

C Compiler • Real-Time OS • Simulator • Training • Evaluation Boards

Using the Keil Monitor-51 / μVision2 Debugger Application Note 151 with the Analog Devices ADuC812 Evaluation Board

May 17, 2000, Munich, Germany

by Keil Support, Keil Elektronik GmbH support.intl@keil.com ++49 89 456040-0

This Application Note describes the steps that are required to install and use the Keil Monitor-51 together with the Analog Devices ADuC812 Evaluation Board, Rev. B01

Hardware Modification – Von-Neumann wired code/xdata Memory

A requirement for the Keil Monitor-51 is von-Neumann wired **code** and **xdata** memory. The Analog Devices evaluation board comes without this hardware feature. Therefore you need to modify the hardware to match this requirement.

Von-Neumann means that you can read a physically the same memory bytes from code and xdata space. This is necessary to download software into **code** space since the 8051 does not provide CPU instructions to write into code memory. Typically a AND gate is used to combine the RD/ and PSEN/ signals of the CPU and generate a RD/ signal for the RAM device.



The figure below shows you the

signals on the board that you need to cut and wire to the AND gate.



Figure: Modifications and Signals on the Analog Devices ADuC812 Circuit Board

Download the Keil Monitor-51 to the on-chip code Flash ROM

This application note comes with a pre-configured Monitor that can be directly downloaded to the ADuC812 Evaluation Board using the AD Download utility. This download utility is located on the Analog Devices QuickStart Development System CD ROM in the folder DOWNLOAD.

The pre-configured Monitor is located in **ADUC812MONITORMON51.HEX**. This monitor was generated with the batch file INSTALL.BAT with the following parameters:

INSTALL 8 7F

This creates a Monitor program that uses the Timer 2 as baudrate generator with auto-adjust baudrate. The Monitor code area starts at default address 0 and the monitor data area starts at xdata address 0x7F00.

NOTE

Due to a chip problem in the ADuC812 silicon it is required to use the monitor version that comes with this application note. The standard Keil Monitor-51 included with the Keil 8051 toolchain will not work on the ADuC812 device.

The following steps are required to download the Keil Monitor-51 on the ADuC812 board:

- close jumper LK3 on the board
- insert jumper LK4 to position A (pin 1)
- insert jumper LK6 to position A (pin 1)
- connect the board via a 9-PIN serial cable to the COM1 interface of your PC
- connect power supply to the board (LED PWR is on)
- press the RESET button on the board
- open a MSDOS command prompt window on your PC
- enter G:\DOWNLOAD\DOWNLOAD \ADUC812\MONITOR\MON51.HEX
 G: is the path to your CD ROM drive
 \ADUC812\ is the path to the application note folder

This should give the following output:

Now remove jumper LK3 and press the RESET button to start the Keil Monitor-51.

Memory Mapping on the ADuC812 Board

After this modifications your Analog Devices ADuC812 board has the following memory layout.



This gives you for your application program 8KB xdata RAM from 0x0000 - 0x1FFF and about 24KB program code space from 0x2000 - 0x7EFF. However you may use portions of the von-Neumann RAM also as data space.

Using µVision2

To provide a quick start with the μ Vision2 IDE this application note comes with a small example program that is located in the **EXAMPLE** folder. This example is called **BLINKY** and flashes the P3.4 LED on the board. It covers also an interrupt service routine and contains a configured version of the **STARTUP.A51** file.

After installation of the μ Vision2 tool chain you can start the IDE and load the project file with Project – Open Project: **\ADUC812\EXAMPLES\BLINKY.UV2**. This gives you the following screen output.



µVision2 Screen after loading the BLINKY project

Then you can start using the Keil 8051 tool chain. The following shows you some of the possibilities.

Build	×
🗇 🏝 🕮 📥 🖍 Monitor	-
Monitor	
Flash	

To create and test the application with Monitor-51, you need to select the **Monitor** as target. Our example program is correctly configured for the ADuC812 board. For detailed information about the tool setup for the ADuC812 board refer to "Create Own μ Vision2 Projects" on page 7. The Flash target is a configuration for loading the application program into the Flash ROM of the ADuC812 device.



200

Compiling and Linking: When you are ready to compile and link the application, use the **Build Target** command from the Project menu or the toolbar. μ Vision2 begins to compile and link the source files in BLINKY and displays a message in the **Output Window – Build** when the build is finished.

Setup Monitor Communication: Open the **Project** – **Options for Target - Debug** dialog and select **Use: Keil Monitor-51**. Then use the **Settings** button to configure the serial communication with the ADuC812 board. μ Vision2 presents you with the following dialog:

Options for Target 'Monitor'	? ×
Target Output Listing C51 A51 L51 Locate L51 Miss C Use Simulator	c Debug ● <u>U</u> se: Keil Monitor-51 Driver ▼ Settings
Load Application at Startup Go till main() Initialization File:	Load Application at Startup Go till main() Initialization File:
Restore Debug Sessi Comm Port Settings Breakpoints Baudrate: Watchpoints & Pr Stop Program Execution with CPU DLL: Pare S8051.DLL OK	Cache Options Cache DATA (SFR) Cache IDATA Cache ZDATA Cache ZDATA Cache CODE Cancel
Dialog DLL: Parameter: DP51.DLL -p52 OK Cancel Defaults	

Since the Monitor-51 is configured for automatic baudrate detection you may use any baudrates up to 57600 bps. A good choice is a Baudrate of 38400.

NOTES

If you have problems starting the debug session with the Monitor check the following:

- [?] Jumper LK3 on the ADuC812 board must be open.
- ? The board must be connected via a serial link to the PC and the power supply must be ON.
- ? Reset the board again and use the **Try Again** button in the μ Vision2 debugger.
- Then you may start the debug session. The μVision2 debugger connects to the ADuC812 board, downloads the application program. In case of communication problems a dialog box opens that displays further options and instructions.
- Review the program code with View Disassembly Window. If the disassembly does not start at address C:0x2000 enter =C:0x2000 in the Output Command window of μ Vision2. The μ Vision2 screen should look like shown below:you may start the debug session. This should give the following screen output:

🖳 blinky - μVision2			
Eile Edit View Project Debug Per	ripherals Tools SVCS Window Help		
Montor Surve Code Surve Code Strinky.C Startus.S1 Abstracttd	C\aduc812\example\Abstract.bd The E Disassembly 91: 92: 92: STARTUP1: 93: 94: 94: IF IDATALEN <> 0 When Cinoz000 Cinoz000 025544 Cinoz005 08 Monit Cinoz006 Cinoz006 00 Mor Cinoz006 Cinoz006 00 Mor Cinoz006 Cinoz006 NOP Cinoz006 NOP Cinoz006 NOP Cinoz006 NOP Cinoz006 NOP Cinoz008 PZ06 Mor Cinoz000 Cinoz001 2425 Cinoz011 E531 Cinoz012 PB Cinoz014 Disasse	Image: Image	
Sifemented to Maniton E1 3	V2-0		
S-Onnected to Monitor-51 Load "C: \oduc812\\example S-0x2000 ASM ASSIGN BreakDisable	vs.∪ lev-Blinky" 	Address:	emory #2 \ Memory #3 \ Memory #4 /
Ready			

You may now enter **g**, **main** in the **Output** – **Command** window to execute the C compiler startup code. Then close the **Disassembly Window** and open the source file **BLINKY.C**. This can be done with the μ Vision2 command **Debug** – **Show next statement**. μ Vision2 presents the following screeen:

∰blinky - μVision2	
File Edit View Project Debug Peripherals Tools SVCS Window Help	
🖄 🚓 🗖 📶 X Ba 🙉 🗇 🖓 🖶 🗮 A 🖄 Shiki 📴 statmach inc	
Begister Value	
B Regs	
-10 0x00 The BLINKY P Note that an a	
Analog Devic there is no ope	rating system to return to). It
12 0x00 must loop and e	xecute forever.
rd 0.00 It contains	*/
15 000 /* setup the	timer 0 interrupt */
m6 0x00 When you run m10 = (unsign	ed char) FERIOD; /- set timer period -/
17 0x00 This example TMOD = TMOD	0x02; /* select mode 2 */
B-Sys and how you TRO = 1;	/* start timer 0 */
a 0x00 Monitor-51. ETO = 1;	/* enable timer 0 interrupt */
b 0x00 BA = 1;	/* global interrupt enable */
sp UX35	
DC & D-240a	812 Test Program\n\n"):
Flash #endif	
while (1) (
if (_testb:	t_(display)) {
#ifdef PRINT fir	e (D:0x11) = struct clock { hour=0x00, min=0x00, sec=0x00, msec=0x0000 }
if (time.	msec == 0) {
printf	("\rElapsed time: %02bd:%02bd:%02bd", time.hour, time.min, time.sec
#endit	
E ADStract.ot E Blinky.c	
Connected to Monitor-51 V3.0	Address
Load "C:\\aduc812\\example\\Blinky"	
5=C:Ux2000	
3, min	
ASM ASSIGN BreakDisable BreakEnable BreakKill BreakLis	BreakSet COVERAGE 🔽
I Build Command / Find in Files /	Memory #2 \ Memory #4 /
Ready	NUM R/W

The Monitor-51 supports common debugging features. You may single step through code, set breakpoints and run your application. Variables can be viewed by placing the mouse cursor on the variable or within the watch window. For detailed information about using the μVision2 debugger refer to the "μVision2 Getting Started with the 8051" User's Guide "Chapter 5. Testing Programs" (Keil\C51\HLP\GS51.PDF) that is included in the Keil 8051 tool chain.
 The Keil Monitor-51 allows you to share the serial interface that is used for Monitor communication with user I/O. The Serial Window #1 shows the *printf* and *putchar* output. Disable the option Monitor Driver Settings – Serial Interrupt to enter characters that are sent to the user program.

NOTES

It is important that you skip the initialization of the serial interface in the user application, since the Monitor performs the UART setup. You may use conditional compilation as shown in our program examples. Also it is impossible to single step through **putchar** or **getkey** I/O functions.

- If the option Monitor Driver Settings Serial Interrupt is enabled, you may stop program execution with Halt command from the Debug menu or the toolbar or type ESC in Output Window Command page. The Monitor uses the serial interrupt to halt the user program. If the Monitor cannot stop your program (because the user application has disabled interrupts or so) a dialog box opens that displays further options and instructions.
- RST

The μ Vision2 **Reset** command sets the program counter to 0. However it should be noted that peripherals and SFRs of the ADuC812 device are not set into reset state. Therefore this command it is not identical with a hardware reset of the CPU.

Create Own µVision2 Projects

To create a new project file select from the μ Vision2 menu **Project** – **New Project...**. This opens a standard Windows dialog that asks you for the new project file name.

We suggest that you use a separate folder for each project. You can simply use the icon **Create New Folder** in this dialog to get a new empty folder. Then select this folder and enter the file name for the new project, i.e. **Project1**. μ Vision2 creates a new project file with the name **PROJECT1.UV2** which contains a default target and file group name. You can see these names in the **Project Window – Files**.

Now use from the menu **Project** – **Select Device for Target** and select the Analog Devices ADuC812 CPU for your project. The **Select Device** dialog box shows the μ Vision2 device database. This selection sets necessary tool options for the ADuC812 device and simplifies in this way the tool configuration.

You may create a new source file with the menu option **File** – **New**. This opens an empty editor window where you can enter your source code. μ Vision2 enables the C color syntax highlighting when you save your file with the dialog **File** – **Save As...** under a filename with the extension *.C.



Once you have created your source file you can add this file to your project. μ Vision2 offers several ways to add source files to a project. For example, you can select the file group in the **Project Window** – **Files** page and click with the right mouse key to open a local menu. The option **Add Files** opens the standard files dialog. Select the file **MAIN.C** you have just created.

When you are using the Monitor-51 on the ADuC812 board you need a modified version of the C startup code. We

recommend that you simply copy the file \ADUC812\EXAMPLE\STARTUP.A51 from the BLINKY example and insert this file into your project. Important is that this file has the statement CSEG AT 2000H before the LJMP STARTUP1 instruction.

Set Tool Options for Using the Monitor-51

 μ Vision2 lets you set options for your target hardware. The dialog **Options for Target** opens via the toolbar icon. In the **Target** tab you specify the CPU Xtal frequency, and the address ranges for the code memory and the xdata memory. Refer to "Memory Mapping on the ADuC812 Board" on page 4 for a discussion.

Options for Target 'Monitor'	? ×	
Target Output Listing C51 A51 L51 Locate L51 Misc Debug		
Analog Devices ADuC812		
⊠tal (MHz): 11.0592 □ Use On-chip ROM (0x0-0x1FFF)		
Memory Model: Small: variables in DATA		
Code Rom Size: Large: 64K program		
Operating system: None		
Off-chip Code memory Start Size: Off-chip Xdata memory Eprom #1: 0x2000 0x5F00 Ram #1: 0x0000 0x2000 Eprom #2:		
Code Banking Start: End: 🗖 Xdata Banking Start:	End:	
Banks: 2 Y Bank Area: 0x0000 0x10000 Banks: 2 Y Bank Area: 0x0000 0x	0000	
OK Cancel Defaults		

In our program example we have used conditional translation to adapt the program code to the Monitor-51. Therefore enter in the **C51** tab under **Define: MONITOR**. Since the interrupt vectors on the ADuC812 board are redirected you need to enter also **Interrupt vectors at address: 0x2000** in this dialog page. Now you may use the #ifdef MONITOR and #ifndef MONITOR in your C sources to modify the program code for using with the Monitor-51 or using on a standard ADuC812 device.

Options for Tar	get 'Monitor'	? ×
Target Outpu	t Listing C51 A51 L51 Locate L51 Misc Debug	
Preprocess	or Symbols	
Define:	MONITOR	
U <u>n</u> define:		
Code Optin Level: Emphasis:	ization 8: Common tail merging Favor execution speed Global Register Coloring Don't use absolute register accesses	Warnings: Warninglevel 2 Bits to round for float compare: 3 Image: 10x2000 Ima
Include Paths Controls Compiler control string	BROWSE INTVECTOR(0x2000) DEFINE (MONITOR) DEBUG	G OBJECTEXTEND CODE
	OK Cancel	Defaults

We have used the same MONITOR symbol also in the STARTUP.A51 file to redirect the reset jump. Therefore you must enter in the A51 tab under **Set: MONITOR**.

Options for Target 'Monitor'		?×
Target Output Listing C51 A51	L51 Locate L51 Misc Debug	
Conditional assembly control Symbol	s	
Set: MONITOR		
<u>R</u> eset		

To configure the μ Vision2 debugger, you select in the **Debug** tab **Use: Keil Monitor-51**. Then use the **Settings** button as described under "Using μ Vision2" on page 4 to configure the serial communication with the ADuC812 board. This completes the tool setup for the ADuC812 board.

Set Tool Options to Download the Application into Flash ROM

You may create with μ Vision2 a second target that allows you to run the program directly from the Flash ROM. μ Vision2 offers **Targets** that allow you different tool settings for the same project. You create a target with the **Project – Target, Groups, Files...** dialog.

Then you use the Project – Options for Target dialog to change the tool settings:

- in the **Target** tab enter the memory layout as shown below.
- the Flash **DOWNLOAD.EXE** utility form Analog Devices needs a Intel HEX-file. Therefore enable in the **Output** tab **Create HEX file**.
- in the C51 tab remove the **Define: MONITOR** are enter **Interrupt vectors at address setting: 0**
- in the A51 tab remove **Set: MONITOR**.

ptions for Target 'Flash'	<u>?</u>
Target Output Listing C51 A51 L51 Locate L51 Mise	c Debug
Analog Devices ADuC812	
<u>X</u> tal (MHz): 11.0592 Memory Model: Small: variables in DATA Code Rom Size: Large: 64K program Operating system: None	☑ Use On-chip ROM (0x0-0x1FFF)
Off-chip Code memory Eprom #1: Eprom #2: Eprom #2:	Off-chip Xdata memory Start Size: Ram #1: 0x0000 0x8000 Ram #2:
Code Banking Start End: Banks: 2 Sank Area: 0x0000 0x10000	Kan #3. Xdata Banking Start: End: Banks: 2 Bank Area: 0x0000
OK Cancel Defaults	

Settings in the Target Tab for using the on-chip Flash ROM.

Targets, Groups, Files	? ×
Targets Groups / Add Files	
	1
_ Target to Add:	
Flash <u>A</u> dd	
Copy all Settings from Current Target	
Available Targets:	
Monitor	
Flash	
Set as Ourrent Tarrent Bemove Tarrent	
UK Cancel	

Using the µVision2 Simulator

In the C51 Version 6.02 package we provide also simulation support for the ADuC812 device. You may configure the μ Vision2 simulator under **Options for Target – Debug**. Select **Use Simulator** to debug your application with the μ Vision2 simulator.

)ptions for Target 'Flash'	? 🗙	
Target Output Listing C51 A51 L51 Locate L51 Misc Debug		
	O Use: Keil Monitor-51 Driver Settings	
Load Application at Startup Go till main()	C Load Application at Startup	
Initialization File:	Initialization File:	
Browse	Browse	
Restore Debug Session Settings Breakpoints Watchpoints & PA Memory Display	Restore Debug Session Settings IF Breakpoints IF Toolbox IF Watchpoints IF Memory Display	
CPU DLL: Parameter:	Driver DLL: Parameter:	
Dialog DLL: Parameter: DP51.DLL -pAD812	Dialog DLL: Parameter: TP51.DLL -pAD812	
OK Cancel Defaults		

The μ Vision2 debugger simulates the ADuC812 specify peripherals and provides in the peripherals menu of the μ Vision2 debugger dialogs for the ADuC812 specific peripherals.

Analog/Digital Converter		×
Analog Digital Converter		
Mode: Power Down	ADCCON1: 0x20	CONVST#
Clock Divide Ratio: 4	ADCCON2: 0x00	CCONV
Acquisition Clocks: 1	ADCCON3: 0x00	I DMA EXC
Sample Rate: 176470	ADCDATA: 0x0000	T2C
Channel: AIN0	DMAPHL: 0x000000	
Analog Input Channels		
AIN0: 0.0000 ANI1:	0.0000 AIN2: 0.0000	AIN3: 0.0000
AIN4: 0.0000 AIN5:	0.0000 AIN6: 0.0000	AIN7: 0.0000
Temp [*C]:	25.0 Vref: 2.5000	

Digital/Analog Converter	×
Digital Analog Converter	
DACCON: 0x07 MODE	Vref: 2.5000 AVDD: 5.0000
Analog Output 0	-Analog Output 1
DACOH: 0x01 PD0	DAC1H: 0x00 PD1
DACOL: 0x20 CLR0	DAC1L: 0x00 CLR1
DAC0: 0.0000	DAC1: 0.0000

Digital/Analog Converter Dialog

Analog/Digital Converter Dialog

The simulator also defines several VTREG's that can be used with the μ Vision2 debug functions. This allows you to write signal functions or to use automated tests. For detailed information about using VTREG's refer to the " μ Vision2 Getting Started with the 8051" User's Guide "Chapter 6. μ Vision2 Debug Functions" (Keil\C51\HLP\GS51.PDF) that is included in the Keil 8051 tool chain.

Conclusion

Keil C51 Version 6 is the most efficient and flexible 8051 development tool platform available. Keil C51 Version 6.02 or higher contains full support for the ADuC812 software development. Once you understand the concept and handling of the μ Vision2 IDE you will find the tool chain easy to use. This allows you to create ADuC812 applications faster than ever before.

On <u>http://www.keil.com</u> we provide you with current product information and a multitude of application notes. We also publish on the Internet our "support solutions" database, containing over 1000 current technical questions and answers.