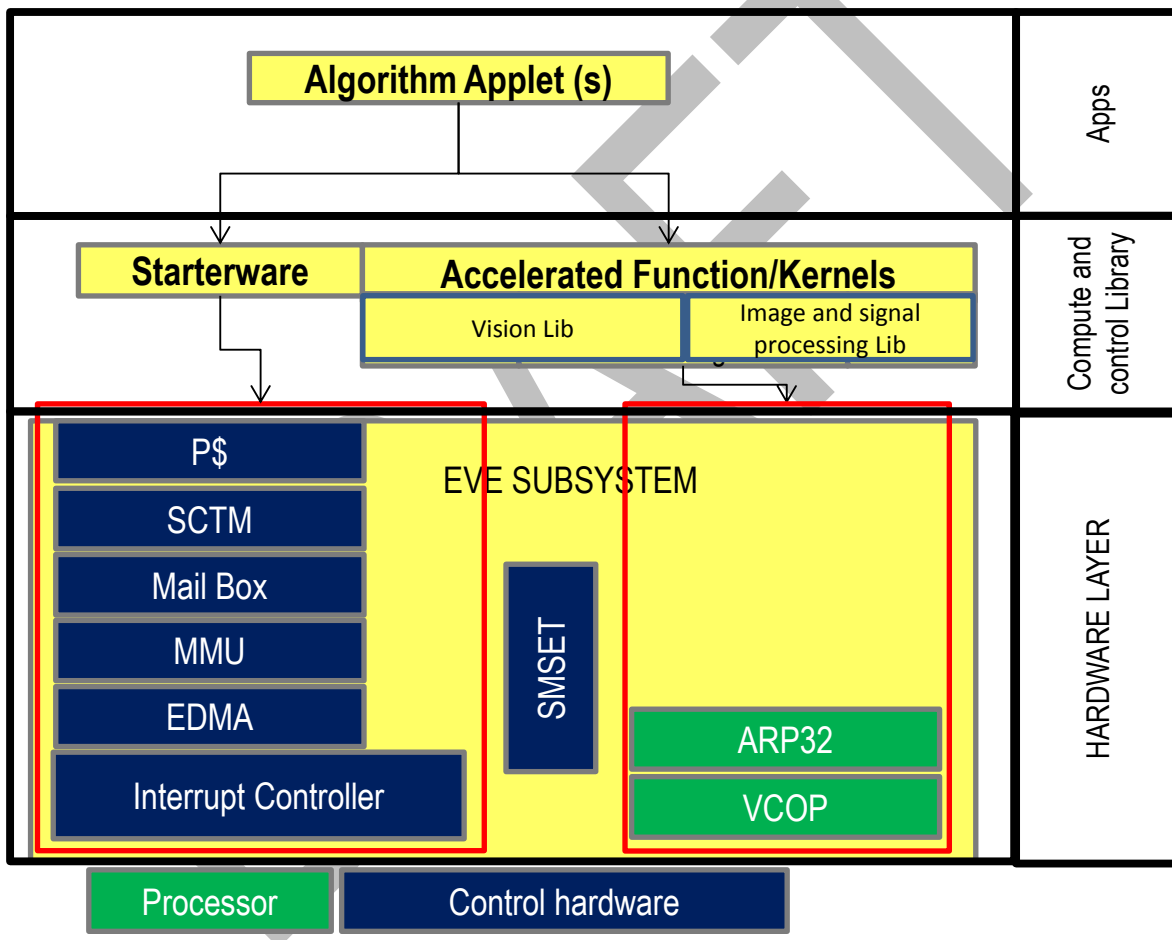
1. Software Components

EVE software has three key components

Starterware – This component contains the APIs to program different control modules of EVE subsystem

Accelerated Functions – It is set of accelerated functions utilizing EVE Vector Co-processor (VCOP) for different applications (vision and imaging). These functions expect input and output in EVE subsystem memory.

Apps – These are high-level applications working on the data in external memory and underneath utilizing starterware and accelerated functions. Example of such applications are resizing of an image, Harris corner detection etc.



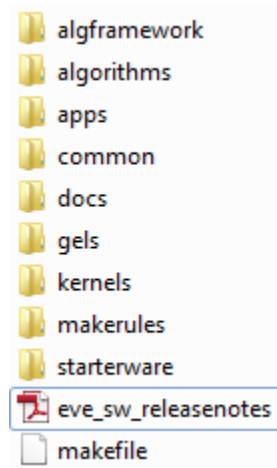
2. Installation Overview

EVE software is released as an installer for both windows and Linux

Windows Installer - eve_sw_<version>_setup.exe (Example for version 01_02_00_01 it is eve_sw_01_02_00_01_setup.exe)

Linux Installer - eve_sw_<version>_setup.bin (Example for version 01_02_00_01 it is eve_sw_01_02_00_01_setup.exe)

After installation below directory, structure will be present on local hard disk



| Sub-Directory | Description |
|---------------|---|
| \algframework | Files related to algorithm framework including BAM. |
| \algorithms | High Level demo algorithm to demonstrate the usage of apps and kernels - example sparse optical flow, ORB |
| \apps | High-level applications, example of such applications are resizing of an image, Harris corner detection, Image Pyramid, Pyramidal Lucas Kanade Tracker etc. |
| \docs | Data sheet for this package. |
| \kernels | Vision and Imaging Kernels optimized for EVE |
| \makerules | Files to aid make |
| \starterware | EVE Starterware drivers, EDMA utilities |
| \gels | Additional gel files required for EVE software usage |
| \common | place holder for common files, example - utility functions, trace etc. |

3. Build Environment and Procedure

As part of EVE software, pre-built libraries are available but for the users who want to modify and build, this section is relevant. The package currently supports only windows build.

3.1 Dependent software components

The following are the software requirements to build and use the EVE software:

- ❑ **Development Environment:** Code Composer Studio (Code Composer Studio v5) version 5.4.0.00091
- ❑ http://processors.wiki.ti.com/index.php/Category:Code_Composer_Studio_v5
- ❑ **ARP32 Code Generation Tools 1.0.7**
- ❑ **C6000 Code Generation Tools 7.4.2**
- ❑ https://www-a.ti.com/downloads/sds_support/TICodegenerationTools/download.htm
- ❑ XDAIS 7.24
- ❑ http://downloads.ti.com/dsps/dsps_public_sw/sdo_sb/targetcontent/xdais/
- ❑ **GNU Make version 3.78.1 or higher**

Available as part of CCS installation (ccsv5\utils\bin\gmake.exe)

- ❑ **GEL file** - Available as part of TDAxx Chip Support Package
 - **TDA2X** : CCS_CSP_TDA2x_SR1.1_NDA_TRM_v^_gels10
 - **TDA3X** : CCS_CSP_TDA3x_SR1.0_NDA_TRM_v^_gels4

Where “^” can be any alphabet. You can also search the same in CDDS by typing *TDA3X*gel* or *TDA2X*gel* and picking the ones mentioned above

3.2 Installation of Dependent software components

3.2.1 Installing ARP32 tools

Installation of ARP32 tools is done by updating CCS tools using P2 server

1. In the CCSv5 main window, select Help->Install New Software...

- This brings up the Available Software window.

2. In the Available Software window, type the EVE P2 installation server URL into the "Work with:" text box, and press Enter.

Server URL for windows:

http://software-dl.ti.com/dsps/dsps_public_sw/sdo_ccstudio/codegen/Updates/p2win32

Server URL for Linux:

http://software-dl.ti.com/dsps/dsps_public_sw/sdo_ccstudio/codegen/Updates/p2linux

The contents of the site should appear in the selection area in the middle of the window.

You should see "CCSv5 Windows updates".

3. Expand "CCSv5 Windows updates".
 - You should see "EVE Compiler Tools v1.0.7".
4. Select the EVE Compiler Tools checkbox, then click "Next>".
 - The Install Details window will be displayed. If the "Details" section says "Cannot complete the install because of a conflicting dependency", try the following steps (The exact wording may differ):
 - a. Click the "Back" button on the Install Details window. This takes you back to the Available Software window.
 - b. Uncheck the box marked "Group items by category" in the lower left portion of the Available Software window.
 - You should now see "REQUIRED CCS Update Installation Tool" as well as "EVE Compiler Tools v1.0.7".
 - c. Select the checkbox for both of these items, then click "Next>".
 - This will bring up the Installation Details window. The window may indicate that your original request has been modified; that is okay.
 - d. Continue through the installation process as described above.
5. Continue through the install process, accept the license agreement, and finish the installation.
6. Restart CCS to execute the installation.

NOTE: The installer for the EVE Compiler Tools is executed in silent mode. If the copy of the XML file called "ARP32.xml" required to activate the EVE tools in CCS fails, you will not be notified. The easiest way to tell if this occurred is to create a New CCS Project and look for EVE in the Device Family drop down menu. If the EVE tools are not shown here, please refer to the README.txt included in the release directory (<CCSInstallDir>\ccsv5\tools\compiler\ arp32_1.0.x). If the directory does not exist, then installation is not completed

User should check for updates by going to Help->Check for Updates to update to the latest version of ARP32 CG tools

3.3 Environment Variables

Build system of EVE software assumes below environment variables being available in the system

| Environment Variable | Description |
|----------------------|---|
| ARP32_TOOLS | Directory pointing to ARP32 compiler (can be found inside CCS installation at following location <CCS_INSTALLATION_DIR>\ccsv5\tools\compiler/ |
| DSP_TOOLS | Directory pointing to DSP compiler (can be found inside CCS installation at following location <CCS_INSTALLATION_DIR>\ccsv5\tools\compiler/) |
| xdais_PATH | Directory pointing to root of XDAIS installation. |
| EVE_SW_ROOT | Directory Pointing to eve root directory, the directory containing starterware folder. The directory should not end with a "/" at the end |

| Environment Variable | Description |
|----------------------|--|
| UTILS_PATH | Directory pointing to utils command like mkdir, rm (can be found inside CCS installation at following location on windows <CCS_INSTALLATION_DIR>/ccsv5/utis/Cygwin |

Please note that the path names in the above environment variables should use only forward slash (/) and should not contain quotes or spaces. As an example, if we want to set ARP32 compiler path as C:/Program Files/Texas Instruments/ARP32_tools, use

```
set ARP32_TOOLS=C:/PROGRA~1/TEXASI~1/ARP32_tools
```

3.4 Build Procedure

Eve Software uses GNU make system for building. To build all modules of EVE Software you will need to define environment variables as described in previous section. After setting the variables use following commands from the root directory of EVE software:

```
gmake
```

This by default will build all modules of EVE software in release mode. If you want to build in debug mode use following command:

```
gmake TARGET_BUILD=debug
```

Output for the applets and algorithms will get generated in elf_out folder present in respective applet/algorithms test folder.

EVE software supports host emulation build in which code can be compiled for visual studio compiler and can directory be run on visual studio. It is important that while using host emulation all the environment variables related to visual studio should be defined for the command window in which you are working. For building EVE Software in host emulation use the following command:

```
gmake TARGET_PLATFORM=PC
```

All the temporary generated files during build are generated in \$(EVE_SW_ROOT)/out folder. You can provide your own directory for temporary generated files by setting BUILD_FOLDER variable.

Output of the host emulation mode will also be generated inside elf_out folder present in respective applet/algorithms test folder. Output file name will contain .exe as extension

It is important to note that before building for host emulation the EVE SW project should have been built at least once for the target to generate some of the kernel header files needed by host emulation

Each kernel can also be build and tested individually. For details on how to build individual kernels kindly refer to eve_kernels_userguide located at kernels/docs/ folder.

Individual applets can also be build using above procedure by going to respective applets test folder and using above commands.

3.5 Memory map / Linker Command File

Applications utilizing EVE library should have the memory section as specified in apps\imagePyramid_u8\test\linker.cmd file.

3.6 Setup EVE in CCS

TDA2X :

1. Goto <CCS Installation directory>/ccs_base/emulation/TDA2X and set EVE_SW_CONFIG macro to 1 in TDA2xx_multicore_reset.gel file
2. In CCS, launch the Target Configuration for TDA2X.
3. Select the 'CortexA15_0' Target -> Do a 'System Reset' -> Select 'Connect Target'.
4. Select the 'CortexA15_0'. Goto 'Scripts' menu -> 'TDA2XX MULTICORE Initialization' -> EVESSCIkEnable_API
5. Select 'ARP32_EVE_x' Target -> Do a 'Connect Target' -> Do a 'CPU Reset'.
6. Any binary can now be loaded to ARP32_EVE_x and executed.
7. To execute EVE starterware mailbox example – additional steps will be required for interrupt mapping – please refer starter ware documentation for the details

TDA3X:

1. In CCS, launch the Target Configuration for TDA3X.
2. Select the 'Cortex_M4_IPU1_C0' Target -> Do a 'System Reset' -> Select 'Connect Target'.
3. Select the 'Cortex_M4_IPU1_C0'. Goto 'Scripts' menu -> 'TDA3XX MULTICORE Initialization' -> EVESSCIkEnable_API
4. Select 'ARP32_EVE_1' Target -> Do a 'Connect Target' -> Do a 'CPU Reset'.
5. Any binary can now be loaded to ARP32_EVE_1 and executed.

3.7 Installing TDA2X simulator in CCS

1. Follow this CDDS link:
<https://cdds.ext.ti.com/ematrix/common/emxNavigator.jsp?objectId=28670.42872.14530.49358>
2. Go to VSS28 Simulator and read the document Instructions to Use Latest Sim Update:
3. URLs for the Simulator Updates
Windows: http://software-dl.ti.com/dsp/dsp_public_sw/sdo_ccstudio/CCSv5.1/Updates/simulation/vayu_ss_sim_csp/p2win32
Linux: http://software-dl.ti.com/dsp/dsp_public_sw/sdo_ccstudio/CCSv5.1/Updates/simulation/vayu_ss_sim_csp/p2linux
4. If CCS5.5 and above is used, the simulator will crash and you will need to follow instructions contained in this packaged to work around the issue :
<https://cdds.ext.ti.com/ematrix/common/emxTree.jsp?objectId=28670.42872.1737.54683&fromContent=true>

Documents

Below table provides the link and description of different documents provided as part of this software package

| Document Link | Purpose |
|---|---|
| Release Notes | Document to capture the Version number, New Features, Bug Fix and Known Limitations of the release |
| Data Sheet | Data sheet for accelerated functions (Kernels) and applications on EVE |
| EVE Starter ware user guide | API Guide for EVE starterware |
| EVE Kernels User Guide | API Guide for accelerated functions (Kernels) on EVE |
| EVE Applets_userguide | User Guide for EVE applets |
| API Guide for BAM | API Guide for framework to accelerate block based processing - BAM (Block-based Acceleration Manager) |

4. Defect Reporting

TI engineer can report defects on this component by following below details.

Defect database is OMAP CQ

Repository: OMAP5.2002.06.00

Component: ALG-EVESW

Release: EVESW_01.x

Below link is Query for all issues on EVE SW. same link can be used to file defects

<http://cqweb.itg.ti.com/cqweb/#/OMAPS.2002.06.00/OMAPS/QUERY/40217650&noframes=true&format=HTML>

External customers can use CSR (Customer Support Request) or assistance from TI field engineer to file the defects

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, enhancements, improvements and other changes to its semiconductor products and services per JESD46, latest issue, and to discontinue any product or service per JESD48, latest issue. Buyers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All semiconductor products (also referred to herein as “components”) are sold subject to TI’s terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its components to the specifications applicable at the time of sale, in accordance with the warranty in TI’s terms and conditions of sale of semiconductor products. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by applicable law, testing of all parameters of each component is not necessarily performed.

TI assumes no liability for applications assistance or the design of Buyers’ products. Buyers are responsible for their products and applications using TI components. To minimize the risks associated with Buyers’ products and applications, Buyers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right relating to any combination, machine, or process in which TI components or services are used. Information published by TI regarding third-party products or services does not constitute a license to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of significant portions of TI information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Resale of TI components or services with statements different from or beyond the parameters stated by TI for that component or service voids all express and any implied warranties for the associated TI component or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Buyer acknowledges and agrees that it is solely responsible for compliance with all legal, regulatory and safety-related requirements concerning its products, and any use of TI components in its applications, notwithstanding any applications-related information or support that may be provided by TI. Buyer represents and agrees that it has all the necessary expertise to create and implement safeguards which anticipate dangerous consequences of failures, monitor failures and their consequences, lessen the likelihood of failures that might cause harm and take appropriate remedial actions. Buyer will fully indemnify TI and its representatives against any damages arising out of the use of any TI components in safety-critical applications.

In some cases, TI components may be promoted specifically to facilitate safety-related applications. With such components, TI’s goal is to help enable customers to design and create their own end-product solutions that meet applicable functional safety standards and requirements. Nonetheless, such components are subject to these terms.

No TI components are authorized for use in FDA Class III (or similar life-critical medical equipment) unless authorized officers of the parties have executed a special agreement specifically governing such use.

Only those TI components, which TI has specifically designated as military grade or “enhanced plastic”, are designed and intended for use in military/aerospace applications or environments. Buyer acknowledges and agrees that any military or aerospace use of TI components which have **not** been so designated is solely at the Buyer’s risk, and that Buyer is solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI has specifically designated certain components as meeting ISO/TS16949 requirements, mainly for automotive use. In any case of use of non-designated products, TI will not be responsible for any failure to meet ISO/TS16949.

Products

| | |
|------------------------------|--|
| Audio | www.ti.com/audio |
| Amplifiers | amplifier.ti.com |
| Data Converters | dataconverter.ti.com |
| DLP® Products | www.dlp.com |
| DSP | dsp.ti.com |
| Clocks and Timers | www.ti.com/clocks |
| Interface | interface.ti.com |
| Logic | logic.ti.com |
| Power Mgmt | power.ti.com |
| Microcontrollers | microcontroller.ti.com |
| RFID | www.ti-rfid.com |
| OMAP Applications Processors | www.ti.com/omap |
| Wireless Connectivity | www.ti.com/wirelessconnectivity |

Applications

| | |
|-----------------------------|--|
| Automotive & Transportation | www.ti.com/automotive |
| Communications & Telecom | www.ti.com/communications |
| Computers & Peripherals | www.ti.com/computers |
| Consumer Electronics | www.ti.com/consumer-apps |
| Energy and Lighting | www.ti.com/energyapps |
| Industrial | www.ti.com/industrial |
| Medical | www.ti.com/medical |
| Security | www.ti.com/security |
| Space, Avionics & Defense | www.ti.com/space-avionics-defense |
| Video & Imaging | www.ti.com/video |

TI E2E Community

e2e.ti.com

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265 Copyright©
2013, Texas Instruments Incorporated