

ADuC7XXX MicroConverter™ Get Started Guide

A tutorial guide for use with the ADuC7XXX QuickStart™ and ADuC7XXX QuickStart Plus Development Systems

The following products are covered by this guide

ADuC7020/21/22/24/25/26/27/28/29 ADuC7023 ADuC7034/6/9 ADuC7060/61 ADuC7121/22 ADuC7124/26 ADuC7128/29

Version 0.4



A tutorial guide for use with some of the ADuC7XXX Development Systems

CONTENTS:

Introduction	pg 3
(1) Installation	pg 4
(2) Keil µVision3 Integrated Development Environment	pg 7
(3) IAR Embedded Workbench IDE and ROM Monitor	pg 20
(4) The Windows I2C Downloader	pg 23
(5) Elves	pg 24
(6) PLA Tool	pg 26
(7) The Windows Serial Downloader	pg 28
(8) Windows Analog Software Program (WASP)	pg 31
(9) LIN Windows Serial Downloader	pg 34
(10) Installed Documentation and Code Directory	pg 35

Technical Support:

North America and ROW: Europe: China: linear.apps@analog.com euro.linear@analog.com china.support@analog.com



A tutorial guide for use with the ADuC7XXX Development Systems

INTRODUCTION

The following Get Started tutorial guide will bring the user through the various tools that are part of the MicroConverter development systems. This application will install tools required for the ADuC702X, ADuC706X, ADuC7034/6/9 and ADuC712x, ADuC7124/26 generics.

The tools discussed during this Get Started tutorial guide are as follows:

Tool	Executable	Function
Keil µVision3 Integrated Development Environment	MDK380a.exe	μVision3 is a complete IDE (Integrated Development Environment) integrating all the tools necessary to edit, assemble/compile/link and debug ARM/Thumb code and C code, via the JTAG port. A simulator is also available, simulating the peripherals of the ADuC7XXX parts. Non-intrusive emulation is done through JTAG using the mIDAS-Link provided in the QuickStart Plus development system.
IAR Embedded Workbench IDE	EWARM-KS- WEB- 5407.exe.exe	The IAR Embedded Workbench IDE provided in a zip file in the folder, C:\ADuC7xxxV0.2\Applications\IAR is a complete IDE (Integrated Development Environment) integrating all the tools necessary to edit, assemble/compile/link and debug ARM/Thumb code and C code, via the JTAG port . Non-intrusive emulation is done through JTAG using the mIDAS-Link provided in the QuickStart Plus development system. Debugging via UART is done using a ROM monitor included on the CD.
Downloader	I2CWSD.exe	The I2C Downloader is a windows software program that allows a user to download Extended Intel Hex files as created by μVision3 to the MicroConverter via the I2C bus while in circuit. The package also contains a specific version of the I2C Downloader specifically for the ADuC7023 evaluation boards. This Special I2CWSD version does not require an external dongle – it can interface directly with the ADuC7023 evaluation board via the USB interface.
PLA tool	PLAtool.exe	The PLA tool is a graphical tool allowing the user to easily choose the input/output and function of the elements and generating C or assembly code.
Downloader	ARMWSD.exe	The Serial Downloader is a windows software program that allows a user to serially download Extended Intel Hex files as created by μ Vision3 to the MicroConverter while in circuit.
Analysis	WASP7.exe	The Windows Analog Software Program (WASP) is a windows software program for all ADuC702x MicroConverter products that allows analysis of their analog performance.
Software Generation	Elves.exe	Elves.exe is an application that assists a C programmer in choosing appropriate functions from ADI libraries and simplifies deciding what values to place in the function parameters.





(1) Installation

(1) INSTALLATION

Installing from CD or FTP site: (<u>ftp://ftp.analog.com/pub/MicroConverter/ADuC7XXXV0.2/</u>)

- <u>:</u>
- Close all your applications.
 Insert the MicroConverter[®] Development System CD ROM into you CD ROM drive and double-click on the file "ADuC7xxxV02.exe" if installation doesn't start automatically.
- Follow the installation process to the final menu. The following applications will be installed
 - Multidongle Application
 - Segger J-LINK Application
 - o Microsoft .Net Framework (Application is likely already installed on user machine so installation is optional)
 - O PLA Tools
 - Various WSD Applications
- Press "yes" to the above applications. Then follow the on-screen instructions to install the software on your PC.
- Although you can install the MicroConverter[®] Development System, the IAR Embedded workbench and Keil software onto any hard drive and into any directory you wish, for the purposes of simplicity, the rest of this document will assume that you've installed at the default location of C:\ADuC7XXXV0.2, C:\Program Files and C:\keil. Also the Keil tools will automatically be installed under an ARM directory and are fully compatible with uVision3 or tools for 8051.

Tools installation options:

During the installing documents, code examples and utilities provided by ADI, the option to install Microsoft .NET Framework version 2.0 on the user PC will appear. In the event that .NET Framework is not previously installed (To check if the .NET Framework is installed, go to Windows start menu, Control Panel, "Add or Remove Programs" in the control panel and see if "Microsoft .NET Framework" is installed, if .NET Framework is previously installed and an attempt is made to re-install this software, the installer for the .NET Framework will fail), this application should be installed if not previously done so, as it is required for the installation of the ADI PLA tools. In addition, the installation will offer you the option to select the Keil uVision3 tools to be installed automatically by this setup. In addition using the separate IAR Embedded Workbench CD or using the IAR installer located at C:\ADuC7XXXV0.2\Applications\IAR, the IAR tools may be installed.

- Keil uVision3 installation:
- The Keil µVision3 Installation is initiated by ticking the "Yes, I want to install Keil uVision3 tools" as shown below:

InstallShield Wizard	
	InstallShield Wizard Complete
<u> </u>	Setup has finished installing the ADuC7XXX QuickStart Development System Version 0.2 Tools on your computer.
	Setup can install Keil tools (C compiler and C simulator)on your machine.
	💌 Yes, I want to install Keil uVision3 tools.
	The IAR Tools may be installed post the main installation using the files located at C:_ADuC7XXXV0.2_Applications_IAR
	< Back Finish Cancel

A message will appear to confirm that the installs have been selected:



(1) Installation

Welcome to Keil μVisio Release 8/2009	on			Company
This SETUP program ins	stalls:			
RealView Microc	controller Developm	ent Kit V3.80a		
	ay be used to update a previ ake a backup copy before pr			
It is recommended that y	ou exit all Windows program	s before continuing with	SETUP.	
Follow the instructions to	complete the product instal	lation.		
Keil µVision3 Setup				
i nen prinerie e energe		<< Back	Next>>	Cancel

• Press "Next" and follow the on screen instruction.

ANALOG

• If you choose not to install the tools now, you can install them later by double clicking on the file "MDK380a.EXE" in the Applications\Keil folder on the CD.

C:\Keil		
File Edit View Favorites	Tools Help	
🚱 Back 🔹 🕥 🚽 🏂	Search 🕞 Folders	»
Address 🛅 C: Keil		Go
Folders	× Name 🔺	
🖃 🚞 Keil		
🗉 🚞 ARM	C51	
🖽 🫅 C51	UV2	
🛅 UV2	💛 🔂 UV3	
🛅 UV3	TOOLS.INI	
<		>

• When plugging the mIDAS-Link emulator for the first time, the following window appears:

Found New Hardware Wiz	ard
	Welcome to the Found New Hardware Wizard
	This wizard helps you install software for:
	J-Link
	If your hardware came with an installation CD or floppy disk, insert it now.
	What do you want the wizard to do?
	Install the software automatically (Recommended) Install from a list or specific location (Advanced)
	Click Next to continue.
	< Back Next > Cancel

Select "Install from a list or specific location" and press next. Then select "Include this location in the search" and enter the path of the USB driver: C:\ADuC7XXXV0.2\Applications\usbdriver. Press next.

Found New Hardware Wizard
Please choose your search and installation options.
Search for the best driver in these locations.
Use the check boxes below to limit or expand the default search, which includes local paths and removable media. The best driver found will be installed.
Search removable media (floppy, CD-ROM)
✓ Include this location in the search:
C:\ADuC7XXXV0.1\Applications\usbdriver 🔗 Browse
O Don't search. I will choose the driver to install.
Choose this option to select the device driver from a list. Windows does not guarantee that the driver you choose will be the best match for your hardware.
< Back Next > Cancel



(1) Installation

In the hardware installation window, select Continue anyway.



Finally	when	the	following	window	appears	press	the	Finish
button.								

When launching the I2CWSD application for the first time it may be required to point to the directory where the applications USB drivers are installed. If prompted please enter the following location C:\Program Files\Analog Devices Inc\MultiDongle\Driver.

IAR Embedded Workbench installation:

- The IAR Embedded Workbench is available as part of the ADuC7XXX package, however it is not automatically launched as part of the installation procedure. If you choose not to install the IAR Embedded Workbench now, you can install it later by double clicking on the file "EWARM-KS-WEB-5407.exe" at C:\ADuC7XXXV0.2\Applications\IAR. Then the file C:\ADuC7XXXV0.2\Applications\IAR\copy.bat needs to be run after IAR has been installed in order to include the most up to date ADI products.
- To install, follow the on screen instructions. IAR Embedded Workbench will require registering on IAR website to obtain a license key. At this point the latest edition of the IAR tools can be downloaded instead of the version of the tools included on this CD.



(2) KEIL µVISION3 INTEGRATED DEVELOPMENT ENVIRONMENT

The μ Vision3 IDE integrates all the tools necessary to edit, assemble and debug code. The ADuC7XXX Development System supports non-intrusive emulation limited to 32kByte code. This section describes the project setup steps in order to download and debug code on an ADuC7XXX evaluation system. The ADuC7023 is used as the primary example in this section. ADI recommends using the J-Link debugger driver as described in the following section however the RDI debugger driver described below may also be used.

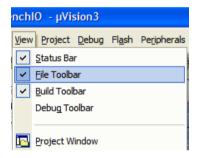
2.1 Starting µVision3

1. From the START menu choose All Programs \rightarrow Keil uVision3. This loads the μ Vision3 IDE. The μ Vision3 executable is located at C:\Keil\UV3\Uv3.exe.

2.2 Toolbars

- 2. Under the VIEW menu, four toolbars are available:
 - a. Status Bar
 - b. File Toolbar
 - c. Build Toolbar
 - d. Debug Toolbar

The Build Toolbar will be active only when the IDE is in Edit/Compile mode. The Debug toolbar will be active only in Download/Debug mode.



2.3 Starting a project

3. From the project menu select New Project as shown below.

<u>P</u> rojec	t <u>D</u> ebug	Fl <u>a</u> sh	Peripherals	Tools	SVCS	<u>W</u> indow	<u>H</u> elp
N	ew µ <u>V</u> isior	n Projec	t				
Ν	ew Projec	t <u>W</u> orks	space				
Į	mport µVis	ion 1 Pro	oject				
0	ipen Proje	ct					
g	lose Proje	ct					
-							
Ν	lanage						+

4. Create a new folder ADIdemo under C:\Keil\ARM\ Examples and enter the name of the project as Demo. You will then be asked to select the device. Select the part number corresponding to your evaluation board under Analog Devices. NEW

Create New F	Project			? 🛛
Save in: 🔀	ADIdemo	• 🕈	۵ 🖻	* 📰 •
File name:	D¦∋mo		- [Save
Save as type:	Project Files (*.uv2)	-	• [Cancel

Options for Target 'Target 1'		
Device Target Output Listing Us	er C/C++ Asm Linker Debug Utilities	
Database: Generic CPU Da	a Base	
Vendor: Analog Devices		
Device: ADuC7023		
Toolset: ARM		
Analog Devices Analog Devices ADUC7019 ADUC7020 ADUC7021 ADUC7022 ADUC7022 ADUC7024 ADUC7025 ADUC7025 ADUC7025 ADUC7026 ADUC7028	ARM/TDMI based controller with 124h 1MSPS 16-Channel ADC, Voltage Comparison, 52XB conchip Rash/tsz Programming (ISP) and 8XB RAM. I2C and SPI setai interface, JTAG port for download Jeduau, 3 Timers, 152th PVM generator, 20 General purpose I/O pins, On-chip Programmable Logic, CPU clock up to 41.78 MHz, On-chip crystal oscillator and On-chip PLL.	
ADuC7029 ADuC7030 ADuC7032 ADuC7032 ADuC7033		2
	OK Cancel Defaults Help	
	OK Cancel Defaults Help	_

The following message appears. Select "No" not to include automatically the startup file startup.s to your project.



µVision.	3 🔀
2	Copy ADuC702x Startup Code to Project Folder and Add File to Project ?
	Yes No

(2) Keil µVision3 Integrated Development Environment

It is po	ssible to change compiler by selecting 🖲 "setup file extensions, books and environment", "folder extension".
•	For this demo, select the RealView Compiler:

Components, Enviror	ment and Books			
Project Components Fo	Iders/Extensions Books			
Development Tool Fol	ders		Default File Exte	nsions:
🔲 Use Settings from			C Source:	.c
Tool Base Folde	r: C:\Keil\ARM\		C++ Source:	.cpp
BIN: C:\Keil\	ARM\BIN\		Asm Source:	".s"; ".src; ".a"
INC:			Object:	
LIB:			Library:	
Regfile:			Document:	".bt; ".h; ".inc
Select ARM Developr	nent Tools			
Use RealView Compiler	RealView Folder: BIN31\			
- Use GNU	GNU-Tool-Prefix: am-uclibc-			
Compiler	GNU-Tool Folder: C:\Program Files\Code	Sourcery\S	ourcery G++ Lite\	
	ОК Са	ncel		Help

5. Right click on Target1 and select "Option for Target1" to configure the settings of this project. By default, uVision3 will use the RealView compiler.

Add "0x80000" into the R/O Base and "0x10000" in to the R/W Base to indicate to the Compiler the Flash and RAM start addresses. Check the button "Use Memory Layout from Target Dialog". a.

Options for Target	Target 1'			
Device Target Out	out Listing User C/C++ Asm	Linker Debug	Utilities	
│ Ma <u>k</u> e RW Se │ M <u>a</u> ke RO Se	out from Target Dialog ctions Position Independent ctions Position Independent Standard Libraries	<u>R</u> /O Base: R/ <u>W</u> Base	0x00080000 0x00010000	,
	fail'Conditions as Errors	<u>d</u> isable Wamings:		
Scatter File	ct			Edit
Misc controls				~
	e DARMAD *.ostrictscatter ''demo.s tsummary_stderrinfo summarysizes -		hsymbols	×
	OK Ca	ncel Defa	aults	Help

b. In the "Output" panel select "Create HEX File". The HEX file can be used by the I2C downloader.



Opt	ions for Target 'Target 1'	
De	vice Target Output Listing User C/C++ Asm Linker Debug Utilities	
	Select Folder for Objects Name of Executable: ADIDEMO	
	 Create Executable: .\ADIDEMO Debug Information Create HEX File 	Create Batch File
	□ Big Endian	
l		
	OK Cancel Defaults	Help

c. In the "Target" panel, ensure the IROM1 and IRAM1 Start and Size tabs are filled in correctly.

Options for Target 'ADC RealView Toolchain'										
Devic	е	Target 0	utput Listing	g User C/C+	+ Asm	Linker	Debug	Utilities		
Analog Devices ADuC7028										
				Xtal (MHz): 0.0	32768		de Genera ARM-Mode		•	
00	erati	ng system:	None		•	L L	Use Cro	ss-Module Opti	imization	
Sp.			Triono			Г	Use Mic	roLIB		
Re	ad/	Only Memo	ory Areas			Read/	Write Mem	iory Areas		
defa	ault	off-chip	Start	Size	Startup	default	off-chip	Start	Size	Nolnit
Г		ROM1:			0		RAM1:			
Г		ROM2:			0		RAM2:			
Г		ROM3:			0		RAM3:			
		on-chip					on-chip			
l I	7	IROM1:	0x80000	0xF800	0	◄	IRAM1:	0x10000	0x2000	
Г		IROM2:			0		IRAM2:			
				ОК	Ca	incel	Defau	lts		Help

d. If you have an emulator, power up the evaluation board using a 9V power supply or, via the USB connection. Connect the evaluation board to the mIDAS-Link and the mIDAS-Link to your PC's USB port using the provided USB cable.
 <u>Note:</u> a green LED on the mIDAS-Link emulator blink a few times before staying on, indicating that the emulator is communicating correctly with the PC.

2.4 Configuring the J-Link Debugger driver

a. If you are using an evaluation board and want to use the RDI driver, then continue to section 2.4.1.

Right click on Target1 and select "Option for Target1" to configure the settings of this project. By default, uVision3 will use the RealView. In the "Debug" panel, select Use ... and choose J-LINK / J-TRACE if you have a mIDAS-Link emulator. Select the settings button and configure as follows.



JLink/JTrace Interface Driver Setup					
JTAG Speed: C Auto Selection C Adaptive Clocking (* KHz) 750	Debug Cache Options:	JTrace Trace Options: Enable Trace Cycle-accurate Max, Trace Samples: 64K			
Reset Strategy: Software for Analog Device A	Info JLink Target				

If your development system doesn't include a JTAG emulator, use the simulator for this exercise.

b. In the "Utilities" panel, select Use Target Driver for Flash Programming", chose "J-LINK / J-TRACE" and tick the option "Update Target before Debugging".

Options for Target 'Tar	get 1'
Device Target Output	Listing User C/C++ Asm Linker Debug Utilities
Configure Flash Menu C	mmand
Use Target Driver for	r Flash Programming
J-LINK /	J-TRACE Settings Vpdate Target before Debugging
Init File:	Edit
C Use External Tool fo	r Flash Programming
Command:	
Arguments:	
🗖 Bun I	ndependent
	OK Cancel Defaults Help

Hit the "Settings" option and the following dialog box will appear -

Dialog			
Download Function C Erase Full Chip C Erase Sectors C Do not Erase	 ✓ Program ✓ Verify ✓ Reset and Run 		Algorithm Dx00010000 Size: 0x0800
Programming Algorithm			
Description	Device Type	Device Size	Address Range
		Start: 🗍	Size:
Add	Remove	ок	Cancel Help



Select "Add" for the next window. Select the driver for the generic that you are evaluating. For this example use "ADuC702X Flash (v1.1) and press "Add"

Description	Device Type	Device Size	1
ADuC702X Flash (v1.1)	On-chip Flash	62k	
ADuC703X Flash 32KB (v1.4)	On-chip Flash	30k	-
ADuC703X Flash 64KB (v1.1)	On-chip Flash	62k	
ADuC703x Flash 96KB (v1.4)	On-chip Flash	94k	
ADuC706X Flash 32KB (v1.0)	On-chip Flash	30k	
ADuC7124 Flash (v1.0)	On-chip Flash	126k	
ADuC712x Flash (v1.2)	On-chip Flash	126k	
ADuC7229 Flash (v1.2)	On-chip Flash	126k	
AM29F160DB Flash	Ext. Flash 16-bit	2M	
AM29F160DT Flash	Ext. Flash 16-bit	2M	
AM29F320DB Flash	Ext. Flash 16-bit	4M	
AM29F320DB Dual Flash	Ext. Flash 32-bit	8M	
AM29F320DT Flash	Ext. Flash 16-bit	4M	
AM29F320DT Dual Flash	Ext. Flash 32-bit	8M	
AM29x033 Flash	Ext. Flash 8-bit	4M	
AM29x128 Flash	Ext. Flash 16-bit	16M	

Then the following window will appear - Press "ok" at this point.

Flash Do	wnload Setup				×
	Function Erase Full Chip Erase Sectors Do not Erase			Algorithm	
Program	ming Algorithm				_
Descr	iption	Device Type	Device Size	Address Range	
ADuC	702X Flash (v1.1)	On-chip Flash	62k	00080000H - 0008F7FFH	
			Start:	Size:	
	Add	Remove	ОК	Cancel <u>H</u> elp]

c. Press OK again, all the options should be properly configured to compile, assemble, link, download and debug using mIDAS-Link or the simulator.

2.4.1 Configuring the RDI Debugger driver on the ADuC7023

In the "Debug" panel, select Use ... and choose RDI Interface Driver if you have a mIDAS-Link emulator. Select the settings button and give the path to the JLinkRDI.dll - C:\Program Files\SEGGER\JLinkARM_V408l\JLinkRDI.dll

If your development system doesn't include a JTAG emulator, use the simulator for this exercise.



(2) Keil µVision3 Integrated Development Environment

Options for Target 'Target 1'	
Device Target Output Listing C Asm LA Locc C Use Simulator Limit Speed to Real-Time	ate LA Misc Debug Utilities
Coad Application at Startup Initialization File: Edit. Restore Debug Session Settings Breakpoints Vatchpoints & PA Watchpoints & PA Wency Display	Image: Construction at Startup Image: Run to main() Initialization File: Restore Debug Session Settings Image: Restore Debug Session Settings Image: Restore Debug Session Settings Image: Restore Debug Session Settings Image: Restore Debug Session Settings Image: Restore Debug Session Settings Image: Restore Debug Session Settings Image: Restore Debug Session Settings Image: Restore Debug Session Settings Image: Restore Debug Session Settings Image: Restore Debug Session Settings Image: Restore Debug Session Settings Image: Restore Debug Session Settings Image: Restore Debug Session Settings Image: Restore Debug Session Settings Image: Restore Debug Session Settings Image: Restore Debug Session Settings Image: Restore Debug Session Settings Image: Restore Debug Session Settings Image: Restore Debug Session Settings Image: Restore Debug Session Settings Image: Restore Debug Session Settings Image: Restore Debug Session Settings Image: Restore Debug Session Settings Image: Restore Debug Session Settings Image: Restore Debug Session Settings Image: Restore Debug Session Settings Image: Restore Debug Session Settings Image: Restore Debug Session Settings Image:
CPU DLL: Parameter: SARM.DLL cADuC70	Driver DLL: Parameter: SARM.DLL cADuC70
Dialog DLL: Parameter: DARMAD.DLL pADuC7026	Dialog DLL: Parameter: TARMAD.DLL pADuC7026
OK Ca	ncel Defaults Help

RDI Interface Driver Setup		×
Browse for RDI Driver DLL C:\Program Files\SEGGER\JLinkAR!	M_V408I\JLinkRDI.dll	
Browse for ToolConf File Debug Cache Options Cache Code Cache Memory	Configure <u>R</u> DI Driver	
	OK Cancel	<u>H</u> elp

a. Select the "configure RDI Driver" button. And in the "Flash" panel, clear the "Enable Flash programming "option. Everything else in this window should be disabled.





Enable flash programming Allows programming the flash. This is required to download a program into flash memory or to set software breakpoints in flash (flash breakpoints). Device Analog ADuC7020x62	
memory or to set software breakpoints in flash (flash breakpoints).	
Device Analog ADuC7020x62	
RAM 8 KB @ address 0x10000	
Flash 62 KB @ address 0x80000	
🔽 Flash is <u>m</u> irrored @ address 0x0	
Cache flash contents	
Allows caching of flash contents. This avoids reading data twice and speeds up the transfer between debugger and target.	
Verify flash contents	
Allows verifying of flash contents. This is useful to check if the program was downloaded to flash memory correctly.	
Allow flash download	
Allows program download to flash. Your debugger does not need to have a flash	
loader. This feature requires an additional license (FlashDL).	
🔽 Skip download on CRC match	

In the "JTAG" tab, ensure the JTAG speed is set to 600KHz.See the screenshot below.

🛃 SEGGER J-Link RDI V-	4.03g (beta) Configuration	? 🗙
General Init JTAG	Flash Breakpoints CPU Log	
JTAG speed		
C Auto selection		
C Adaptive <u>c</u> locking		
 € 600 ▼ kHz 		
- JTAG scan chain with	a multiple devices	
Position 0	<u>I</u> R len 0	
0 is closest to TD0.	Sum of IRLens of devices closer to TDD. IRLen of ARM chips is 4.	
	<u>V</u> erify JTAG confi	
	<u>⊻</u> enys rAd com	9
	OK Cancel Ap	ply

In the "Breakpoints" tab, ensure "Software Breakpoints" are enabled.





BI SEGGER J-Link RDI V4.03g (beta) Configuration	?	×
General Init JTAG Flash Breakpoints CPU Log		
Use software breakpoints		
Software breakpoints (as opposed to hardware breakpoints) are breakpoints which modify program memory. This allows setting an unlimited number of breakpoints if the program is located in RAM.		
Use flash breakpoints	7	
Allows setting an unlimited number of breakpoints if the program is located in RAM or flash, which is extremely valuable when debugging a program located in flash.		
This feature is available only if flash programming is enabled!		
Show info window during program		
OK Cancel	ply	

In the "CPU" window, ensure it is configured like the following screenshot:

SEGGER J-Link RDI V4.03g (beta) Configuration
General Init JTAG Flash Breakpoints CPU Log
Allow instruction set simulation
Allows the emulator to simulate individual instructions when single stepping instructions. This does not normally have any disadvantages and makes debugging much faster, especially when using flash breakpoints.
Endian
⊙ Little endian
C <u>B</u> ig endian
Beset strategy J-Link supports different reset strategies. This is necessary because there is no single way of resetting and halting an ARM core before it starts to execute instructions. Software, for Analog Devices ADuC7xxx MCUs ▼ Delay after reset 0 ms
The following sequence is executed: - The CPU is halted - A soft reset sequence is downloaded to RAM - A breakpoint at 0 is set - The soft reset sequence is executed This sequence performs a reset of CPU and peripherals and halts the CPU before executing instructions of the user program. It is recommended reset sequence for Analog Devices ADuC7xxx MCUs and works with these chips only.
OK Cancel Apply

b. In the "Utilities" panel, select Use Target Driver for Flash Programming", chose "RDI Interface Driver" and tick the option "Update Target before Debugging".



(2) Keil µVision3 Integrated Development Environment

Options for Target 'Target 1'	×
Device Target Output Listing C Asm LA Locate LA Misc Debug Utilities	
Configure Flash Menu Command	
Use Target Driver for Flash Programming	
RDI Interface Driver 💽 Settings 🔽 Update Target before Debugging	
Init File: Edit	
C Use External Tool for Flash Programming	
Command:	
Arguments:	
Final Run Independent	
OK Cancel Defaults Help	

Hit the "Settings" option and the following dialog box will appear - "uVision Flash Programmer" and press "Ok".

Select Flash Programmer	×
µVision Flash Programmer ▼	
,	
OK	

Then, a new dialog window will appear as below.

Dialog				×
Download Function C Erase Full Chip C Erase Sectors C Do not Erase	Verify		Algorithm Dx00010000 Size: (0x0800	
Description	Device Type	Device Size	Address Range	-
		2		-
Add	Remove	Start: OK	Size: Cancel Help	

Select "Add" for the next window. Select "ADuC702X Flash (v1.1) and press "Add"



Description	Device Type	Device Size	~
ADuC702X Flash (v1.1)	On-chip Flash	62k	
ADuC703X Flash 32KB (v1.4)	On-chip Flash	30k	-
ADuC703X Flash 64KB (v1.1)	On-chip Flash	62k	
ADuC703x Flash 96KB (v1.4)	On-chip Flash	94k	
ADuC706X Flash 32KB (v1.0)	On-chip Flash	30k	
ADuC7124 Flash (v1.0)	On-chip Flash	126k	
ADuC712x Flash (v1.2)	On-chip Flash	126k	
ADuC7229 Flash (v1.2)	On-chip Flash	126k	
AM29F160DB Flash	Ext. Flash 16-bit	2M	
AM29F160DT Flash	Ext. Flash 16-bit	2M	
AM29F320DB Flash	Ext. Flash 16-bit	4M	
AM29F320DB Dual Flash	Ext. Flash 32-bit	8M	
AM29F320DT Flash	Ext. Flash 16-bit	4M	
AM29F320DT Dual Flash	Ext. Flash 32-bit	8M	
AM29x033 Flash	Ext. Flash 8-bit	4M	
AM29x128 Flash	Ext. Flash 16-bit	16M	~

Then the following window will appear - Press "ok" at this point.

Download Function C Erase Full Chip C Erase Sectors C Do not Erase Programming Algorithm			Algorithm Dx00010000 Size: 0x0800
Description ADuC702X Flash (v1.1)	Device Type On-chip Flash	Device Size 62k	Address Range 00080000H - 0008F7FFH
,		-	Dx00080000 Size: Dx0000F800

c. Press OK again, all the options should be properly configured to compile, assemble, link, download and debug using mIDAS-Link or the simulator.

2.5 Adding Project files

a. All the files relative to the project will be in the folder C:\ADuC7XXXV0.2\ADuC7023\Code\Realview\Demo.

b. From Windows Explorer or any other way, copy the files C:\ADuC7XXXV0.2\ADuC7023\Code\Realview\Demo\Demo.c, irq_arm.c and ADuC702x.s into the following directory C:\Keil\ARM\Examples\ADIdemo.

To add the files to the project right click on the "Source Group" folder in the "Project Workspace" and select "Add Files to Group".



Project Workspace		× x
🖃 🔁 Target 1		
🖻 🖷 🤤 Sour	^-	
····· 📰 :		Options for Group 'Source Group 1'
		Open Lis <u>t</u> File
		Open Map File
		Open File
		<u>R</u> ebuild target
		Build target F7
		Tr <u>a</u> nslate File
	X	Stop build
		New Group
		Add Files to Group 'Source Group 1'
		Manage Components
		Remove Group 'Source Group 1' and it's Files
	~	Include Dependencies

Note that you can use under Project the option "Component, Environment, Books" to rename the target and add the file relative to your project.

Project Workspace	* X
🖃 🔁 Target 1	
🖻 📇 Source Group 1	
···· 🔝 irq_arm.c	
🔝 ADuC702x.s	
🗄 🗠 🔛 Demo.C	

Double click on the file name (Demo.c) in the Workspace window to open the source file. c.

Proje

Assembling/Compiling Code 2.6

To compile/link Demo.c click on the 🖄 (translate current file) icon in the toolbar. The file should compile correctly and the following will a. be seen in the status window. If there are errors in your source code these will appear in the status window. To identify the line of code which corresponds to the error double click on the error in the Output window and an arrow will appear highlighting the line of code in which the error appears.

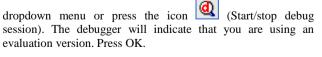
The ADuC7023.h file will automatically be included under Demo.c in the Project Workspace window.

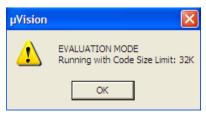
Before the code can be downloaded to the MicroConverter the entire project must be built. This is done by clicking on the (rebuild all b. target files) icon on the toolbar. It will also create a pot.elf file used by the debugger.

2	Build target 'Target 1'
1	compiling irg_arm.c
	assembling ADuC702x.s
	compiling Demo.C
	linking
	Program Size: Code=1516 RO-data=16416 RW-data=24 ZI-data=1760
	"GPIO.axf" - 0 Error(s), 0 Warning(s).

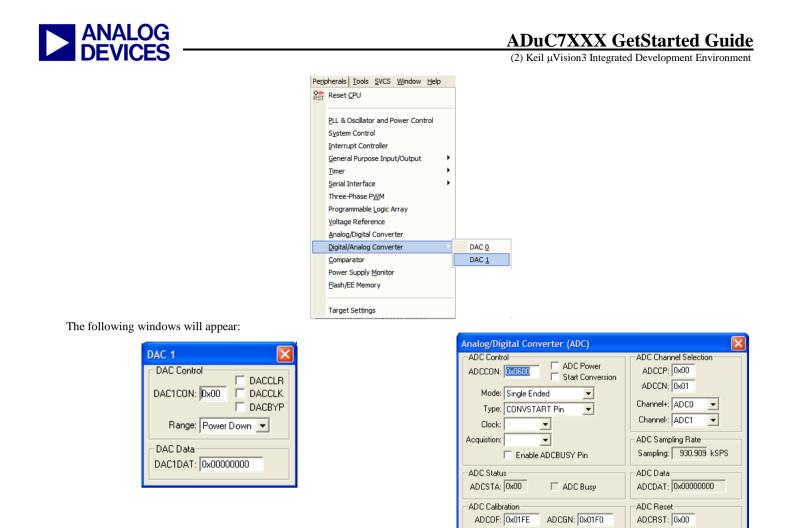
Downloading/Debugging Code 2.7

- Check on the evaluation board that \$1.3 and \$1.8 is in ON position for the DAC1 and POT. a.
- To start debugging, select Start/Stop Debug in the Debug b.





From the Peripheral pull down menu, select Digital/Analog Converter, DAC1 and select Analog/Digital Converter. c.



d. Close the disassembly window. Go to Demo.c file. Set a breakpoint on the instruction ADCCON = 0x6A3;. This is done by right clicking on the line of code and then selecting "insert/remove breakpoint" or by double clicking on the left of the instruction. Notice that the breakpoint is indicated by a large red dot to the left of the line.



```
19
                        on DAC1 output. This has the affect of controlling the
  20
                        brightness of the LED, D1.
Ensure "POT" and "DAC" switches of S1 are "On"
  21
                                                                         .<u>.</u>
*********/
       *****
  22
      #include <aduc7023.h>
  23
      unsigned int uiPLLTST = 0;
  24
      volatile unsigned int ucTest = 0;
  25
  26
  27
      void ADCpoweron(int);
  28
      int main(void)
  29 🗌 {
  30
           POWKEY1 = 0 \times 01;
                                              // Configure CPU Clock for 41.78MHz, CD=0
           POWCON = 0 \times 00;
  31
  32
           POWKEY2 = 0xF4;
   33
          DAC1CON = 0x13;
  34
                                              // Set DAC1 output to 0-AVDD range and turn DAC1 on
  35
          ADCpoweron (20000);
                                              // power on ADC
          ADCCP = 0x00;
  36
                                              // select ADC channel 0
          REFCON = 0 \times 01;
  37
                                              // internal 2.5V reference. 2.5V on Vref pin
  38
           while (1)
  39
  40
           {
                                             // software conv., single-ended, conv. enabled
// wait for end of conversion
  41
      ADCCON = 0 \times 6A3:
               while (!ADCSTA) {}
  42
               DAC1DAT = ADCDAT;
                                              // result format is identical for DAC and ADC
  43
  44
           3
  45
  46
  47
     void ADCpoweron(int time)
  48
  49 🖂 {
4
```

e. Press the run code button twice. The ADCDAT and DAC1DAT values should be identical.

f. Change the value of the potentiometer press run and view the value of ADCDAT changing and the intensity of the LED changing. Remove the breakpoint, press run. When moving the potentiometer the intensity of the LED changes.

g. To stop the code running press \bigotimes and to stop debugging press \bigotimes .





(3) IAR EMBEDDED WORKBENCH IDE

The IAR Embedded Workbench IDE (EWARM) integrates all the tools necessary to edit, assemble and debug code via JTAG

3.1 Installing the IAR Embedded Workbench

To install the IAR Embedded Workbench software, follow these instructions.

Post installation of the ADuC7XXX CD the IAR Embedded Workbench installation executable needs to be manually selected. This executable is found at C:\ADuC7XXXV0.2\Applications\IAR\EWARM-KS-WEB-

5407.zip. The IAR Embedded Workbench should be installed at C:\Program Files for this installation to function correctly. Post installation a license key is required from IAR in order to operate the evaluation software. Once IAR has been downloaded, double click on the file copy.bat located at C:\ADuC7XXXV0.2\Applications\IAR. This will copy across the latest ADI Generics information not yet available in the latest release of IAR Embedded Workbench.

Before the installation is complete, you will need to register on the IAR webpage. To register, go to the "downloads" page and select the "ARM KickStart version" for download – this will result in the registration page being displayed for you. Note, you do not need to download a new version of the IAR evaluation software for ARM7 parts. The version 5.407 on this CD is recommended as this version is compatible with the IAR example code provided on the CD.

3.1.1 Installing the IAR Embedded Workbench support for the ADuC7124/26, ADuC7029, ADuC7023 and ADuC7122

Once IAR has been downloaded, double click on the file copy.bat located at C:\ADuC7XXXV0.2\Applications\IAR. This will copy across ADI Generics information not included in this release of IAR Embedded Workbench.

3.2 Starting IAR Embedded Workbench

 From the "START" menu choose All Programs → IAR systems → IAR Embedded Workbench for ARM Kickstart → IAR Embedded Workbench. This loads the IAR Embedded Workbench IDE. The following window will appear, close it.

Embedde	d Workbench Startup	X	
	Create new project in current workspace		
	Add existing project to current workspace		
	Open existing workspace		
	Example workspaces		
Recent workspaces:			
🗌 Do no	Do not show this window at startup.		
	Close		

3.3 Starting a project in an existing workspace

an existing project directory.

and selec

UIKSDACC

🔏 IAR Embedded	l Workbench	n IDE	
File Edit View Pr	oject Tools	Window Help	
New	+.		
Open	۱.	File	CTRL+O
Close		Workspace	
Save Workspace		Header/Source F	File CTRL+SHIFT+H
Close Workspace	L		
Save	CTRL+S		
Save As			
Save All			
Page Setup			
Print	CTRL+P		
Recent Files	+		
Recent Workspace	es ▶		
Exit			

All the code examples given on this CD are part of this workspace:

5 - 52 - 632	<u>+</u> 12	5 🙎
Workspace		×
Debug		•
Files	8 2	0 ; ;
🗆 💽 Example - Debug		
🛏 🗄 cstartup.s		
🗕 🕀 🔝 main.c		
🖵 🖽 🧰 Output		

3. From the project menu select Create New Project as shown below and choose Empty Project in the Project Template list. Press OK.

REV 04



ile Edit View	Project RDI Tools \	Window Help
🗅 😅 🖬 🖪	Add Files	
Vorkspace	Add Group	
Debug	Import File List	
	Edit Configurations	
Files	Remove	
⊟ <u>ि</u> Exampl: —⊞ 📓 cstartı	Create New Project	
-⊞ B main.	Add Existing Project	
L-⊞ 🚞 Outpu	Options	ALT+F7
	Options	ALITI7
	Source Code Control	•
	Make	F7
	Compile	CTRL+F7
	Rebuild All	
	Clean	
	Batch build	F8
	Stop Build	CTRL+Break
	Debug	CTRL+D
	Make & Restart Debu	gger
Create New Project	1 1 18	

Tool chain: ARM			
Project templates:			
Empty project			
tr_asm			
in C++ in C			
Externally built executable			
Description:			
Creates an empty project.			
OK Cancel			

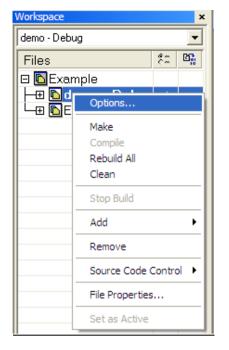
4. Name the new project "Demo" and save it under C:\ADuC7XXXV0.2\ADuC7122\Code\IARexamples\Demo. The new project will automatically appear in the workspace.

File Edit View Project Si	mulato	or Too
🗅 🚅 🖬 🕼 🎒 🐰	Ē	e -
Workspace		x
demo - Debug		-
Files	8 <u>~</u>	10 7
🗉 🔂 Example *		
📙 🕒 demo - Deb	×	
🛛 🖵 🗈 Example - De	×	
11		

5. Right click on "Demo" in the workspace and select add files. Add Timer.c and cstartup.s from the Timer folder under IARexamples. Note that you need to change "file of type" to "all file" to add cstartup.s

6. Right click on Demo in the workspace and select options.





The following window appears: change the processor variant to device and choose the device that you have in your development system. In the "category" list, select "linker".

Options for node "De Category: General Options C/C++ Compiler Assembler Custom Build Build Actions Linker Debugger Simulator	Target Output Library Configuration Library options MISRA C Processor variant
Simulator Angel IAR ROM-monitor J-Link Macraigor RDI Third-Party Driver	FPU None FPU Generate interwork code Processor mode C Am C Am C Little Big OK Cancel

3.4 Configuration for the mIDAS-Link

7. Select Override Default in the linker command file box and choose the file "ADuC7122_Flash_Standalone.icf" under: C:\ProgramFiles\IARSystems\EmbeddedWorkbench5.4 Kickstart\ARM\examples\AnalogDevices\SupportFiles.



Category: General Options	Factory Settings
C/C++ Compiler Assembler Output Converter Custom Build Build Actions Linker Debugger Simulator Angel GDB Server IAR ROM-monitor J-Link/J-Trace LMI FTDI Macraigor RDI	Config Library Input Output List #define Diagnostics Chr. ▲ Linker configuration file Image: Configuration file symbol definitions: (one per line) Image: Configuration file symbol definitions: (one per line) Image: Configuration file Image: Configuration fi
ST-Link Third-Party Driver	

8. In the Debugger options, select the Setup panel and choose J-LINK/J-TRACE as driver. Also check the override default tick box under the device description file heading and set the path as follows

\$TOOLKIT_DIR\$\CONFIG\debugger\AnalogDevices\ioaduc 7122.ddf. Configure menus as follows.

Category: Factory Settings General Options C/C++ Compiler Assembler Output Converter Output Converter Download Build Actions Download Linker Imain Debugger Setup macros Simulator Setup macros Angel Use macro file(s) GDB Server IAR ROM-monitor J-Link/J-Trace Imain Devige description file Imain St-Link Third-Party Driver
OK Cancel

(3)	IAR

Options for node "De	mo"
Category: General Options C/C++ Compiler Assembiler Output Converter Custom Build Build Actions Linker Debugger Simulator Angel GDB Server IAR ROM-monitor J-Link/J-Trace LMI FTDI Macraigor RDI ST-Link Third-Party Driver	Setup Download Extra Options Images Plugins Attach to program Verify download Suppress download Use flash loader(s) Qvenide default board file \$TOOLKIT_DIRS\config\flashloader\AnalogDevices} Edt
	OK Cancel

9. In the J-LINK/J-TRACE panel, set the reset type as software and configure the JTAG speed to 750 kHz. Press OK, the project is now configured to debug via the mIDAS-Link.

Options for node "De	emo"	X
Category: General Options C/C++ Compiler Assembler Output Converter Custom Build Build Actions Linker Debugger Simulator Angel GDB Server IAR ROM-monitor 3-Link/J-Trace LMI FTDI Macraigor RDI ST-Link Third-Party Driver	Setup Connection Breakpoints Reset 0 JTAG/SWD speed Catch exceptions Atg Beset B0 ipital 32 kHz © Exed 750 kHz © Adaptive Prefetch	Factory Settings
	OK	Cancel

10. To start debugging press the icon \blacktriangleright or select Debug in the Project dropdown menu.



(4) I2C Downloader

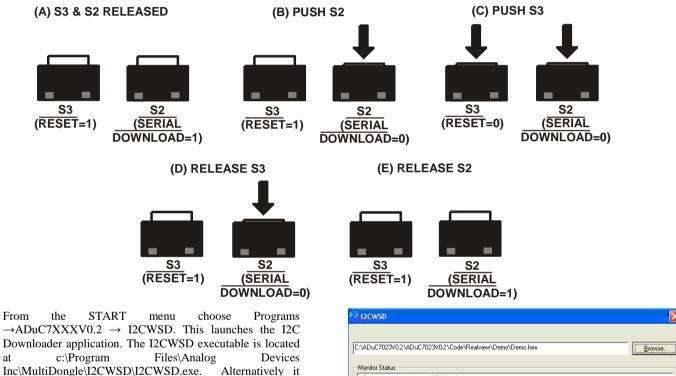
(4) THE WINDOWS I2C DOWNLOADER

The Windows I2C Downloader for ARM based part (I2CWSD) is a windows software program that allows a user to download Intel Extended Hex files as created by assembler/compilers to the MicroConverter via the I2C port. The Intel Extended Hex file is downloaded into the on-chip Flash/EE program memory via a USB-I2C dongle (MCV COMBI DONGLE) in the case of this evaluation kit.

For the ADuC7023 evaluation board, no dongle is required to program the part via USB. But, the ADuC7023 specific version of I2CWSD must be used. This version is located in the following folder: C:\ADuC7XXXV0.2\Documentation\ADuC7023\EvalDoc\ADuC7023 Eval board Programming

4.1 Opening the I2CWSD

- 1. Power up the evaluation board using the USB interface.
- 2. Connect the MCV COMBI dongle to your PC via the USB interface. Note, for the ADuC7023 evaluation board, there is a specific I2CWSD application that does not require the MCV COMBI dongle only the USB cable.
- 3. Connect the 4-pin I2C header of the dongle to the temporary I2C header of the ADuC7023 evaluation board.
- 4. Ensure that Flash address 0x80014 is erased (equal to 0xFFFFFFF).
- 5. The user should put the MicroConverter into I2C download mode using the following sequence:



C:\ADuC7023V02\ADuC7023V02\Code\Realview\Demo\Demo\new	Browse.
Monitor Status	-
Please connect board to USB-Dongle For download Press Reset on Hardware before click >Start<	
Configure Start Flash Run from 0x0	Exit
Checksum from imported HEX File MassErase on Program on Verify Communication: none Mode: offline 100 Kauld	CPLI- ADUC 702

4.2 Downloading using I2CWSD

Devices Inc \rightarrow I2CWSD.

7. Select the file at C:\ADuC7XXXV0.2\ADuC7023\Code\Realview\Demo\demo.hex.

can be launched from "start \rightarrow all programs \rightarrow Analog

6.

at



(5) Elves

- 8. Press the Configure button and in the Flash panel select "Mass Erase" and "Program". Press OK.
- 9. Press the Start button. The I2CWSD automatically sends the reset command to the MicroConverter. If the MicroConverter is in I2C download mode and the comms between the PC and the evaluation board are setup correctly, then the text "Connected to ADuC7023.." should appear.
- 10. Press the "Flash" button to start downloading the hex file. A progress bar while appear when the file is downloading. Once the file has been successfully downloaded monitor status box will be updated with "Download Complete Click Run to run program".

C:\ADuC7XXXV0.1\ADuC7023\Code\Realview\Demo\Demo.hex	Browse
Monitor Status	
Building up connection	
Requesting CPU ID	
Waiting up to 20 seconds for CPU ID to be returned	
CPU ID Response Received	
CPU ID Response Received Connected to ADuC7023i -62 A5Y.	
CPU ID Response Received Connected to ADuC7023i -62 A5Y.	
	Exit



(5) Elves

(5) ELVES

5.1 Using Elves.exe with IAR Embedded Workbench

 In IAR Embedded Workbench IDE, using the previous example, select Elves in the Tools pull down menu. On first use a text file will be launched explaining how to setup this tool. Click on the setup button, check the path of the browser and select details. Exit this window.

🔏 IAR Embedded Workbench IDE			
File Edit View Project	Tools	Window Help	
🗅 🛩 🖬 🕼 🕼	Opt	tions	
Workspace	Cor	nfigure Tools	
Blink - Debug		name Extensions	
Files		MWSD	
	Elve	es	
IAR Code Exam	PLA	Tool	

2. In the Select Library, a list of library should be visible. Choose LibDac702x.h.

😵 Elves 🛛 🔀
Select Library Linist Add C:\ADuC7xxxV0.1\ADuC702x\code\IAR Code Examples\LibIAR\ADC\LibAdc702x.h
Choose <u>F</u> unction
int AdcGo(int iChanP, int iChanN, int iStart)
Source Code
C Function prototype: int AdcGo(int iChanP, int iChanN, int iStart); ChanP:(ADC_0ADC_1ADC_2ADC_3ADC_4ADC_5ADC_6ADC_7ADC_8ADC_9ADC_10ADC_11, iChanN:(ADC_0ADC_1ADC_2ADC_3ADC_4ADC_5ADC_6ADC_7ADC_8ADC_9ADC_10ADC_11, iStart(ADC_0FFADC_T2S_ADC_T3S_ADC_SWS_ADC_CCS_ADC_PLAS_ADC_IRQS); Description of Function: Start ADC conversion. User interface: S et iChanP to set which signal will be connected to the positive input of the ADC; S of ADC_0ADC15 for external channels 0-15 (if available). S of ADC_TEMP for internal temperature (iChanN must be 17 (ADC_GND)).
Choose Parameters int iChanP iChanN iStart AdcGo(ADC_0 ▼ ADC_0 ▼ ADC_0FF ▼)opy
Help F1= <u>H</u> elp Setup

- 3. In the Choose function, select int DACRng(). You can now see a description of the function and the code. Fill the parameters, channel 1 and AVDD_RNG. Press the copy button. In your C code, replace DAC1 configuration by the function (right click and paste).
- Repeat this with int DacOut() function, channel 1 and value = ADCDAT. Replace the line DAC1DAT = ADCDAT with DacOut function.
 In the project workspace add the file libDAC702x.79 that is under C:\ADuC7XXXV0.1\ADuC702x\code\IAR Code Examples\LibIAR\DAC. In the main code include the two files:
- #include" C:\ADuC7XXXV0.2\ADuC702x\code\IAR Code Examples\LibIAR\DAC \libDac702x.h"
- 6. Other functions are also available under :\ADuC7XXXV0.2\ADuC702x\code\IAR Code Examples\LibIAR.

5.2 Using Elves.exe with uVision3

The Elves can also be used in a similar way under Keil uVision3. Select Elves under the Tools pull down menu.

🕎 DAC - μVision3	
<u>File Edit View Project Debug Flash Peripherals</u>	Tools SVCS Window Help
🆀 😅 🖬 🍠 👗 🖻 🛍 🛛 으 오 🛛 建 🕸	Setup PC-Lint
😵 🍱 🎬 👗 🙀 🎊 DAC Debugger	Lint All C-Source Files
Project Workspace v x	Customize Tools Menu
DACsine.c	Elfes

In pot.c add the following files:

#include" C:\ADuC7XXXV0.2\ADuC702x\code\RealView Code examples\LibKeil\DAC \libDac702x.h"

And in the option for target, under linker, add the name and path to the library:



Options for T	Target 'Target 1'		×
Device Targe	et Output Listing CC Assembler Linker Debug Utilities		
	□ Enable Garbage Collection Text Start: □ Do not use Standard System Startup Files Data Start: □ Do not use Standard System Libraries BSS Start: □ Use Math Libraries Ess Start:		
File: Include Libraries	Dac702x	Edit	
Misc controls			
Linker control string	-L C:\ADuC702x\code\KeilCodeExamples\LibKeil\DAC *.o -IDac702x	 	
	OK Cancel Defaults	Help	



(6) PLA tool

(6) PLA TOOL

The PLA tool is a graphical tool allowing configuring the PLA easily.

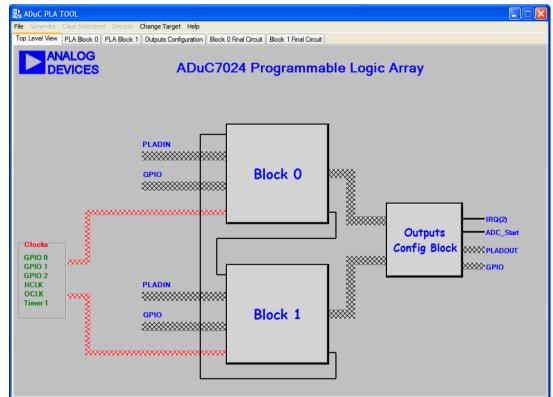
The following example is part of the code example, PLA, under Code\KeilCodeExamples.

The code generated corresponds to the file PLAinit.c of ADCPLA code example. ADC conversion will occur if P1.2 is high and PLADIN bit 2 is set. Note that the ADC must be configured to start conversion on a PLA start convert signal. More comprehensive examples are in the PLAtool help.

6.1 Opening the PLATool

- Launch PLAtool.exe from the START menu under All Programs → ADuC7XXX → PLAtool or from c:\ADuC7XXXV0.2\Applications\PLAandElves\PLATool_V2. 5_Setup.msi and select a part number.
- 2. Select ADuC7020 and press OK. The window below appears.

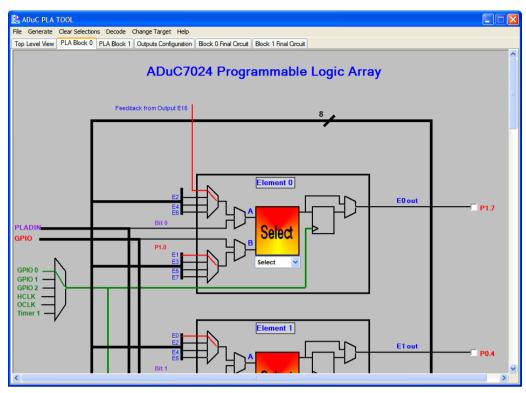




6.2 Configuring the gates and output

3. Select block 0.





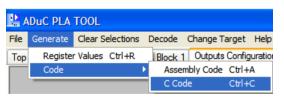
4. Select AND gate for element 2, with multiplexer A on Bit 2 bus i.e. MMR PLADIN and multiplexer B on P1.2. Choose to bypass the flip flop. Select the "Output" panel, scroll down and choose to start an ADC conversion using element 2 output.

😫 ADUC PLA TOOL	
File Generate Clear Selections Decode Change Target Help	
Top Level View PLA Block 0 PLA Block 1 Outputs Configuration Block 0 Final Circuit Block 1 Final Circuit	
ADC st_conv	
ADC SL_CONV	
Block 0	
Elem 0	
Elem 2	
Elem 3	
Elem 4	
Elem 5 \	
Elem 6	
Elem 7 \	
	- ADC start
Block 1	
Elem 8	
Elem 9	

The PLA peripheral is now configured as in example ADCPLA.

6.3 Generating C code

5. Select Generate \rightarrow Code \rightarrow C Code.



The code generated corresponds to the file PLAinit.c under C:\ADuC7XXXV0.1\ADuC702x\code\RealView Code examples\PLA.

ADuC PLA	Tool - CodeWindow	
.,	PLA Tool Code Window	
V	Code Generated By the ADuC 702X PLA Tool	_
//FileTy	pe: C PLA Configuration File	
//Sourc	e: C Source Code	
//Date:	20/07/2004 14:43:05	
//====		
#includ	e "ADuC7024.h"	
void {	plalnitialize()	
	// Configure Port Pins for PLA mode	
	GP1CON = 0x0300;	
	Close	



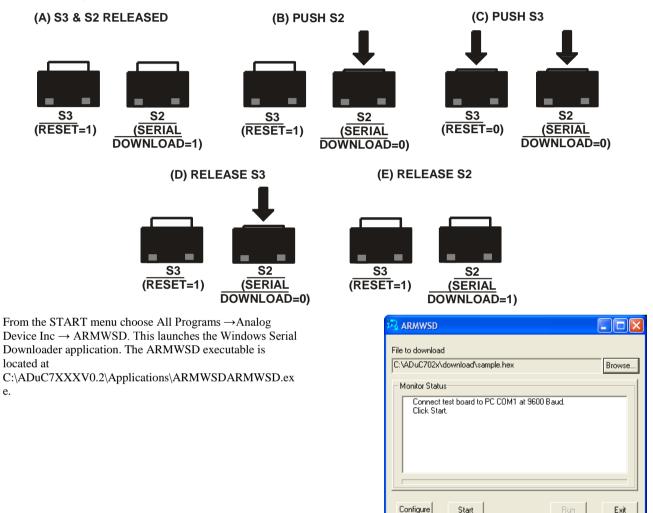
(7) Windows Serial Downloader

(7) THE WINDOWS SERIAL DOWNLOADER

The Windows Serial Downloader for ARM based part (ARMWSD) is a windows software program that allows a user to serially download Intel Extended Hex files as created by assembler/compilers to the MicroConverter 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 (COM1 to COM31).

7.1 **Opening the ARMWSD**

- 11. Power up the evaluation board using the 9V power supply. Connect the evaluation board header J1 to your PC's COM1 serial port using the RS-232 dongle cable provided. The PC serial COM port may be changed from COM1 via the WSD 'configuration' option...see section 4.3 below.
- 12. The user should put the MicroConverter into serial download mode using the following sequence:



7.2 Downloading using ARMWSD

- Select the file at C C:\ADuC7XXXV0.2\ADuC702x\code\IAR Code Examples\Misc\blink.hex. 14.
- 15. Press the Configure button and in the Parts panel select the part you are using. Press OK.
- 16. Press the Start button. The ARMWSD automatically sends the reset command to the MicroConverter. If the MicroConverter is in serial download mode and the comms between the PC and the evaluation board are setup correctly then the ARMWSD should start downloading the hex file and display a progress bar while the file is downloading. Once the file has been successfully downloaded monitor status box will be updated with "Download Complete Click Run to run program".

💀 ARMWSD
File to download
C:\ADuC7XXXV0.1\ADuC702x\code\\AR Code Examples\Misc\Bli Browse
Monitor Status
Connect test board to PC COM1 at 9600 Baud. Click Start. Ready © 9600 baud n,8,1. Press Download and pulse Reset on hardware. ADuC7026 -62 H3T received. Erasing 2 pages of 512 bytes each. Downloading 729 bytes
Configure Start Run Exit

13.

e.



(7) Windows Serial Downloader

7.3 Running the downloaded file

Running using the ARMWSD

17. Click the run button. The monitor status box will be updated with the message "Running. Click Start for new Download". The program start running from 80000h (start of the Flash/EE).

Manual Run option

18. Press RESET on the evaluation board with the SERIAL DOWNLOAD switch released. The program starts running automatically after reset as can be seen by the flashing LED.

7.4 Additional options

The ADuC702x family incorporates a serial download protocol that allows various options (see ARMWSD_protocol technote). These options can be selected in the Configuration window as shown below.

Configure
Parts Help Comms Flash
Browser - full path and extension Browse c:\Program Files\Internet Explorer\vexplore.exe
Help Path Browse
OK Cancel
Configure 2
Parts Help Comms Rash
Serial Port Baudrate COM1 Geodetic Baudrate
OK Cancel
Configure 🛛
Parts Help Comms Flash
Rashing options ☐ Mass erase ☑ Program ☑ Venfy ☐ Protect
OK Cancel



(7) Windows Serial Downloader

If the protect option is selected, another panel is available. Select the pages to protect, enter the key and press OK. Note that using the ARMWSD, only a mass erase will be able to erase the protection.

Configure	×
Parts Help Comms Flash Protection	
Highlight ranges to protect	
Pages 096-099 (0x0c000-0x0c7ff) 🔨	
Pages 100-103 (0x0c800-0x0cfff)	
Pages 104-107 (0x0d000-0x0d7ff) Pages 108-111 (0x0d800-0x0dfff)	
Pages 112-115 (0x0e000-0x0e7ff)	
Pages 116-119 (0x0e800-0x0efff)	
Pages 120-123 (0x0f000-0x0f7ff) ⊻	
Key0	
All None 0x ABCDEF01	
OK Cancel	

NOTE: use of the PC COM port

Only one application may use the PC serial port at any time. Close the ARMWSD before using any other application that uses the PC COM serial port



(8) Windows Analog Software Program (WASP)

(8) WINDOWS ANALOG SOFTWARE PROGRAM (WASP)

The Windows Analog Software Program (WASP) is a windows software program for all ADuC702x MicroConverter products that allows analysis of their analog performance.

This application is currently in beta sampling. ADI is not liable for the use of this software.

8.1 Setup

The WASP uses a mIDASLink emulator to configure the MicroConverter and to acquire results.

Usage of the mIDASLink emulator requires that the USB driver for this emulator be installed before use. Refer to Section <u>Tools installation options</u> in the first chapter for information about installing this driver if necessary.

Before launching the WASP, you should ensure that

- The mIDASLink emulator is connected to the PC via the USB cable.
- The USB driver for the mIDASLink emulator has been provided to Windows.
- The 20 way IDC mIDASLink emulator JTAG cable is connected to the IDC JTAG connector on your target board.
- The target board is powered up.

8.2 Usage

The WASP consists of three separate pages: Connection, Configuration, and Results.

😹 WASP7				
File Pages Help				
🛋 🗅 🜌				
Connection	🕸 Connection			
Noise Analysis	ROI Select the Remote Debug Interface DLL (RDI) to use			
	Program Files Analog Revices (Sec. MASS201.IntSch. Ed			
	Status			
Ready Ready	Running			
WASP Connection Page				

The Connection Page contains information about any previous connection attempt and also the possibility to reconfigure the mIDASLink if required.

The information shown in the Status box on this page may be useful if the WASP failed to connect to the ADuC702x for any reason.

There is also the possibility to re-configure the mIDASLink via the *Configure* button. However, re-configuring will not normally be necessary as the WASP will use a suitable default set-up initially.



(8) Windows Analog Software Program (WASP)

🗱 WASP7				
File Pages Help				
🌃 🗅 🜌				
Connection	Configuration Select perioheral to modify Peripheral ADC ADC ADC ADC ADCCN ADC	ADC O Parameter ADC Mode Positive Channel Negative Channel Clock Speed Acquisition Time ADCauser prin Conversion Input Int. Ref. Chalpel External Reference No. of Samples Click or hover over the links	Setting Single-ended mode ACC0 AGND (ADC2) & clocks disabled Single Software Enabled Powered up 1.00 V 4000 to modify the state of each	I setting
Ready		Running		10

WASP Configuration Page

The **Configuration Page** allows the ADC to be configured according to the user's preference. The set-up described here will be used in any subsequent acquisition of results.

ADC Configuration

You can modify the current setting of any field by either clicking or hovering over an underlined link on the page.

Parameter	Setting		
ADC Mode	Single-ended mode	-	
Positive Channel	ADC0		le-ended mode
Negative Channel	AGND		erential mode
Clock Speed	fADC/2	Psu	edo differential mode
Acquisition Time	8 clocks		
ADC _{BUSY} pin	disabled		1
Conversion Input	Single Software		
Int. Ref. Output	Enabled		
Int. Ref. Enabled	Powered up		
External Reference	<u>1.00 V</u>		
No. of Samples	4000		

Clicking on an underlined link.

Clicking on a link brings up a menu from which the desired option can be selected.

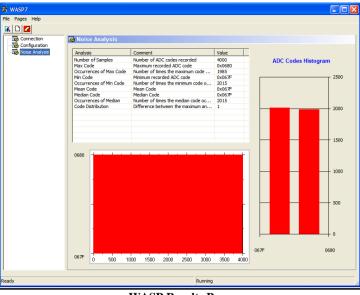
ADC	Configuration	n			
Parameter	Setting				
ADC Mode	Single-en	ded mode			
Positive Channel	ADC0				
Negative Channel	AGND	r	Conversion Mode		
Clock Speed	fADC/2		Conversion mode		
Acquisition Time	8 clocks				
ADC _{BUSY} pin	ADC _{BUSY} pin disabled		Single-ended mode Differential mode Psuedo differential mode		
Conversion Input	Conversion Input Single So				
Int. Ref. Output	Int. Ref. Output Enabled				
Int. Ref. Enabled	Powered				
External Reference	<u>1.00 V</u>	Single Ended Mode			
No. of Samples	<u>4000</u>				
Click or hover over the lini	s to modify t	ended mode. I	accepts an analog input range of 0 to VREF when operating in single- n single-ended modes, the input range is 0 V to VREF.		

Hovering over an underlined link

Alternatively, hovering over an underlined link will bring up a descriptive balloon from which the desired option can be selected.



(8) Windows Analog Software Program (WASP)



WASP Results Page

The Results Page shows the analysis of results from the last acquisition. The results analysis consists of the following

- A strip chart showing all results
- A histogram showing the relative frequency of acquired codes.
- Various statistics calculated from the acquired results.

Several toolbar buttons are also provided which perform some useful functions

Acquire Results

Connects to the MicroConverter and acquires results using the current configuration.

Reset Configuration

Reset the currently configured ADC configuration to a known default configuration.



1

Clear Results

Discards all currently acquired results.

Typical usage of the WASP program would involve the following

- Configure the ADC, using the **Configuration** page.
- Click the Acquire Results toolbar button.
- View results and analysis on the **Results** page.



(9) LIN Windows Serial Downloader

(9) LIN WINDOWS SERIAL DOWNLOADER

The applications LINBWSD and LINHWSD are a windows software program for the ADuC703x MicroConverter products that allows serial download of an Intel hex file to flash memory. The LINWSD is automatically extracted and installed by the ADuC7XXX CD to C:\Program Files\Analog Devices Inc\MultiDongle\LINBWSD and C:\Program Files\Analog Devices Inc\MultiDongle\LINHWSD. Further information can be found at C:\ADuC7xxxV0.2\ADuC7034_36\documentation\application notes\ADuC7032 LINWSD Manual.pdf



(10) Installed Documentation and Code Directory

(10) INSTALLED DOCUMENTATION AND CODE DIRECTORY

Installed Documentation Directory

Installing the MicroConverter[®] Development System CD installs documentation for the ADuC702x, ADuC706X, ADuC7034, ADuC7036, ADuC7039 and ADuC7122, ADuC7124/6 products at C:\ADuC7XXXV0.2\ documentation.

Mouser Electronics

Authorized Distributor

Click to View Pricing, Inventory, Delivery & Lifecycle Information:

Analog Devices Inc.: <u>EVAL-ADUC7024QSZ</u> EVAL-ADUC7029QSZ