

# Rebuilding the Flash and Boot Utils Package

---

To rebuild the serial flashing tool on a Windows PC, please follow the instructions

## Contents

---

### Download the Flash and Boot Utilities

#### Install and configure the required software

- Cygwin
- Microsoft .Net Framework
- Compiler Tools
  - ARM Compiler Tools (CodeSourcery G++ Lite)
  - C6x Compiler Tools
- CCSv3
- CCSv5

#### Rebuilding the Flash and Boot Utils package for a particular platform

#### Rebuilding the HexAIS utility for OMAPL13x

## Download the Flash and Boot Utilities

---

From Sourceforge (<http://sourceforge.net/projects/dvflashutils/files/>)

For users who are using version 2.36, we strongly recommend upgrading to version 2.40 for a better setup and build experience.

## Install and configure the required software

---

### Cygwin

- Download it here: <http://www.cygwin.com/>
- When installing, add the following packages which are not selected by default:
  - Devel-->make: The GNU version of the 'make' utility
  - Devel-->subversion: A version control subsystem
  - Editor-->vim (or similar)
- After installing, verify that the variables TMP and TEMP both contain /tmp.

```
echo $TMP  
echo $TEMP
```

- Both commands should above should return "/tmp". If some reason they do not you will need to edit the cygwin\home\[user]\.bashrc file to create them:
  - export TMP=/tmp
  - export TEMP=/tmp
- You will need to restart cygwin after this step.

### Microsoft .Net Framework

- Download the latest version of the .NET Framework (4.0 or higher): <http://www.microsoft.com/downloads/details.aspx?FamilyID=9cfb2d51-5ff4-4491-b0e5-b386f32c0992&displaylang=en>
- Add the location of the C# compiler (csc.exe) to the system path environment variable.
  - Typically this is C:\WINDOWS\Microsoft.NET\Framework\v4.0

### Compiler Tools

Note that both ARM and C6x compiler tools are required to build all components of the serial flasher, even for ARM-only parts.

#### ARM Compiler Tools (CodeSourcery G++ Lite)

- Download the ARM compiler tools here: <http://www.codesourcery.com/sgpp/lite/arm/portal/subscription3057>
  - If using v2.40 or later: Edit the Common/build.mak file to have the correct ARM\_TOOLS\_PATH and ARM\_TOOLS\_PREFIX variables
  - If using v2.36 or earlier: Add the bin directory of the ARM cross-compiler tools to the system path environment variable.

#### C6x Compiler Tools

- These are available free of charge: [https://www-a.ti.com/downloads/sds\\_support/TICodegenerationTools/download.htm](https://www-a.ti.com/downloads/sds_support/TICodegenerationTools/download.htm)
  - If using v2.40 or later: Edit the Common/build.mak file to have the correct DSP\_TOOLS\_PATH variable
  - If using v2.36 or earlier: Add the bin directory of the T1 C6000 Code generation tools to the system path environment variable (i.e. "C:\Program Files\Texas Instruments\C6000 Code Generation Tools 6.1.11\bin")

**Note:**

Since cl6x program (C6000 compiler) is a native windows app, it expects windows include and library search paths hence set DSP\_TOOLS\_PATH using cygwin path but DSP\_LIB\_PATH using Windows conventions. The cygwin path will help your make setup find the compiler and the windows library search path will enable the compiler find the rts libraries.

For eg

```
DSP_TOOLS_PATH?=cygdrive/c/ccsv4/tools/compiler/c6000/
DSP_LIB_PATH?="C:\ccsv4\tools\compiler\c6000\lib"
```

**CCSv3**

If you wish to rebuild the CCSv3 projects included in the package from the command-line, you need to have the timake tool in the system path as well. This path is typically <CCSv3 Install Path>\cc\bin.

**CCSv5**

The CCSv3 projects files can be imported into the newer versions of Code Composer Studio, such as CCS v5.1.0. It is recommended to use the default import options when doing so. Do not copy the project files into your workspace. They should be left in place.

## Rebuilding the Flash and Boot Utils package for a particular platform

- For a particular platform, the extracted package will consist of a 'Common' directory and a '<PlatformName>' directory.
- Open a Cygwin prompt, which is like a Unix/Linux prompt under Windows.
  - If using v2.36 or earlier, add necessary components to path

```
export PATH=<arm-compiler-root>/bin:<C6000-Compiler-Root>/bin:$PATH
```

- If using v2.40 or later, edit the ARM\_TOOLS\_PATH, ARM\_TOOLS\_PREFIX, and DSP\_TOOLS\_PATH variables in Common/build.mak as needed for your system
- Enter into the <PlatformName> directory.

```
cd <PlatformName>
```

- Edit device.mak to include only the part number and flash type required, in order to speed up the build process

```
vim device.mak
```

- [Optional]** To rebuild **only the command-line tools** (not the CCS projects), cd into the GNU directory.

```
cd GNU
```

- [Optional]** To rebuild **only the CCS projects** (requires CCS v3), cd into the CCS directory

```
cd CCS
```

- If you're rebuilding **everything** then stay in the <Platform> directory
- Run 'make clean' and 'make'.

```
make clean
make
```

- If you wish to clean-up already built components, run 'make clean' from the path you wish to clean.

## Rebuilding the HexAIS utility for OMAPL13x

HexAIS utility is a boot image generation command line utility provided in the Serial Flash and Boot Utilities package the source for the package can be found in the Common directory in the root directory. The source files are distributed under the AIS and UtilLib directories. The host utility has been written in C#. To rebuild the utility set the environment variables as explained earlier on this wiki. Change the directory to OMAP-L137/GNU/AISUtils/HexAIS and run 'make clean' and 'make'.

**Note:** You will need to build this utility before trying to build the SFH and SFT packages.

<pre> {{ 1. switchcategory:MultiCore=   ■ For technical support on MultiCore devices, please post your questions in the C6000 MultiCore Forum   ■ For questions related to the BIOS MultiCore SDK </pre>	<p>Keystone=</p> <ul style="list-style-type: none"> <li>For technical support on MultiCore devices, please post your questions in the C6000 MultiCore Forum</li> <li>For questions related to the</li> </ul>	<p>C2000=For technical support on the C2000 please post your questions on The C2000 Forum. Please</p>	<p>DaVinci=For technical support on the DaVincoplease post your questions on The DaVinci Forum. Please post only comments about the</p>	<p>MSP430=For technical support on the MSP430 please post your questions on The MSP430 Forum. Please post only</p>	<p>OMAP35x=For technical support on the OMAP please post your questions on The OMAP Forum. Please post only comments about the</p>	<p>OMAPL1=For technical support on the OMAP please post your questions on The MAVRK please post only comments</p>	<p>MAVRK=For technical support on the MAVRK please post only comments</p>	<p>For technical support please post your questions at <a href="http://e2e.ti.com">http://e2e.ti.com</a> Please post on comments about article <b>Rebuild Flash and Boot Package here.</b>}}</p>
--	--	---	---	--	--	---	---	--

(MCSDK), please use the BIOS Forum  
Please post only comments related to the article **Rebuilding the Flash and Boot Utils Package** here.

BIOS MultiCore SDK (MCSDK), please use the BIOS Forum  
Please post only comments related to the article **Rebuilding the Flash and Boot Utils Package** here.

*post only comments about the article **Rebuilding the Flash and Boot Utils Package** here.*

*comments about the article **Rebuilding the Flash and Boot Utils Package** here.*

*article **Rebuilding the Flash and Boot Utils Package** here.*

*about the article **Rebuilding the Flash and Boot Utils Package** here.*  
*Please post only comments about the article **Rebuilding the Flash and Boot Utils Package** here.*

## Links



[Amplifiers & Linear](#)  
[Audio](#)  
[Broadband RF/IF & Digital Radio](#)  
[Clocks & Timers](#)  
[Data Converters](#)

[DLP & MEMS](#)  
[High-Reliability Interface](#)  
[Logic](#)  
[Power Management](#)

[Processors](#)  
▪ [ARM Processors](#)  
▪ [Digital Signal Processors \(DSP\)](#)  
▪ [Microcontrollers \(MCU\)](#)  
▪ [OMAP Applications Processors](#)

[Switches & Multiplexers](#)  
[Temperature Sensors & Control ICs](#)  
[Wireless Connectivity](#)

Retrieved from "[https://processors.wiki.ti.com/index.php?title=Rebuilding\\_the\\_Flash\\_and\\_Boot\\_Utils\\_Package&oldid=157240](https://processors.wiki.ti.com/index.php?title=Rebuilding_the_Flash_and_Boot_Utils_Package&oldid=157240)"

This page was last edited on 24 July 2013, at 17:09.

Content is available under [Creative Commons Attribution-ShareAlike](#) unless otherwise noted.