		User Ma
MIDASI	JINK 在 IAR 下仿真调试报	告
Author	Neil Zhao	

Revision History

Date	Revision History	Reviser
2009-07-07	Rev. V1.0	Neil Zhao

Abbreviation

Table of Contents

MIDASLINK 在 IAR 下仿真调试报告	
实验目的	
实验仪器	2
<u>实验</u> 步骤	2

MIDASLINK 在 IAR 下仿真调试报告

实验目的

完成 MIDASJLINK 仿真器在 IAR 编译环境下的调试报告,方便客户开发利用。

实验仪器

硬件: EVAL-ADuC7060, MIDASJLINK 仿真器 软件: IAR V5.11

实验步骤

MIDASLINK 在 IAR 下可以用 J-Link 进行调试,也可以用 RDI 进行调试,下面将会分别介绍。

1. IAR 的安装

首先安装 ADuC7060 评估软件, 然后修改其目录 C:\ADuC706xV1.1\为C:\ADuC706xV1.0\。对IAR 的安装,可以选择在装 ADuC7060 评估软件的同时安装,也可以在安装完 ADuC7060 评估软件后单独安装,这里选择单独安装介绍。安装完毕 ADuC7060 评估软件后,在 C:\ADuC706xV1.0\IAR 中找到 EWARM-KS-CD-511 程序压缩包,双击打开,解压后会看到如图 1 所示界面。



点击图 1 中的第二个按钮: Instsll IAR Embedded Workbench, 进入图 2 所示界面。



图 2

点击图 2 中的 Get a license, 会自动连接到 IAR 公司网站,填完信息后,会得到一个 license。 网站地址为 <u>http://www.iar.com/Register/?tag=EWARM-KS32-pehx&version=5.11</u>。这样就可以 选择第二项进行安装了。我们这里安装的版本是 5.11,下面的各种配置都是在该版本下进行。 2. IAR 工程的建立

双击 IAR 图标,运行程序,在弹出的对话框中选择 Create new project 选项,随后根据 不同的情况选择相应的语言(asm、C++、C)。注意,只有点击各种语言前的+,展开各个 项,选中其中的 main, OK 项才可以点击建立新的项目。这里点击 C 后的 main 建立一个新 的项目。然后在弹出的对话框中选择要保存的 project 的地址和键入相应的 project 名,点击 保存,生成一个 ewp 文件,即为 project 文件。进入界面后,点击 File 菜单,在下拉项中选 择 Save Workspace,在弹出的框中选择要保存的地址并键入一个 Workspace 名,点击保存, 生成一个 eww 文件,即为 Workspace 文件。这样一个项目及其 Workspace 即建立起来,可 以在 main.c 中编写程序,也可以加入其它的文件,进行编译、连接、生成执行文件以及进 行调试。

3. IAR 的配置

IAR 的配置分为工程配置和编译环境等其它配置,工程配置主要在 Project->Options 中,编译环境配置主要在 Tools->Options 中。如果编译环境没配置正确或者不符合个人习惯的话,使用起来就会比较麻烦。点击 Tools->Options 就会弹出一个对话框来,如图 3 所示。

IDE Options		×
Common Fonts Key Bindings Language ECILOR Messages Project Source Code Control Debugger Stack	Iab size: 8 Indent size: 2 Tab Key Function: ○ ○ Insert tab ○ ○ Indent with spaces EOL characters: PC I Show right margin ○ ○ Printing edge ○ ○ Columns: 80	 Syntax highlighting Auto indent Jonfigure Show line numbers Scan for changed files Show hookmarks Enable yirtual space Remove trailing blanks
		确定 取消 应用 (A) 帮助

点击 Editor 选项卡,可以看到右边有很多选项,这些选项关系到使用的习惯和便捷性。 Tab size 表示 tab 的字符宽度,默认值为 8; Indent size 表示缩进的字符宽度,默认为 2; Tab key function 指的是按下 tab 键所起到的作用,第一个表示插入制表符,第二个表示插入空格, 一般选择插入空格(insert with space)这个选项。

工程设定非常重要,如果没有设置好,有可能编译不通过或者出错。我们开始选择 empty project 模版的时候,配置都是默认的,但是有些关键配置还是需要手动配置的,这里列举了几个关键选项的配置。点击 Project 菜单,进入 Options 选项, IAR EWARM 允许为工作区中的任何一级目录和文件单独设置选项,但是用户必须首先为整个项目设置通用的选项General Options。点击 General Options 选项,选中 Target 选项卡,在 Processor Variant 区域中选中 device,如果和所用的处理器不同,则点击后面器件选择按钮进行选择即可,如图 4 所示。

Category:	
General Options C/C++ Compiler Assembler Output Converter Custom Build Build Actions	Target Output Library Configuration Library Options MISRA C Processor variant
Linker Debugger Simulator Angel GDB Server	Endian mode EPU
IAR ROM-monitor J-Link/J-Trace LMI FTDI Macraigor RDI Third-Party Driver	Big G BE <u>32 C BEB </u>
	OK Cance

China Application Support Team Toll-free number: 800-810-1742 Toll-free fax: 800-810-1747 Email: china.support@analog.com

点击 C/C++ Complier, 在 Output 选项卡中, 勾选 Generate Debug Information, 如图 5 所示。

Options for node "TE	ST IAR"
Options for node "TE Category: General Options 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	ST IAR"
Macraigor RDI Third-Party Driver	OK Cancel

图 5

点击 Linker,选择 Config,在 Linker configuration file 区域中勾选 Override Default,并输入 \$TOOLKIT_DIR\$\examples\AnalogDevices\SupportFiles\ADuC7060_FLASH_Standalone.icf, 如图 6 所示。

Options for node "T	EST IAR"
Category: General Options 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 Third-Party Driver	Factory Settings Config Library Input Output List #define Diagnostics Che • • Unker Governide Gevernide Gevernide
	OK Cancel

图 6

或者选择添加 ADuC7060_FLASH_Standalone.icf, 其路径如图 7 所示。

5



图 7

点击 Debugger 选项,选择 Setup,在 Driver 下拉框中为调试工具,如果采用 J-Link 进行调试,则要选择 J-link/J-Trace,并勾选 Run to,下面填入 main 表示调试从 main 函数开始,如 图 8 所示。

图 8

而后选择 Download, 勾选 Verify download 和 Use flash loader(s), 如图 9 所示。

6

Options for node "TE	ST IAR"	X
Category: General Options 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 Third-Party Driver	Factory Settings Setup Download Attach to program ✓ Yenfy download Suppress download ✓ Use flash loader(s) (default), Edit	
	OK Cancel	

在 Debugger 下面的几项中,只有在图 8 所示的 Driver 下拉框中选中的 J-link/J-Trace 项可以 配置,其它的为灰色,不能配置,如图 10 所示。这里 J-link/J-Trace 可以配置,但采用默认 配置即可,如图 11 所示。

ategory:		Factory Settings
eneral Options		
C/C++ Compiler	RDI	
Assembler	Manufactures BDL dives	
Output Converter	Manuracturer RDI driver	
Custom Build	Browse to your RDI driver	
Build Actions		- Note
Linker	Allow hardware reset	Use the BDI menu to specify
Debugger		additional driver settings. (This
Simulator		menu is available after the RDI
GDB Server	L EIM trace	driver has been located)
TAR ROM-monitor		Catch exceptions
1-Link/1-Trace		□ Reset □ Data □ FI0
LMI FTDI		
Macraigor		
RDI	Log RDI communication	
Third-Party Driver	\$TOOLKIT_DIR\$\cspycomm.log	
		OK Conset

图 10

China Application Support Team Toll-free number: 800-810-1742 Toll-free fax: 800-810-1747 Email: china.support@analog.com

_ _ _ _ _

General Options 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	Setup Connection Reset Software JTAG/SWD speed Auto Initial 32 C Exed 32 C Adaptive	Breakpoints	Catch excepti <u>R</u> eset I <u>Undef</u> <u>S</u> WI <u>D</u> ata <u>P</u> refetch	Facto	ry Settings
Angel GDB Server IAR ROM-monitor J-Link/J-Trace LMI FTDI Macraigor RDI Third-Party Driver	C <u>E</u> xed 32 C <u>A</u> daptive	kHz	☐ <u>S</u> WI ☐ <u>D</u> ata ☐ <u>P</u> refetch		

如果采用 RDI 进行调试,则要选择 RDI,并勾选 Run to,下面填入 main 表示调试从 main 函数开始,如图 12 所示。

Options for node "TE	ST IAR"		×
Category: General Options 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 Third-Party Driver	Setup Download Extra Options Plugins Driver ✓ Run to RDI ✓ main Setup macros ✓ Use macro file	Factory Settings	
	ΟΚ	Cancel	

图 12

而后选择 Download, 勾选 Verify download, 如图 13 所示。

8

_ _ _ _ _ _ _ _ _ _ _ _ _

Options for node "TE	ST IAR"	X
Category: General Options 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 Third-Party Driver	Setup Download Extra Options Plugins Attach to program Verify download Suppress download Use flash loader(s)	Factory Settings
,	OK	Cancel

在 Debugger 下面的几项中,只有在图 12 所示的 Driver 下拉框中选中的 RDI 可以配置,其它的为灰色,不能配置,如图 14 所示。

Options for node "TE	ST IAR"	
Category: General Options C/C++ Compiler Assembler Output Converter Custom Build Build Actions Linker Debugger Simulator Angel GDB Server IAR ROM-monitor I-Link/J-Trace LMI FTDI Macraigor RDI Third-Party Driver	Setup Connection Breakpoints Reset 0 Software 0 JTAG/SWD speed 0 Auto 1 Initial 32 KHz Eixed 32 KHz Adaptive Prefetch	Factory Settings
	ОК	Cancel

图 14

这里 RDI 可以配置,在 Manufacturer RDI driver 下填入 C:\Keil\ARM\Segger\JLinkRDI.dll, 如果该地址下没有 JLinkRDI.dll,可以在电脑中搜索 JLinkRDI.dll 文件,找到后填入对应的地址。其它项采用默认配置即可,点击 OK 完成该项设置,如图 15 所示。

⁹

Options for node "TE	ST IAR"	
Category: General Options		Factory Settings
C/C++ Compiler Assembler	RDI	1
Output Converter Custom Build	Manufacturer RDI driver C:\Keil\ARM\Segger\JLinkRDI.dll	
Build Actions Linker Debugger	Allow hardware reset	Note Use the RDI menu to specify additional driver settings. (This
Angel GDB Server	ETM trace	menu is available after the RDI driver has been located)
J-Link/J-Trace		□ Reset □ Data □ EIQ □ Undef □ Prefetch
Macraigor RDI Third-Party Driver	Log RDI <u>communication</u> \$TOOLKIT_DIR\$\cspycomm.log	<u></u>
,		OK Cancel

图 15

然后点击主菜单 RDI 项,在下拉框中选择 Configuration 项,如图 16 所示。

🔏 IAR Embedded Wor	kbench IDE	
File Edit View Project	RDI Tools Window Help	
🗅 🚔 🖬 🕼 🎒	Configure	🚽 🗸 🏷 🦎 🗽 🔝 🗈 🤣 🎒 🖬 👯 🕺 💆
Workspace	ETM Trace Window	
Debug	Trace Setup	
Files	Irace Save	
🗆 🖸 TEST IAR	Breakpoint Usage	

图 16

在弹出的框中,选择 Flash 选项卡,勾选 Enable flash programming,在 Device 的下拉框中选择 Analog Devices ADuC7060,其它采用默认配置,点击确定即可。

4. IAR 的调试

```
{
    while(length>0)
    length--;
}
int main(void)
{
```

POWKEY1 = 0x1;

10

```
POWCON0 = 0x78;  // Set core to max CPU speed of 10.24MHz
POWKEY2 = 0xF4;
GP1DAT = 0x40000000;  // Configure P1.6 as an output
while(1)
{
        delay(1000);
        GP1DAT ^= 0x00400000;
}
```

}

编写完程序后,点击 make 按钮,编译连接,接着点击 debug 按钮,进行调试,如图 17 所示。可以进行单步调试,也可以全速运行,如图 18 所示。

🔏 IAR Embedded Wor	kbench IDE		
<u>File Edit View P</u> rojec	rt <u>T</u> ools <u>W</u> indow <u>H</u> elp		
	5 m m ~ /	· · · · · · · · · · · · · · · · · · ·	× 🗩 💭
Workspace ×	main.c	T	<u>^</u> •×
Debug Files Imain c Imain c </td <td><pre>void delay(int length) { void delay(int length) length-0) length; } int main(void) { POWEEY1 = 0x1; POWEEY2 = 0xF4; FOUWET2 = 0xF4; GPIDAT = 0x40000000; vhile(1) { delay(100000); GPIDAT ^= 0X00000000; } } </pre></td> <td>کی سیل // Set core to max CPU speed of 10.24Mnz // Configure Pl.6 as an output</td> <td>∰≩ debug, i¶it</td>	<pre>void delay(int length) { void delay(int length) length-0) length; } int main(void) { POWEEY1 = 0x1; POWEEY2 = 0xF4; FOUWET2 = 0xF4; GPIDAT = 0x40000000; vhile(1) { delay(100000); GPIDAT ^= 0X00000000; } } </pre>	کی سیل // Set core to max CPU speed of 10.24Mnz // Configure Pl.6 as an output	∰≩ debug, i¶it
hi) fo		v N
* Messages			
Building configure Updating build tre Configuration is up	ation: hi - Debug ie p-to-date.		
	////		>
Build Debug Log			×
Make the active project	(build files as needed)	Errors 0, Warnings 0	Ln 38, Col 11 数字 /

图 17

China Application Support Team Toll-free number: 800-810-1742 Toll-free fax: 800-810-1747 Email: china.support@analog.com

💥 IAR Enbedded Vorkbench	IDE		
<u>F</u> ile <u>E</u> dit <u>V</u> iew <u>P</u> roject <u>D</u> ebug	Disassembly J-Link Tools Window Melp		
D 🛩 🖬 🕼 📇 X 🖻 🖻	レロー	<u>s</u>	
5 - 526888	1		
Workspace ×	main.c	c Disassemble	×
Debug 单步运行 👤	void delay(int length)	Goto	<u> </u>
Files 82 Bit		Next	label is a Thu <mark>k</mark>
🗆 🕞 hi - Debug 🖌	while(length>0)	main	
	length;	000	380110 B501
- 🖽 🚞 Output	,	POWI	<pre>KEY1 = 0x1; 000112 400C</pre>
io_macr	int main(void)	000	J80112 480C
ioaduc7		000	J80116 6001
🖵 🖽 🔄 Output	POWKEY1 = 0x1;	PO00	080118 480B
	POWCONO = 0x78; // Set core to max CPU speed of 10.24Mh	000	J8011A 2178
	PUWKEYZ = UxF4;	POW	J8011C 6001 /FV2 = 0vF4
II		000	J8011E 480B
	GPIDAT = 0x40000000; // Configure P1.6 as an output	000	J80120 21F4
	a contraction and the second sec	GP1I	$DAT = 0 \times 400000$
		000	J80124 480A
	while(1)	000	080128 05C9
	A 1	000	J8012A 6001
	-	Next	label is a Thu
		??ma:	in_1: Y
hi l	fo (<u> </u>
× Log			^
Tue May 19 14:03:31 2009: DLL	version: V3.78a, compiled Dec 4 2007 22:18:35		
Tue May 19 14:03:31 2009: Fim	nware: J-Link ARM V6 compiled Dec 03 2007 17:34:18		
Tue May 19 14:03:31 2009: JT4	AG speed is initially set to: 32 kHz		100 C
Tue May 19 14:03:31 2009: Tot	allRLen = 4, IRPrint = 0x01		
Tue May 19 14:03:31 2009: Tot	allRLen = 4, IRPrint = 0x01		
Tue May 1914:03:31 2009: Hal	ting CPU core		
Tue May 19 14:03:31 2009: Soft	ware reset was performed		
T 14 101402 21 2009: Inte	areserwas penumeu		~
Build Build			×
Ready	Ln 38, (Col 11	数字

观察评估板,会看到 P1.6 连接的 LED 随着单步执行,明暗相间的闪烁,说明在线调试成功。 如果遇到用中断的情况,应该在程序中加入_enable_interrupt();函数,这个函数是 IAR 中用 来打开所有中断的一个程序,它和其它的中断寄存器配置结合起来,才能完成中断的整个过程。

NOTES

1. 当使用新出的 ADUC 系列产品时,建议更新安装目录下的 mIDASLink DLL 文件以保证 与 新 的 内 核 相 兼 容 , 最 新 的 mIDASLink DLL 文 件 可 在 http://www.segger.com/download_midaslink.html下载。

2. 请保证在建立工程的过程中所选器件即为要调试的器件,如果发现不符合,可在 Project Options 中的 Device 选项里进行修改。

China Application Support Team Toll-free number: 800-810-1742 Toll-free fax: 800-810-1747 Email: <u>china.support@analog.com</u>
