

3.1 实验目的

• 熟悉在 Keil 工具中编写 C 语言程序;

• 熟悉 C 语言程序的调试。

3.2 实验设备

1. 硬件

PC一台。

- 2. 软件
- (1) Windows 7/8/10 系统;

(2) Keil µVision5 集成开发环境。

3.3 实验内容

在 Keil 工具中通过 C 语言完成 sum = 1+2+···+10 的程序编写,并在仿真模式下进行调试。

3.4 实验预习

- 复习C语言编程;
- 学习在 Keil 工具下如何建立工程及配置。

3.5 实验原理

在调试该 C 语言程序时, 配置本实验教材所使用的开发板, 或以仿真模式运行。

3.6 实验步骤

3.6.1 创建工程

在 Keil 工具中创建一个新工程,在实验 2 创建工程的基础上在 USR 中添加 main 函数。

3.6.2 修改配置

单击如图 3.1 所示方框中的按钮,打开配置界面。



图 3.1 打开配置界面按钮

切换到 Debug 选项卡,如图 3.2 所示。

evice Targe	t Output Listing User C/C++ (Ad	26) Asm Li	nker Debug Utilities		
C Use Simula	tor <u>with restrictions</u> <u>Settings</u> to Real-Time	• Use: ULINK	C2/ME Cortex Debugger Settings		
 Load Applic Initialization File 	cation at Startup 🔽 Run to main()	Load Applicat	tion at Startup 🔽 Run to main()		
	Edit		Edit		
Restore Deb	ug Session Settings	Restore Debug	Session Settings		
F Breakpo	oints 🔽 Toolbox	Breakpoints Toolbox			
₩ Watch	Windows & Performance Analyzer	Vatch W	indows 🔽 Tracepoints		
Memory	Display 🔽 System Viewer	Memory D	lisplay 🔽 System Viewer		
CPU DLL:	Parameter:	Driver DLL:	Parameter:		
SARMCM3.DL	L -REMAP -MPU	SARMCM3.DLL	-MPU		
Dialog DLL:	Parameter:	Dialog DLL:	Parameter:		
DCM.DLL	-pCM4	TCM.DLL	-pCM4		
Warn if out	dated Executable is loaded	□ Warn if outda	ted Executable is loaded		
	Manage Comment M	Desetation Di	- 1		
	Manage Component Vie	ewer Description Fil	es		

图 3.2 Debug 选项卡

将选项卡中的内容改为如图 3.3 所示,注意粗黑框标注的内容,具体修改见下文。

C Use Simulator with restrictions Settings Limit Speed to Real-Time	C Use: ULINK2/ME Cortex Debugger J Settings 取消勾洗
▼ Load Application at Startup ▼ Run to main() Initialization File: 添加init.ini文件	✓ Load Application at Startup ✓ Run to main() Initialization File:
NUSR Vinit ini	Restore Debug Session Settings
CPU DLL: Parameter: 修改	Driver DLL: Parameter:
SARMCM3.DLL -REMAP -MPU	SARMCM3.DLL -MPU
Dialog DLL: Parameter:	Dialog DLL: Parameter:
DARMSTM.DLL pSTM32F429IGT6	TCM.DLL pCM4
Warn it outdated Executable is loaded Manage Component V	Warn if outdated Executable is loaded

图 3.3 需要修改 Debug 选项卡的内容

其中:

Use Simulator 代表要使用软件模拟仿真。

Dialog DLL 的内容是 DARMSTM. DLL,表示使用 STM 系列。

Parameter 的内容是"-pSTM32F429IGT6",这里匹配的是当前开发板的型号。

Initialization File,该选项可以配置映射一些地址,从而实现对 STM32 开发板不同功能的模拟仿真。一个典型的 Initialization File 内容如下:

map	0x4000000,0x4000ffff	read	write
map	0x40010000,0x4001ffff	read	write
map	0x40020000,0x4002ffff	read	write
map	0x48000000,0x4800ffff	read	write

在工程文件夹下新建 init. ini 文件,将上述 4 行内容添加到这个文件中,在 Debug 选项 卡的 Initialization File 选项下添加该文件。

在该文件中,我们设置了4段地址映射,使得可以实现模拟某些功能。一个大致的地址 功能映射表如图 3.4 所示。

上面只是截取了部分存储器的映射情况,这是 AHB1 的地址。例如,如果需要使用 RCC 功能,那么可以看到 0x40021000 至 0x400213FF 这部分地址需要有读写属性,可以在 Initialization File 中添加相应映射地址的字段来启用读写。注意:该地址对于不同型号的 开发板会有变化。

Debug 选项卡中的内容修改完毕后,单击 OK 按钮完成配置。

总线	编址范围	大小	外设	
	0x4002 4000 - 0x4002 43FF	1 KB	TSC	
	0x4002 3400 - 0x4002 3FFF	3 КВ	Reserved	
	0x4002 3000 - 0x4002 33FF	1 KB	CRC	
	0x4002 2400 - 0x4002 2FFF	з КВ	Reserved	
AHB1	0x4002 2000 - 0x4002 23FF	1 KB	FLASH 接口	
	0x4002 1400 - 0x4002 1FFF	3 КВ	Reserved	
1	0x4002 1000 - 0x4002 13FF	1 KB	RCC	

图 3.4 部分存储器的映射

3.6.3 跟踪变量

配置完成后,就可以通过 Debug 进行仿真了。本次实验中,我们通过一个简单的累加 来仿真,验证程序的正确性。main.c 文件内容如下:

在编写好程序后,单击 Keil 左上角的 Build 或 Rebuild 按钮进行编译,如图 3.5 所示。



图 3.5 重新编译

在如图 3.6 所示的 Output 框中看到无错误(0 Error(s))提示后即编译成功。 编译完成后,单击 Debug 按钮,然后单击分析器按钮,如图 3.7 所示。

```
Rebuild started: Project: CProgram

*** Using Compiler 'V6.15', folder: 'C:\Keil_v5\ARM\ARMCLANG\Bin'

Rebuild target 'Target 1'

compiling main.c...

assembling startup_stm32f429xx.s...

compiling system_stm32f4xx.c...

linking...

Program Size: Code=376 RO-data=444 RW-data=0 ZI-data=1632

".\Objects\CProgram.axf" - 0 Error(s), 0 Warning(s).

Build Time Elapsed: 00:00:00
```

图 3.6 重新编译成功的 Output 框

gram.uvprojx - µVision g Peripherals Tools SVCS Window	Help	-		×
← → 作 作 作 作	//= //= 🎒 PendSV	v 🗟 🕫	Q •	٠
D 🔯 🖬 🖶 🚑 - 🗆 - 🖳 -	☆ - 回 - 图 - 次 -	1. 单击De	bug按钅	H 7 ×
Setup Save 	 ▶ Performance Analyzer 2.単i 2.単i ※ System Analyzer ↓ Event Recorder ■ Event Statistics 	自启用分析器	n Tra	nsition v Nex

图 3.7 打开 Debug 窗口并启用分析器

单击分析器左上角的 Setup 按钮,如图 3.8 所示。

				1	
Load	Min Time	Max Time	Grid	Zoom	Min/Max U
Save	0 s	0.148625 ms	1 ms	In Out All	Auto Undo
	1	1	1	1	1
	.oad Save	oad? Min Time Save? 0 s	oad? Min Time Max Time Save? 0 s 0.148625 ms	oad? Min Time Max Time Grid Save? 0 s 0.148625 ms 1 ms	.oad ? Min Time Max Time Grid Zoom Save ? 0 s 0.148625 ms 1 ms In Out All

图 3.8 单击 Setup 按钮

在如图 3.9 所示界面中单击右上角标识处添加变量 x(如果已经有则不用添加,关闭此 框即可),添加完毕后单击 Close 按钮。

10 ×
,
Display Range Max: 0.0 Min: 0.0
>> Shift
Import Signal Definitions

图 3.9 添加变量

在分析器中左侧看到变量 x,即添加成功,如图 3.10 所示。请注意,需要在分析器中跟踪的变量必须为全局变量,若设置为局部变量,则可能无法添加到分析器中。



图 3.10 添加变量成功

因为分析器展示的默认范围非常广 (和变量类型上下限有关),因此需要设 置自动调整范围以更好地展现值的变 化。在分析器窗口中空白位置右击,选 中 Adaptive Min/Max 选项,如图 3.11 所示。

单击 Keil 左上角的 Run 按钮,可以 看到分析器中出现波形,如图 3.12 所示。

图 3.11 设置仿真变量的最大/最小值

Hexadecimal Values

Adaptive Min / Max

Remove Signal

Add " to.

Setup Min/May from Recording

Restore previous Min/Max

Analog

Bit State

[68 us]

artup_stm3

a 🖬 Call :

c Anab



图 3.12 观察变量的波形

可以看到,波形从 0~10 以阶梯形式上升,符合程序设定,每次加 1 直到 10 的结果是一致的。

请注意,如果 Initialization File 没有被加载或者没有正确配置地址,那么在 Command 窗口会出现如图 3.13 所示的错误信息。此时需要重新配置 Initialization File。

Comr	mand								
Load WS WS WS WS	d "E:\\ 1, 'US) 1, 'cnt 1, 'rcc 1, 'ms	Ear: ART_I ts C_clo	12\\STM: RX_TEST DCks.HC	32\\NEU_Co	ceBo cy	oard-STM32F4	10	7VG'	\\Objects\\NEU_CoreBoar
WS :	1, 'val	Lue							
WS :	1, 'Tin	ningl	Delay						
WS :	1, 'Sys	stem	CoreClo	ck					
WS :	1, `x								
LA	`x								
***	error	65:	access	violation	at	0x40023800	:	no	'read' permission
***	error	65:	access	violation	at	0x40023800	:	no	'write' permission
***	error	65:	access	violation	at	0x40023808	:	no	'write' permission
***	error	65:	access	violation	at	0x40023800	:	no	'read' permission
***	error	65:	access	violation	at	0x40023800	:	no	'write' permission
***	error	65:	access	violation	at	0x40023804	:	no	'write' permission
***	error	65:	access	violation	at	0x40023800	:	no	'read' permission
***	error	65:	access	violation	at	0x40023800	:	no	'write' permission
***	error	65:	access	violation	at	0x4002380C	:	no	'write' permission

图 3.13 初始文件配置错误信息

此外,还有一种在设置变量时失败的情况,是在最新版 Keil µVision5 中,默认的 ARM 编译器会使用版本 6 的编译器,在该版本的编译器进行编译后,在 Debug 选项卡的 Setup 里 无法设置变量,提示 Unknown Signal。解决方案是在 Target 选项卡中将 ARM Compiler 切换到版本 5 或以下的版本,如图 3.14 所示。

TMicroelectronics STM32F429IGTx					Code C	Generation Compiler:	Use defau	It compiler vers	sion 6 💌	
perating	g system:	None				Use defau V6.15	ault compiler version 5			
System Viewer File:						Use MicroLIB Dig Endion				
STM32	429x.svd				Floatin	ng Point H	ardware: Sin	gle Precision	•	
Use	Custom Fi	le								
Read/	Only Mem	ory Areas			-Read/	Write Men	nory Areas			
default	off-chip	Start	Size	Startup	default	off-chip	Start	Size	Nolnit	
Г	ROM1:			- C	Г	RAM1:				
	ROM2:			- C	Г	RAM2:				
	ROM3:		-	- C	Г	RAM3:				
Г	on-chip	·				on-chip				
	2011 A. C.	0x8000000	0×100000	•	2	IRAM1:	0x20000000	0x30000	Г	
٦ ٩	IROM1:				1242			-		

图 3.14 切换 ARM 编译器版本

3.7 实验参考程序

3.8 实验总结

本实验给出了如何使用 Keil 软件进行仿真的例子:从 Keil 软件仿真器中模拟运行一个简单的累加计算。初学者经过本节的实验理解和体验 STM32 的仿真过程,熟练掌握 Keil 软件的 STM32 系列开发板的仿真器使用方法。

3.9 思考题

(1) 尝试通过配置其他映射地址, 启用 GPIO 引脚, 观察 GPIO 引脚高低电平变化。

(2) 使用 Debug 模式下的 Watch 功能观察变量的变化,并比较和仿真分析器的相同和 不同点。

(3) 查阅资料了解不同型号的 STM32 的地址映射配置。