

ADuCM360 Development Systems Getting Started Tutorial

INTRODUCTION

The [ADuCM360](#) is a fully integrated, 4 kSPS, 24-bit data acquisition system that incorporates dual, high performance multichannel sigma-delta (Σ - Δ) analog-to-digital converters (ADCs), a 32-bit ARM Cortex™-M3 processor, and Flash/EE memory on a single chip.

The [ADuCM360](#) is designed for direct interfacing to external precision sensors in both wired and battery-powered applications. The [ADuCM361](#) contains all the features of the [ADuCM360](#) except for the ADC0, which was removed.

Refer to the [ADuCM360/ADuCM361](#) product page for future updates.

Additional support for the [ADuCM360/ADuCM361](#) is available through the [EngineerZone®](#) website.

GENERAL DESCRIPTION

The [ADuCM360](#) development system allows evaluation of [ADuCM360](#) silicon. This getting started guide introduces the support features and the tools supplied with the evaluation kit. In addition, it shows and describes how to connect the evaluation hardware.

This guide describes the software files that are included on the DVD and FTP site, and how to download them. The [FTP](#) site, should be opened in Windows® Explorer.

This guide works as a tutorial by providing a step-by-step account of how to download evaluation versions of third-party software tools. Instructions are provided on how to load code examples that are supplied on the CD and on the FTP site. These examples demonstrate simple operation of the [ADuCM360](#).

Working through this guide brings the user to the stage where they can start to generate and download their own user code for use in their own unique end-system requirements.

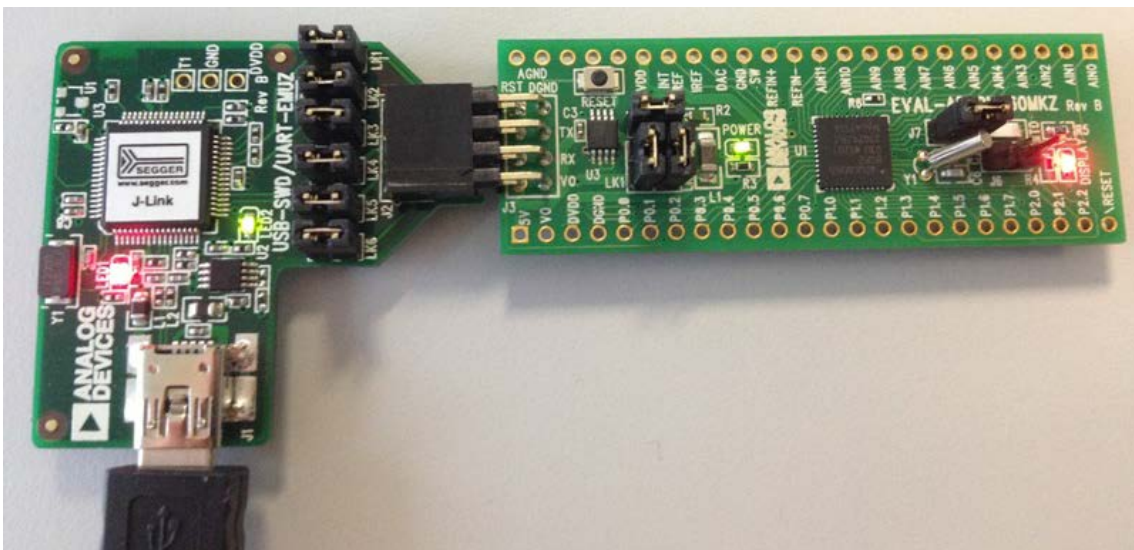


Figure 1. [ADuCM360](#) Mini-Board Connected to Analog Devices, Inc., J-Link OB Emulator

TABLE OF CONTENTS

| | | | |
|---|---|---|----|
| Introduction | 1 | Introduction..... | 6 |
| General Description | 1 | Quick Start Steps | 6 |
| Revision History | 2 | Extra Optional Details on μ Vision..... | 7 |
| Development System Contents..... | 3 | IAR Embedded Workbench IDE..... | 12 |
| Evaluation Board | 3 | Starting IAR Embedded Workbench..... | 12 |
| J-Link OB Emulator | 3 | Quick Start Steps | 12 |
| Connecting the Hardware..... | 3 | Elves..... | 14 |
| Software Installation..... | 4 | Using Elves.exe with μ Vision4 | 14 |
| Software Content Provided..... | 4 | Windows Serial Downloader | 15 |
| Software Installation Instructions..... | 4 | Preparing for Download | 15 |
| Programs Installed | 5 | Downloading Using CM3WSD..... | 15 |
| Keil μ Vision4 Integrated Development Environment | 6 | Running the Downloaded File | 15 |

REVISION HISTORY

4/13—Rev. 0 to Rev. A

Changes to Interface Board, Emulator, and Software Universal
 Replaced All Sections, Tables, and Figures Universal

9/12—Revision 0: Initial Version

DEVELOPMENT SYSTEM CONTENTS

The [EVAL-ADuCM360QSPZ](#) is an evaluation kit for the [ADuCM360](#) and [ADuCM361](#). This kit features a mini-board ([EVAL-ADuCM360MKZ](#)) and an Analog Devices J-Link OB emulator (USB-SWD/UART-EMUZ) that connects to a PC USB port via a USB cable. A comprehensive set of development tools is included on the DVD.

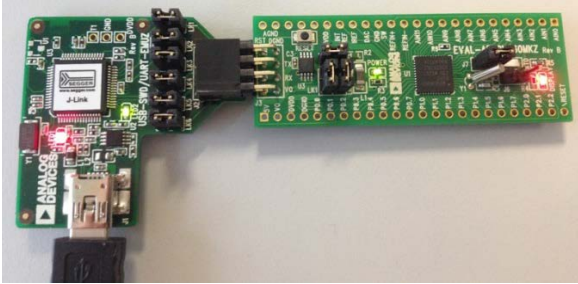


Figure 2. [ADuCM360](#) Mini-Board Connected to J-Link OB Emulator

The development system contains the following:

- An [ADuCM360](#) mini-board
- An Analog Devices J-Link OB emulator
- 1 USB cable
- A DVD.

EVALUATION BOARD

The [EVAL-ADuCM360MKZ](#) mini-board facilitates performance evaluation of the device with a minimum of external components. The board schematic is available after installation of the software in the documentation folder. By default, this is

`\ADuCMxxxV1.3\Documentation\ADuCM360\Evaluation Board\Eval-ADuCM360MKZ`.

J-LINK OB EMULATOR

The J-Link OB emulator provides nonintrusive emulation via a serial wire, and also provides supply and UART communication with the [ADuCM360](#) mini-board. Figure 3 shows a top view of the emulator board. J2 connector plugs into the [ADuCM360](#) mini board. The J2 connector pinout is shown in Figure 4.

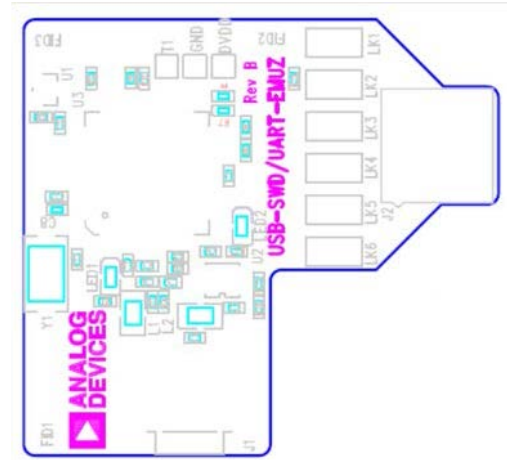


Figure 3. Emulator Top View

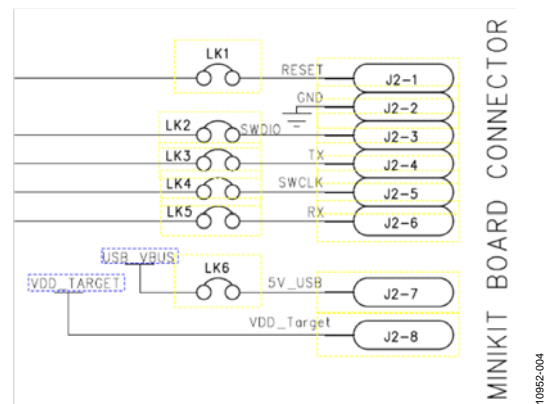


Figure 4. J2 Connector

For downloading and debugging, LK1, LK2, LK4, and LK6 must be inserted. LK3 and LK5 are required to communicate via UART. Required software for the J-Link OB is included in the software installation.

Note that the J-Link OB emulator replaces the J-Link Lite and related interface boards previously shipped with the [ADuCM360](#) development system.

CONNECTING THE HARDWARE

Do not plug in the emulator and mini-board before the software is installed. See the Software Installation section.

SOFTWARE INSTALLATION

A DVD is included in the development system. The DVD content is also available for download on the [FTP](#) site.

SOFTWARE CONTENT PROVIDED

Table 1 shows the tools provided on the DVD.

Table 1. Tools

| Tools | Functions |
|------------------------|--|
| Keil μ Vision | For compiling/debugging and code development, a 32 kB limited version |
| IAR Embedded Workbench | For compiling/debugging and code development, a 32 kB limited version |
| Segger J-Link Software | J-Link software and documentation pack includes USB drivers for the emulator, J-Link Commander, J-Mem, and so on |
| CM3WSD | This utility accepts a hex file and allows it to be downloaded via the USB interface to the ADuCM360 device on your evaluation board |
| Elves | Elves.exe is an application that helps a C programmer choose appropriate functions from Analog Devices libraries and simplifies deciding which values to place in the function parameters. |

SOFTWARE INSTALLATION INSTRUCTIONS

Perform the steps described in this section before plugging any of the USB devices into the PC.

1. Close all open applications.
2. Insert the development system DVD into your DVD drive. Optionally, download the ADuCMxxxV1.3 folder from the FTP and save it to your machine using the same folder structure as on the FTP site. To reduce the download time, consider excluding either the IAR or Keil folders depending on which development environment you chose to use.
3. Double-click on **ADuCMxxxV1.3.exe** and follow the on-screen instructions. A menu displays installation options as shown in Figure 5.

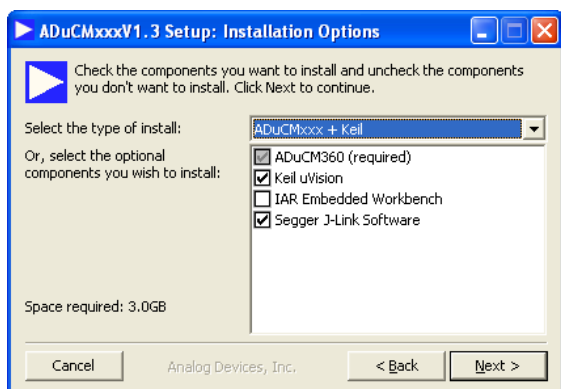


Figure 5. Installation Options

You can choose to install Keil and/or IAR tools as part of the installation, or at later stage by launching

- **mdk470.exe** to install this version of Keil μ Vision.
- **EWARM-CD-6502.exe** to install this version of the IAR Embedded Workbench. The IAR Embedded Workbench requires registering on the IAR website to obtain a free license key.

The Segger J-link software is selected by default in the installation menu. It is advised to leave it selected. This automatically installs the J-Link serial port driver (keep the default settings that appear in the next Segger messages windows).

However, if you decide to unselect the **Segger J-Link Software** install, and install it at a later stage, you will, at that time, need to run the **Setup_JLinkARM_V459d.exe** located under ADuCMxxxV1.3\Segger. Then, select **Install J-Link Serial Port Driver** as shown in Figure 6. Alternatively, you can run **JLinkCDCInstaller_V1.2b.exe** located under ADuCMxxxV1.3\Segger.

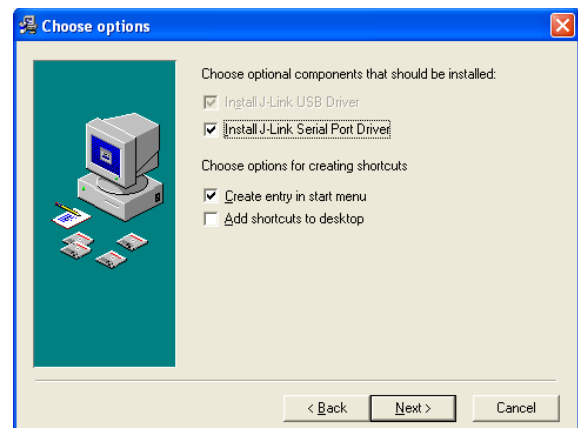


Figure 6. Installing Link Software

Although the development system, the IAR Embedded Workbench®, and the Keil™ software can be installed onto any hard drive and into any directory, for the purposes of simplicity, this user guide assumes it is installed at the default location of **C:\ADuCMxxxV1.3**, **C:\Program Files** and **C:\keil**. In addition, the Keil tools are automatically installed under an ARM directory and are fully compatible with μ Vision3 or tools for 8051.

If you choose the IAR tools, these will be installed by default into **\Program Files\IAR Systems\Embedded Workbench 6.5**.

PROGRAMS INSTALLED

The software described in this section has now been copied or installed.

CM3WSD.exe

The folder `\ADuCMxxxV1.3\Software Tools\CM3WSD` provides an executable called `CM3WSD.exe`. This software accepts a hex file and allows it to be downloaded via the USB interface to the `ADuCM360` device on your evaluation board. You may want to add a shortcut link for this executable to your desktop.

elves.exe

The `\ADuCMxxxV1.3\Software Tools\Elves` folder contains the `elves.exe` files. These files are useful tools that accompany the software function libraries in `\ADuCMxxxV1.3\Code\ADuCM360\common`. Again, no installation is required here, but you may want to add a shortcut link for this executable to your desktop.

Driver

The J-Link OB emulator requires a driver, which is installed automatically when the **Segger J-Link Software** is selected (see Step 3 of the Software Installation Instructions section). At this point, check that the driver is installed correctly.

Plug in the emulator and check the device manager (see Figure 7).

Check that it appears in the Windows Device Manager in both the communications port and the USB controllers lists.

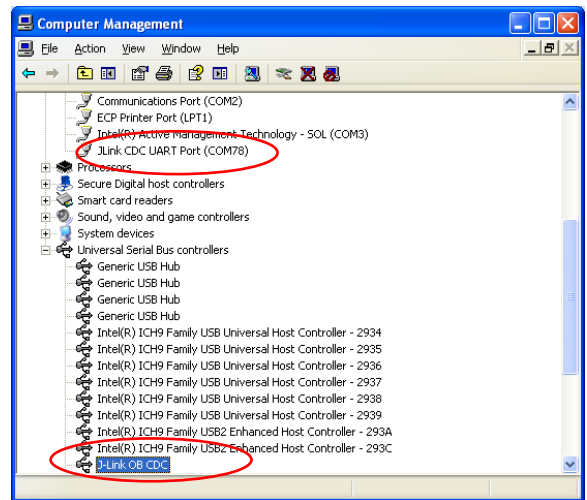


Figure 7. Device Manager

The J-Link OB emulator drivers `JLinkCDCInstaller_V1.2b.exe` can be installed from the `ADuCMxxxV1.3\Segger` folder, if J-Link does not appear in the device manager as shown in Figure 7.

Note that the drivers for J-Link Lite and interface boards previously shipped in the `ADuCM360` development system are available [online](#).

KEIL μ VISION4 INTEGRATED DEVELOPMENT ENVIRONMENT

INTRODUCTION

The μ Vision4 Integrated Development Environment (IDE) integrates all the tools necessary to edit, assemble, and debug code. The ADuCM360 development system supports nonintrusive emulation limited to 32 kB code. This section describes the project setup steps in order to download and debug code on an ADuCM360 evaluation system. Analog Devices recommends using the J-Link debugger driver.

QUICK START STEPS

From the **Start** menu, choose **All Programs>Keil μ Vision4**. This loads the μ Vision4 IDE. The μ Vision4 executable is located at **C:\Keil\UV4\UV4.exe**.

1. To open one of the prepared Keil μ Vision example projects in μ Vision, select **Project>Open Project**.

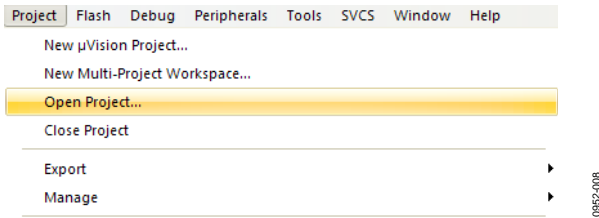


Figure 8. Open an Example Project

2. In the folder **\ADuCMxxxV1.3\Code\ADuCM360\examples\RTD_Demo**, select the file **RTD_Demo.uvproj**. This opens the RTD example project.

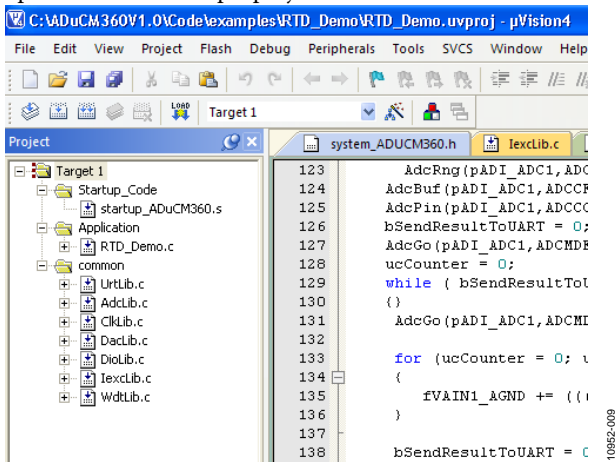


Figure 9. RTD Example Project

To compile and build all files, select the **Build All** icon



Once the build has completed, the code shown in Figure 10 appears.

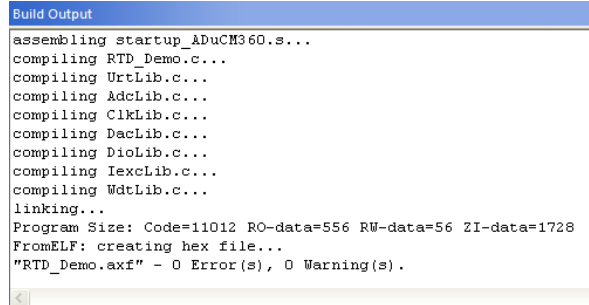


Figure 10. Build Output

3. To download the code to the **EVAL-ADuCM360MKZ** board and begin a debug session, connect the J-Link OB emulator to the **ADuCM360** mini-board and to your PC using the provided USB cable.
4. In μ Vision, click the **Start/Stop Debug** session icon
5. Click **OK** when the window shown in Figure 11 appears.



or press **Ctrl + F5** to start a debugging session.



Figure 11. Evaluation Mode

6. Begin debugging your source code.

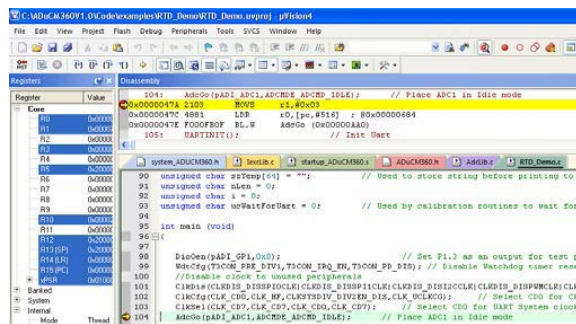


Figure 12. Debug Source Code

EXTRA OPTIONAL DETAILS ON μ VISION

This section provides a more detailed explanation of the setup described in the Quick Start Steps section. Some users may prefer to setup via the Quick Start Steps section.

Toolbars

Under the **View** menu, two toolbars are available.

- File toolbar
- Build toolbar/Debug toolbar

The **File** toolbar is always available. The **Build** toolbar is active only when the IDE is in edit/compile mode. The **Debug** toolbar is active only in download/debug mode.

Starting a Project

1. From the **Project** menu, select **New μ Vision Project**.

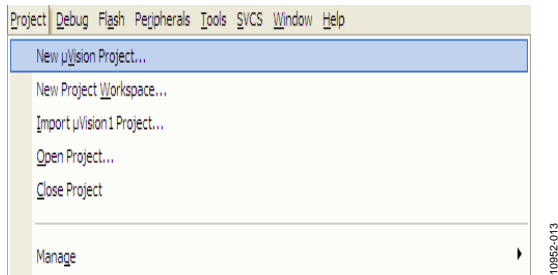


Figure 13. Project Menu

2. Create a new folder (**ADIdemo**). To do so, go to **C:\ADuCMxxxV1.3\Code\ADuCM360\examples\ADIdemo** and enter **Demo** as the project name. If asked to **Select a CPU Data Base File**, select the **Generic CPU Data Base**.

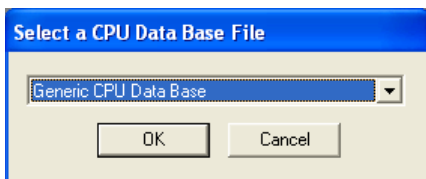


Figure 14. Select a CPU Data Base File

3. Select the **ADuCM360** in the **Generic CPU Data Base**.

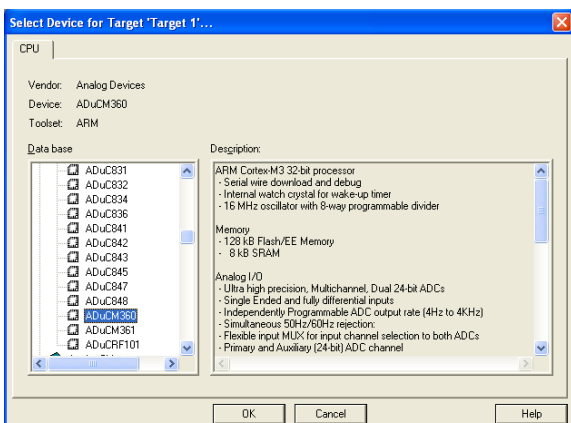


Figure 15. Select ADuCM360

4. Select **No** to the question that appears (see Figure 16). This indicates you are not automatically including the start-up file **startup_ADuCM360.s** to your project.

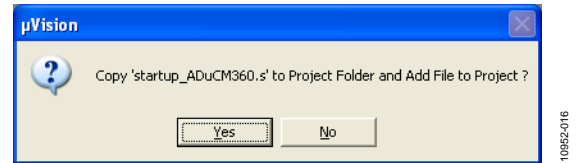


Figure 16. μ Vision Startup Question

Note that it is possible to change the compiler by selecting the **File Extensions, Books and Environment** folder extension.

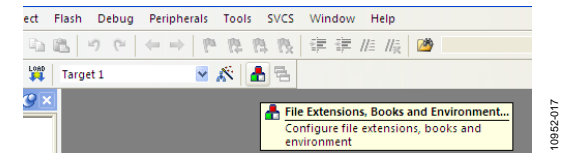


Figure 17. File Extensions, Books and Environment

5. For this demo, select **Use RealView** under the **Folders/Extensions** tab.

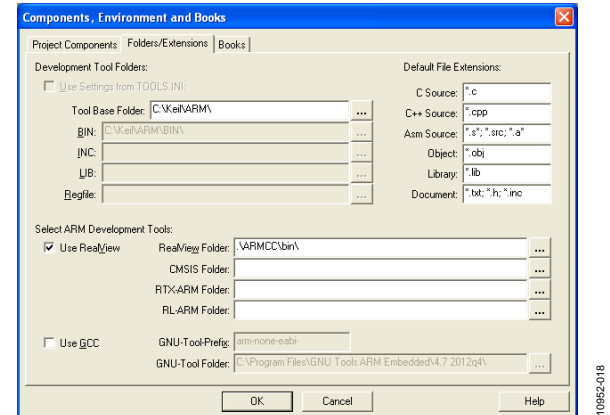


Figure 18. Components, Environment and Books

6. In the project window, right-click on **Target1** and select **Option for Target1** to configure the settings for this project. By default, μ Vision4 uses the RealView compiler.
 - Select the **Target** tab.
 - Ensure the **IROM1** and **IRAM1 Start and Size** fields are filled as shown in Figure 19.
 - Ensure that the **Use MicroLIB** option is enabled.

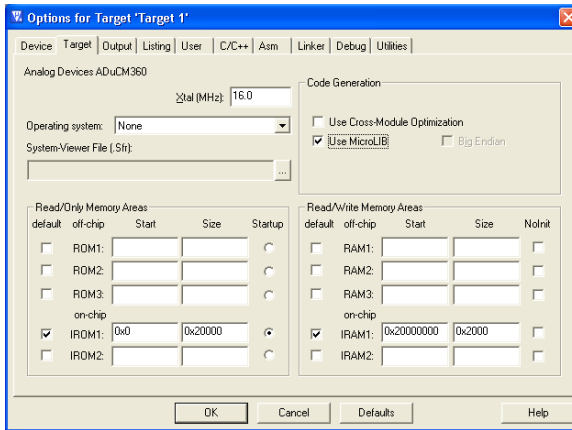


Figure 19. Options for Target

7. Select the **Linker** tab and then select **Use Memory Layout from Target Dialog**.

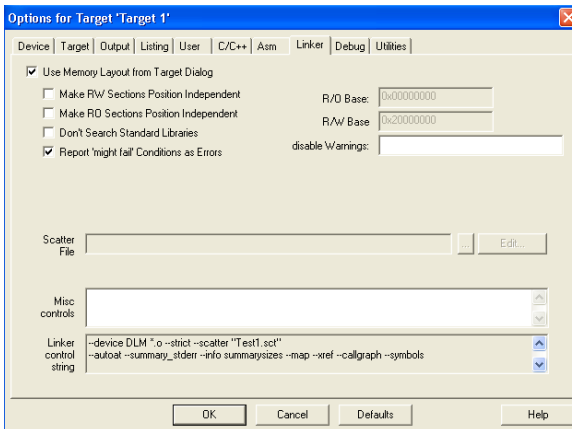


Figure 20. Linker Options

8. In the **Output** tab, select **Create HEX File**. The hex file can be used by the Windows Serial Downloader (CM3WSD). Then, select **OK**.

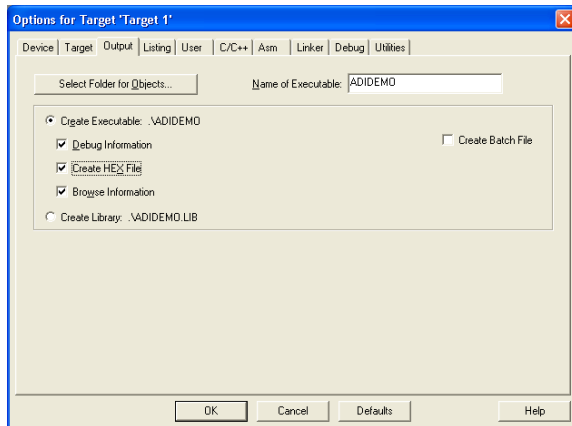


Figure 21. Output Tab

9. Connect the emulator to the **ADuCM360** mini-board and to your PC's USB port using a USB cable.

Note that an LED on the J-Link OB emulator blinks several times before staying on, indicating that the emulator is communicating correctly with the PC.

Configuring the J-Link Debugger Driver

1. In the project window, right-click on **Target1** and select **Option for Target1** to configure the settings of this project.
2. In the **Debug** tab, select **Use:** and then select **J-LINK/J-Trace Cortex**.

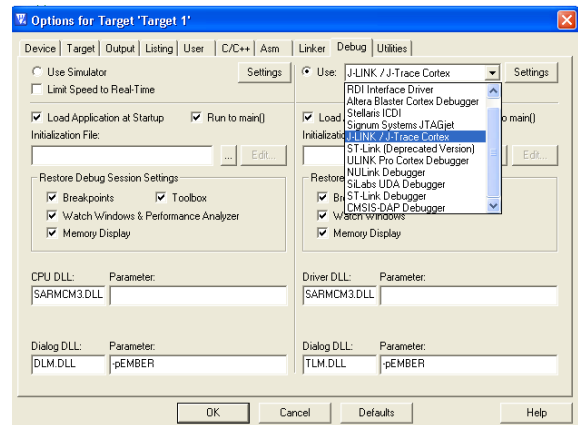


Figure 22. Choose a Debugger

3. Select **Run to main()**.
4. Select **Settings** in Figure 22 and then configure as shown in Figure 23.

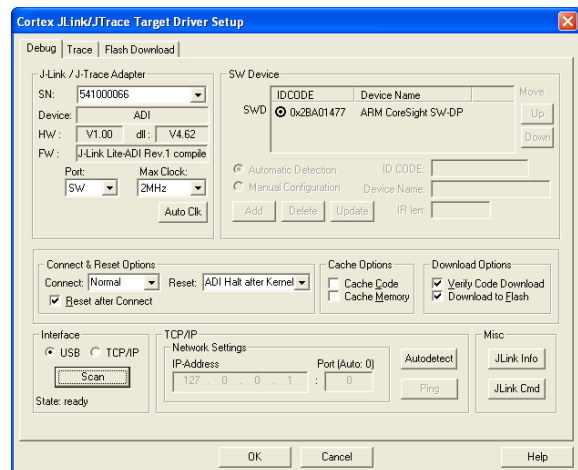


Figure 23. Target Driver Setup

5. Select **OK**.
6. Under **Utilities**, select **Use Target Driver for Flash Programming**. Then, choose **J-LINK/J-Trace Cortex** and select the option **Update Target before Debugging**.

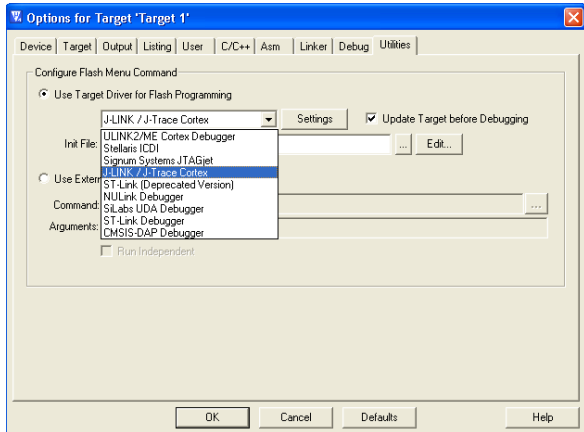


Figure 24. Continuing with Target Driver Setup

7. Select **Settings** to display the dialog box shown in Figure 25.

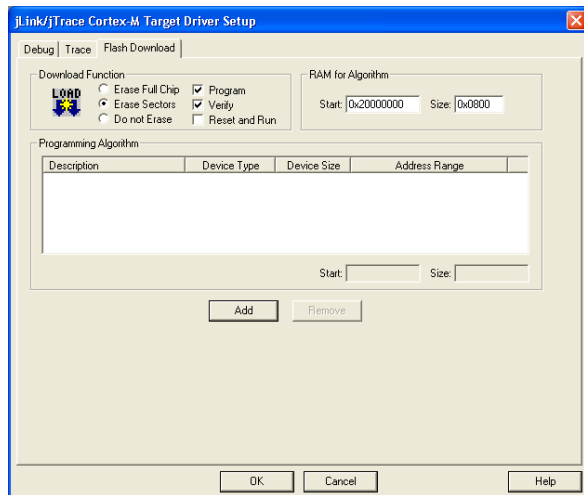


Figure 25. J-Link/J-Trace Cortex Setup

8. Select **Add** to display the window shown in Figure 26.
Select the driver for the generic that you are evaluating.
For this example, use **ADuCM360 128kB Flash** and then select **Add**.

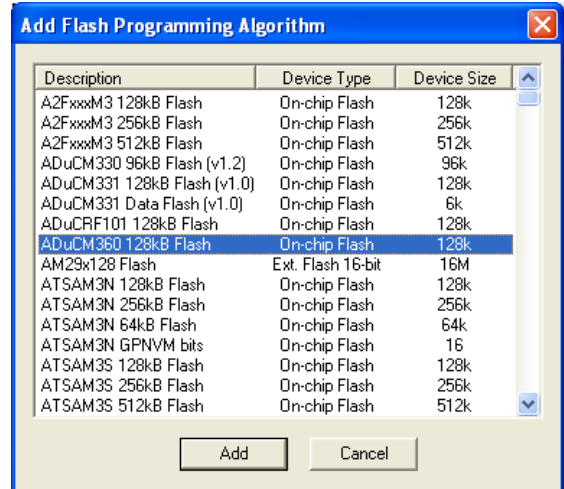


Figure 26. Add Flash Programming Algorithm

9. Select **OK** when the window shown in Figure 27 appears.

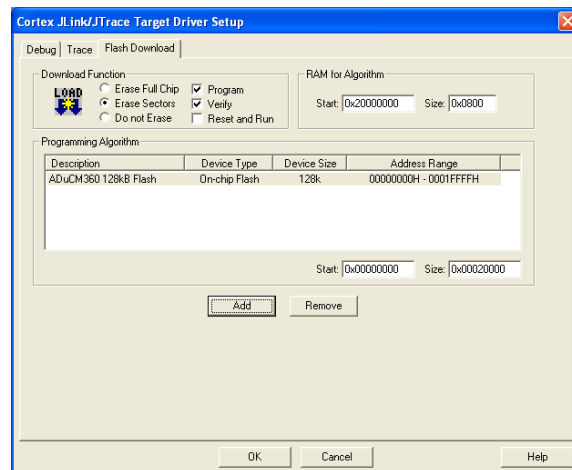


Figure 27. Finishing Target Driver Setup

10. In the **C/C++** tab, add the include path as shown in Figure 28.
11. Select **OK**. All the options should be properly configured to compile, assemble, link, download, and debug using J-Link Lite.

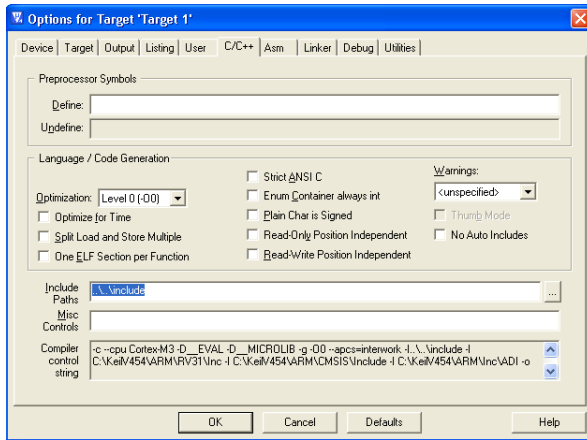


Figure 28. Options for Target Setup

Adding Project Files

All the files relative to the project are in the folder `\ADuCMxxxV1.3\Code\ADuCM360\examples\Adc`.

Copy the file

`\ADuCMxxxV1.3\Code\ADuCM360\examples\Adc\ADCMeter.c` into the new directory `C:\ADuCMxxxV1.3\Code\ADuCM360\examples\ADIdemo`.

1. To add the files to the project, right-click on the **Source Group** folder in the **Project** window and select **Add Files to Group**.

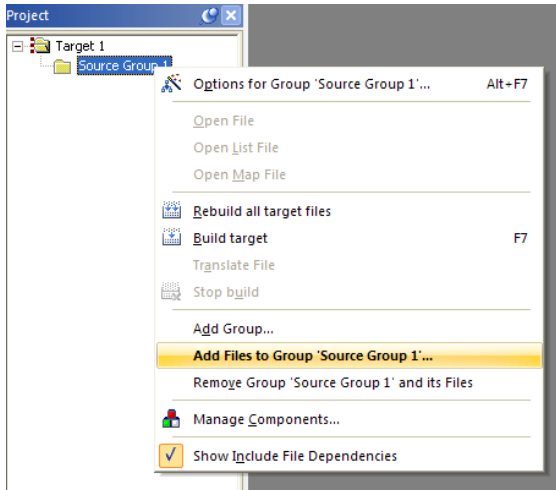


Figure 29. Adding Files to the Project

Note that under **Project>Manage**, the option **Components, Environment, Books** can be used to rename the target and add the file relative to your project.

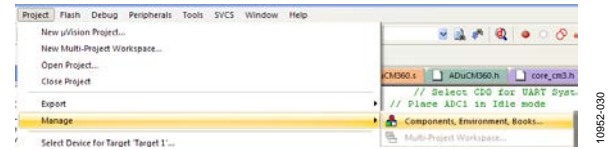


Figure 30. Renaming the Target

2. Add all the files listed in Figure 31 from the directory `\ADuCMxxxV1.3\Code\ADuCM360\common`. The **Startup** file is located in the RealView folder.

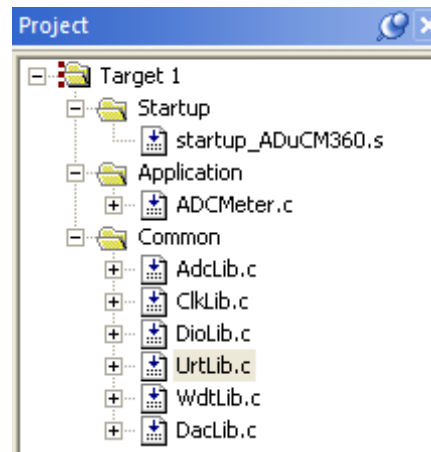




Figure 31. Project Files

3. Double-click on the file name (**ADCMeter.c**) in the **Project** window to open the source file.

Assembling/Compiling Code

To compile/link `ADCMeter.c`, click on the  (translate current file) icon in the toolbar. The file should compile correctly and the information shown in Figure 32 should appear in the Build Output window. If there are errors in your source code, these appear in the status window. To identify the line of code that corresponds to the error, double-click on the error in the Build Output window and an arrow highlights the line of code in error.

Before the code can be downloaded to the [ADuCM360](#), the entire project must be built. This is done by clicking on the  (rebuild all target files) icon on the toolbar. It will also create a `demo.elf` file used by the debugger.

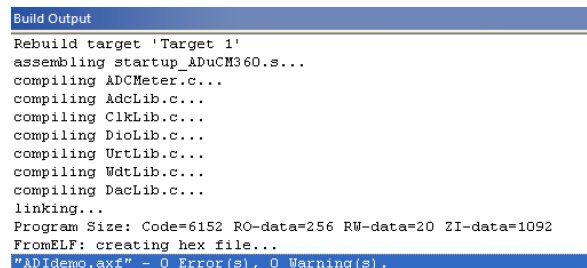



Figure 32. Build Output

Downloading/Debugging Code

Select **Start/Stop Debug** or press the icon  to start debugging (start/stop debug session). The debugger indicates that you are using an evaluation version. Select **OK**.

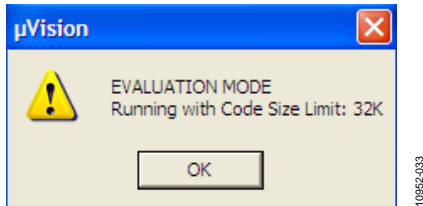


Figure 33. Starting Debugging


1. Close the disassembly window.
2. Go to the **ADCMeter.c** file.
3. Set a breakpoint on the instruction **UARTInit()**. This is done by right-clicking on the line of code and then selecting **insert/remove breakpoint** or by double-clicking to the left of the instruction. Notice that the breakpoint is indicated by a large red dot to the left of the line in Figure 34.

```

51
52 void delay(long int);
53 volatile unsigned char bSendResultToUART = 0; // Flag usec
54 unsigned char s2Temp[64] = ""; // Used to store i
55 unsigned char ucTxBufferEmpty = 0; // Used to indic
56 volatile long ulADC1Result = 0; // Variable tl
57 volatile unsigned char ucComRx = 0;
58 volatile unsigned char ucADCERR = 0;
59 float fVoltage = 0.0; // ADC value coi
60 float fVolts = 0.0;
61
62 int main (void)
63 {
64     unsigned char i = 0;
65     unsigned char nLen = 0;
66
67     pADI_WDT ->T3CON = 0;
68     DioOen(pADI_GPI,0x8); // Set P1.
69     WdtCfg(T3CON_PRE_DIV1,T3CON_IRQ_EN,T3CON_PD_DIS); // D:
70     //Disable clock to unused peripherals
71     ClkDis(CLKDIS_DISSPIOCLK|CLKDIS_DISSPIICLK|CLKDIS_DISI
72     ClkCfg(CLK_CDO,CLK_HF,CLKSYSDIV_DIV2EM_DIS,CLK_UCLKCG);
73     ClkSel(CLK_CD6,CLK_CD7,CLK_CDO,CLK_CD7); // Select
74     AdcGo(pADI_ADC1,ADCMD_ADCMD_IDLE); // Place ADC1
75     UARTINIT(); // Init Uart
76     DACINIT(); // Configu
77     ADC1INIT(); // Setup ADC1
78     AdcGo(pADI_ADC1,ADCMD_ADCMD_CONT); // Start ADC1
79     NVIC_EnableIRQ(ADC1_IRQn); // Enable ADC1 and I
80     NVIC_EnableIRQ(UART_IRQn);

```

Figure 34. ADCMeter.c File

Press the run code button  twice. The program measures the input signal applied across AIN0 and AIN1, converts this to a voltage, and sends this information in an ASCII string to the UART – baud rate 9600-8-N-1.

To stop the code from running, press .

To stop debugging, press .

IAR EMBEDDED WORKBENCH IDE

The IAR Embedded Workbench IDE integrates all the tools necessary to edit, assemble, and debug code. The [ADuCM360](#) development system supports nonintrusive emulation limited to 32 kB code. This section describes the project setup steps in order to download and debug code on an [ADuCM360](#) evaluation system. Analog Devices recommends using the J-Link debugger driver.

STARTING IAR EMBEDDED WORKBENCH

From the **Start** menu, choose **All Programs>IAR Systems>IAR Embedded Workbench for ARM 6.50>IAR Embedded Workbench**. This loads the Embedded Workbench IDE.

QUICK START STEPS

Follow the steps in this section to get up and running with the example code provided with the evaluation software.

These steps use the default driver and compiler settings.

- To open the prepared IAR example projects using the Embedded Workbench tools, select **File>Open>Workspace**. Open the file **ADuCM360.eww** in the directory `\ADuCMxxxV1.3\Code\ADuCM360\examples`.

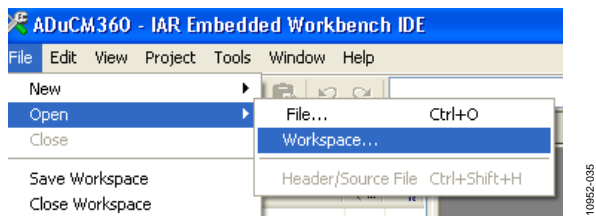


Figure 35. IAR Embedded Workbench

- The workspace opens and the individual projects are launched as shown in Figure 36.

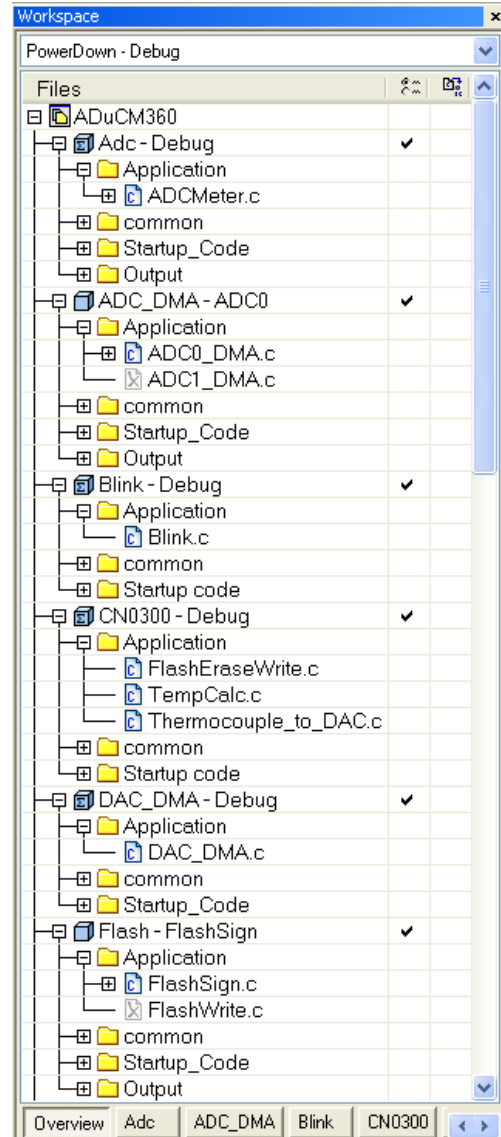


Figure 36. Individual Projects

- To change the selected project, select the required project from the drop-down list.

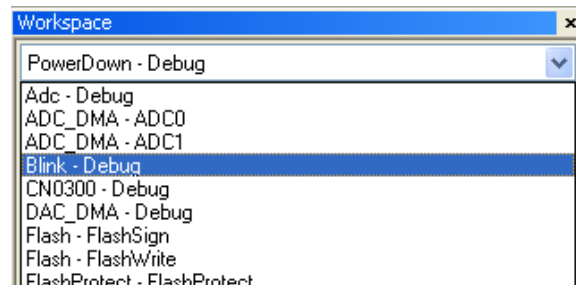


Figure 37. Changing Selected Projects

Choosing, for example, the **Blink** project, toggles the LED connected to P1.3 on the [EVAL-ADuCM360MKZ](#) board.

- To compile all files, select **Project>Rebuild All** files.

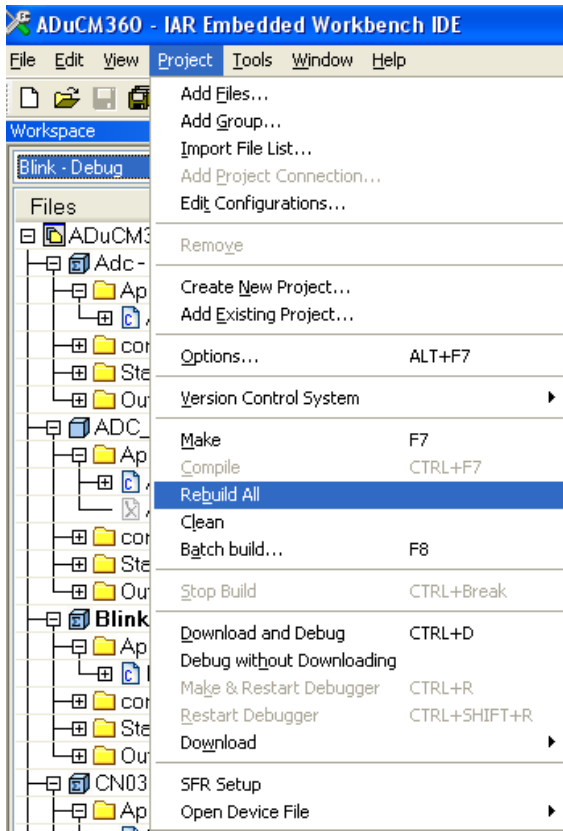


Figure 38. Compiling All Files

- If the build is successful, the information is displayed in the **Build** details window.

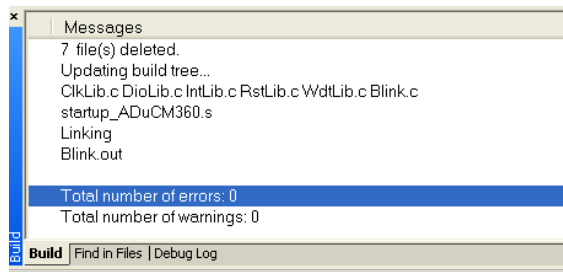


Figure 39. Build Details

- To program the device and to begin debugging the source code, select **Project>Download and Debug**.

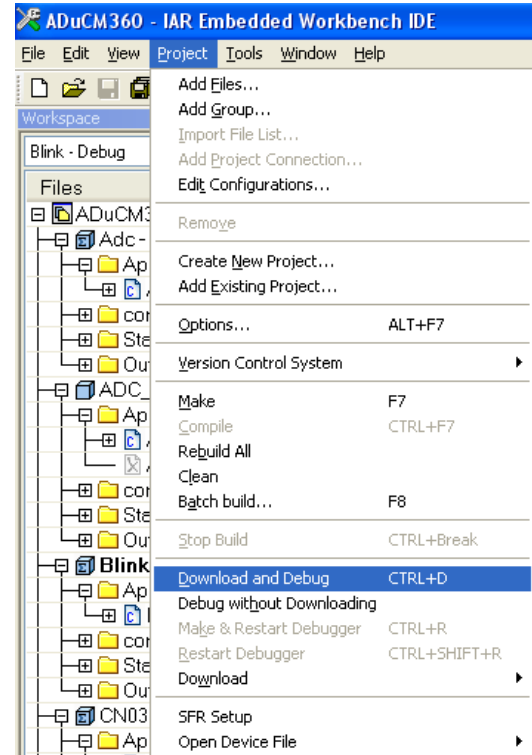


Figure 40. Begin Debugging

This launches the debugger.

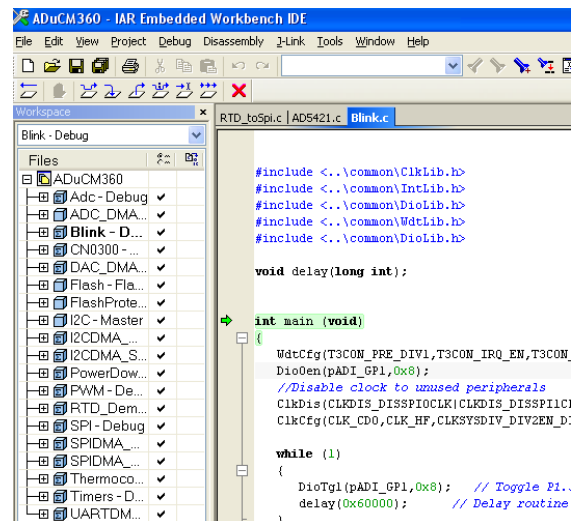


Figure 41. Debugger

- To begin code execution, select the **Go** icon,



You should now see the LED toggle on your **EVAL-ADuCM360MKZ** board.

ELVES

USING ELVES.EXE WITH μ VISION4

Elves is a useful tool for generating simple C function libraries to get started on evaluating any peripheral. All the user needs to do is choose the required parameters for each function and Elves generates the C source code that configures all the appropriate ADuCM360 registers.

1. In the folder, C:\ADuCMxxxV1.3\Software Tools\Elves, open the file **Elves.exe** to launch Elves.

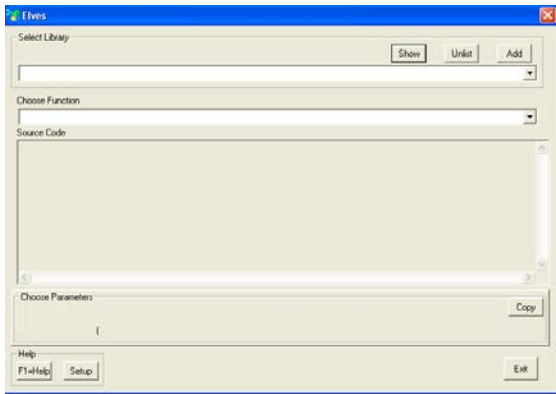


Figure 42. Launching Elves

2. To add a library, click **Add** and go to the directory C:\ADuCMxxxV1.3\Code\ADuCM360\common.

A list of header files is available. Add the header file(s) that you wish to use.

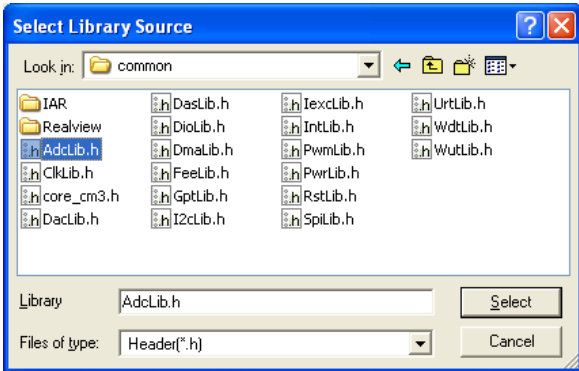


Figure 43. Select Source Library

For example, if the AdcLib.h library is added, the user can generate functions to control the ADC.

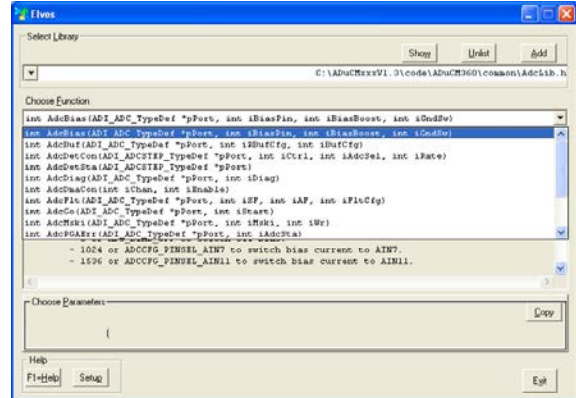


Figure 44. List of Functions

Take, for example, the function **AdcBias**, in the **Choose Function** section as shown in Figure 44. The user configures the parameters to meet their needs and each parameter is explained in the **Source Code** section of the window shown in Figure 45.

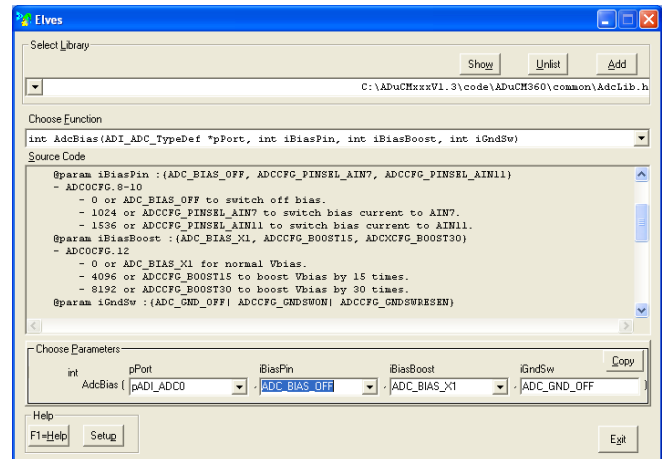


Figure 45. Source Code

3. Once satisfied with the register settings, select **Copy** and then paste this function into your source code in Keil or IAR as shown in Figure 46.

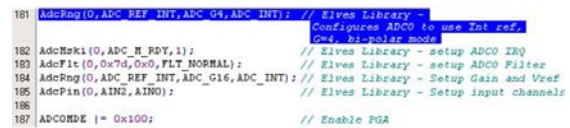


Figure 46. Copying and Pasting Source Code

4. Before using the Help option, click **Setup** and point to the following file: C:\ADuCMxxxV1.3\Software Tools\Elves\help docs\Docs\Start_P.html.

WINDOWS SERIAL DOWNLOADER

The Windows® Serial Downloader for Cortex-M3 based parts (CM3WSD) is a Windows software program that allows a user to serially download Intel Extended Hex files as created by assembler/compiler to the [ADuCM360](#) via the serial port. The Intel Extended Hex file is downloaded into the on-chip Flash/EE program memory via a selected PC serial port.

PREPARING FOR DOWNLOAD

1. Connect the [ADuCM360](#) mini-board ([EVAL-ADuCM360MKZ](#)) to the emulator board, and the emulator board to the PC using a USB cable.
2. Ensure all the links are inserted on both boards.
3. Place the [ADuCM360](#) into serial download mode using the following sequence:
 - Pull P2.2 low.
 - Pull the RESET pin low and then high (float).
 - P2.2 can be left floating once RESET is high.

DOWNLOADING USING CM3WSD

1. In the software tools \CM3WSD folder, open the file [CM3WSD.exe](#).

2. Select the file at `C:\ADuCMxxxV1.3\Code\examples\Adc\AdcExample.hex`.
3. In the **Serial Port** drop-down menu, select the **JLink CDC UART Port** and a baud rate of 38400.
4. Select **Start**. The CM3WSD sends a reset command to the [ADuCM360](#). If the [ADuCM360](#) is in serial download mode, and the COM port between the PC and the mini-board are setup correctly, then the CM3WSD should start downloading the hex file and display a progress bar while the file is downloading. Once the file has been successfully downloaded, the monitor status box is updated with **Flashing Complete Click Reset to run program**.

RUNNING THE DOWNLOADED FILE

Running Using the CM3WSD

Select **Reset** with P2.2 floating or pulled high. The monitor status box updates with the message: **Running**.

Manual Run Option

Pull RESET low, then high (or float) on the mini-board ([EVAL-ADuCM360MKZ](#)) to reset the [ADuCM360](#), with P2.2 floating or pulled high. The program starts running automatically.

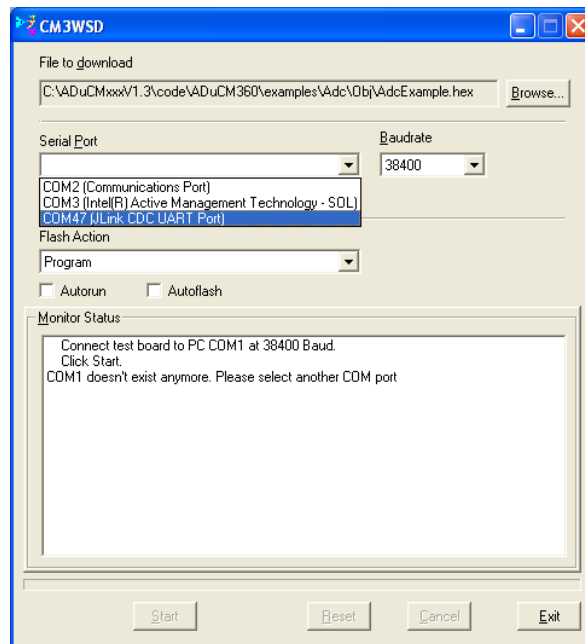


Figure 47. Preparing for Download

NOTES

I²C refers to a communications protocol originally developed by Philips Semiconductors (now NXP Semiconductors).

**ESD Caution**

ESD (electrostatic discharge) sensitive device. Charged devices and circuit boards can discharge without detection. Although this product features patented or proprietary protection circuitry, damage may occur on devices subjected to high energy ESD. Therefore, proper ESD precautions should be taken to avoid performance degradation or loss of functionality.

Legal Terms and Conditions

By using the evaluation board discussed herein (together with any tools, components documentation or support materials, the "Evaluation Board"), you are agreeing to be bound by the terms and conditions set forth below ("Agreement") unless you have purchased the Evaluation Board, in which case the Analog Devices Standard Terms and Conditions of Sale shall govern. Do not use the Evaluation Board until you have read and agreed to the Agreement. Your use of the Evaluation Board shall signify your acceptance of the Agreement. This Agreement is made by and between you ("Customer") and Analog Devices, Inc. ("ADI"), with its principal place of business at One Technology Way, Norwood, MA 02062, USA. Subject to the terms and conditions of the Agreement, ADI hereby grants to Customer a free, limited, personal, temporary, non-exclusive, non-sublicensable, non-transferable license to use the Evaluation Board FOR EVALUATION PURPOSES ONLY. Customer understands and agrees that the Evaluation Board is provided for the sole and exclusive purpose referenced above, and agrees not to use the Evaluation Board for any other purpose. Furthermore, the license granted is expressly made subject to the following additional limitations: Customer shall not (i) rent, lease, display, sell, transfer, assign, sublicense, or distribute the Evaluation Board; and (ii) permit any Third Party to access the Evaluation Board. As used herein, the term "Third Party" includes any entity other than ADI, Customer, their employees, affiliates and in-house consultants. The Evaluation Board is NOT sold to Customer; all rights not expressly granted herein, including ownership of the Evaluation Board, are reserved by ADI. CONFIDENTIALITY. This Agreement and the Evaluation Board shall all be considered the confidential and proprietary information of ADI. Customer may not disclose or transfer any portion of the Evaluation Board to any other party for any reason. Upon discontinuation of use of the Evaluation Board or termination of this Agreement, Customer agrees to promptly return the Evaluation Board to ADI. ADDITIONAL RESTRICTIONS. Customer may not disassemble, decompile or reverse engineer chips on the Evaluation Board. Customer shall inform ADI of any occurred damages or any modifications or alterations it makes to the Evaluation Board, including but not limited to soldering or any other activity that affects the material content of the Evaluation Board. Modifications to the Evaluation Board must comply with applicable law, including but not limited to the RoHS Directive. TERMINATION. ADI may terminate this Agreement at any time upon giving written notice to Customer. Customer agrees to return to ADI the Evaluation Board at that time. LIMITATION OF LIABILITY. THE EVALUATION BOARD PROVIDED HEREUNDER IS PROVIDED "AS IS" AND ADI MAKES NO WARRANTIES OR REPRESENTATIONS OF ANY KIND WITH RESPECT TO IT. ADI SPECIFICALLY DISCLAIMS ANY REPRESENTATIONS, ENDORSEMENTS, GUARANTEES, OR WARRANTIES, EXPRESS OR IMPLIED, RELATED TO THE EVALUATION BOARD INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTY OF MERCHANTABILITY, TITLE, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF INTELLECTUAL PROPERTY RIGHTS. IN NO EVENT WILL ADI AND ITS LICENSORS BE LIABLE FOR ANY INCIDENTAL, SPECIAL, INDIRECT, OR CONSEQUENTIAL DAMAGES RESULTING FROM CUSTOMER'S POSSESSION OR USE OF THE EVALUATION BOARD, INCLUDING BUT NOT LIMITED TO LOST PROFITS, DELAY COSTS, LABOR COSTS OR LOSS OF GOODWILL. ADI'S TOTAL LIABILITY FROM ANY AND ALL CAUSES SHALL BE LIMITED TO THE AMOUNT OF ONE HUNDRED US DOLLARS (\$100.00). EXPORT. Customer agrees that it will not directly or indirectly export the Evaluation Board to another country, and that it will comply with all applicable United States federal laws and regulations relating to exports. GOVERNING LAW. This Agreement shall be governed by and construed in accordance with the substantive laws of the Commonwealth of Massachusetts (excluding conflict of law rules). Any legal action regarding this Agreement will be heard in the state or federal courts having jurisdiction in Suffolk County, Massachusetts, and Customer hereby submits to the personal jurisdiction and venue of such courts. The United Nations Convention on Contracts for the International Sale of Goods shall not apply to this Agreement and is expressly disclaimed.