# **Cortex M0 Assembly Programming**

#### SUMMARY

- 1. Consider you have a simple Cortex M0 based Micro-controller with the following memory map
  - a. Internal SRAM 1 KB Starting Address 0x0000 0000
  - b. A LED Peripheral One word Address 0x5000 0000
- 2. We will write assembly program to toggle the LEDs and simulate the program
- 3. We will analyze the resulting binary file and the disassembly file
- 4. In the Next Section we will see how to design AHB-Lite Compliant LED Peripheral

#### INSTALL KEIL MDK

ARM University Program

Copyright © ARM Ltd 2013

- 1. Install KEIL MDK. You can download the software suite from http://www.keil.com/arm/mdk.asp
- 2. Install ARM Cortex M Profile Device Pack: KEIL ARMCORTEX:DFP as shown below,





3. If you have already installed KEIL MDK-ARM without any packs, you can install the same by clicking pack installer as shown below,

🐺 μ\	/ision						- terms in
File	Edit	View	Proje	t Flash	Debug	Peripherals Tools SVCS Window Help	
	2	9	8	6 <b>6</b>	9 0	← →   巻 魯 魯 魯   譯 譯 //』 //漫   ⑳ SysTick_Config	- 🗟 🥐 🤇
۲		iii 🧼		OAD		🕞 🔊 🔚 🖶 🗇 🎯 <sub>►</sub>	
Projec	ct				₽ 💌	Pack Installer Install Trupdate Software Packs that contain offware Components	

4. Install the license key provided. Steps to install are given in KEIL MDK License Installation Guide.

# CREATE A NEW PROJECT

1. Create a new project

ARM University Program

Copyright © ARM Ltd 2013



🐺 μ\	Vision													-	-	-	-
File	Edit	View	Proj	ect Fl	ash	Debug	Peripherals	Tools	SVCS	Window	Help						
	1			New µ	Vision	Project	•										6
		M 🌽	7	New M	lulti-Pr	oject W	orkspace										
Proje				Open F	Project												
Floje				Save Pr	roject i	n µVisio	n4 format										
				Close F	Project												
				Export												+	
				Manag	je											×	
				Select I	Device	for Targ	et										
				Remov	e Item												
			24	Option	1S										Al	t+F7	
				Clean t	targets												
				Build t	arget											F7	
				Rebuil	d all ta	rget file	s										
			$\langle \rangle$	Batch 8	Build												
			٨	Transla	ite										Ctr	1+F7	
				Stop b	uild												
11																	

- 2. Give the project name as "lab"
- 3. In select Device Target, choose ARM  $\rightarrow$  Cortex M0

Select Device for Target 'Target 1'		×
CPU Vendor: ARM Device: ARMCM0 Toolset: ARM Search: ARM ARM Cortex M0 ARM Cortex M0 ARM Cortex M0 ARM Cortex M3 ARM Cortex M4	Description: The Cortex <sup>**</sup> -M0 processor is an entry-level 32-bit ARM Cortex process for a broad range of embedded applications. It offers significant I including: • simple, easy-to-use programmers model • highly efficient ultra-low power operation • excellent code density • deterministic, high-performance interrupt handling • upward compatibility with the rest of the Cortex-M processor fa	sc 🔺
	OK Cancel He	alp



- 4. Skip the next step "Manage Run Tim environment" by clicking cancel.
- 5. Add file cm0dsasm.s into the project,

**ARM University Program** Copyright © ARM Ltd 2013

😨 C:\Users\kars	rshi01\Desktop\Workshop\USB\Full Material - V2\P2\Lab0\Softw	are\Lal
File Edit Vie	iew Project Flash Debug Peripherals Tools SVCS W	indow
🗅 😅 🖬 🕯	● 2 月 2 日 16 日 10 日 10 日 10 日 10 日 10 日 10 日 20 日 20	*
🔗 🍱 🎬 🤅	🧼 🚉   🙀   Target 1 💿 💉   🛃 🗟 🗇	<u>()</u>
Project	4 🖸	
🖃 🔁 Target 1	L	
Sour	rce Group 1	
	Add New Item to Group 'Source Group 1'	
7	Add Existing Files to Group 'Source Group 1'	
	Add Group	
	Remove Group Source Group 1 and its Files	
	manage Project Items	
	Open File	
	Open List File	
	Open Map File	
	Open Build Log	
	Rebuild all target files	
	Build target F7	
	Translate File	
	Stop build	
	Show Include File Dependencies	

6. Choose "ARM Source Files" in the drop down "File Types" and then select cm0dsasm.s



Look in: Noftware			
Name	Date modified	Туре	Size
i main.c	13/04/2014 11:12	C File	1
retarget.c	13/04/2014 11:02	C File	
•			
File name:			Add
Files of type: C Source file (*.c)		•	Close
C Source file (*.c) Asm Source file (*.s*; *.src; *.a Object file (*.obj)	")		

- 7. Once you select the file, click "Add" and then click "Close"
- 8. The will be added into "Source Group1" under "Target1"

**ARM University Program** 

Copyright © ARM Ltd 2013

9. Right click on Target1 in project navigator and click "Options for Target1"



🐺 C:\	Users\k	arshi	01\Desktop\Workshop\US	B\Full Material - V2	\P2\Lab0\Software\Lab0.uv	proj - µVision	
File	Edit	View	Project Flash Debug	g Peripherals To	ols SVCS Window He	p	
	📬 🔛	0	8 🗈 🛍 🧐 M	🗢 🔿   🥐 🦿	1.18.18.1年年7月。	//👷 🛛 🖄 SysTi	ick_Config
1	🖄 🔛	1 🧼	Target 1	<b>•</b> K	📥 🕾 🗇 🚳		
Project	t		<b>д </b> 🖬	Cm0dsasn	1.5		
0-70	Targe	+ 1		· · ·	1		
i 🗄	Sc 🔁	*	Options for Target 'Target	1' Alt+F7	dule4: A Simple S	oC Applic	ation
			Add New Item to Group		ggle LEDs at a gi	ven freque	ency.
			Add Existing Files to Grou	0			
1			Add Group				
			Demove Item				
			Manage Project Barry		ctor Table Mapped	to Addres	s u at
			Manage Project Items			PRESERV	Æ8
			Open File			THUMB	
			Open List File			1051	DECET
			Open Map File			EXPORT	Vect
			Open Build Log				_
					ctors	DCD	0x0000
			Rebuild all target files			DCD	Reset_
			Build target	F7		DCD	0
			Translate File			DCD	õ
			Stop build			DCD	0
			Stop Band			DCD	0
		$\checkmark$	Show Include File Depend	encies		DCD	0
		_		24	_	DCD	0
				25		DCD	0
				26		DCD	0
10							

10. This opens up the configuration window for your project

#### CHANGING THE CONFIGURATION OPTIONS

1. Go to the linker tab and delete the R/W Base entry.

ſ	😰 Options for Target 'Target 1'	<b>-X</b>
	Device       Target       Output       Listing       User       C/C++       Asm       Linker       Debug       Utilities         Use       Memory Layout from Target Dialog	
	Scatter        Edit         Misc controls        Edit         Linker control string      cpu Cortex-M0 *.o -ro-base 0x00000000 -entry 0x00000000 -entry Reset_Handler -firstVectorsinfo sizes -info totals -info veneers	
	Cancel Defaults Hel	p

**Configuration in Linker Tab** 



ARM University Program Copyright © ARM Ltd 2013

2. Go to the debug tab and change the choice to "USE SIMULATOR"

Options for Target 'Target 1'	X
Device Target Output Listing User C/C++ Asm	Linker Debug Utilities
• Use Simulator             □ Limit Speed to Real-Time	© <u>U</u> se: ULINK2/ME Cortex Debugger <b>▼</b> Settings
I       Load Application at Startup       Image: Run to main()         Initialization File:       Image: Run to main()	I     Load Application at Startup     I     Run to main()       Initialization File:
Edit	Edit
Restore Debug Session Settings	Restore Debug Session Settings
✓ Breakpoints ✓ Toolbox	Breakpoints     Toolbox
Watch Windows & Performance Analyzer	Watch Windows
Memory Display System Viewer	Memory Display System Viewer
CPU DLL: Parameter:	Driver DLL: Parameter:
SARMCM3.DLL	SARMCM3.DLL
	, ,
Dialog DLL: Parameter:	Dialog DLL: Parameter:
DARMCM1.DLL pCM0	TARMCM1.DLL PCM0
J	1 J. J.
OK Ca	ncel Defaults Help

- 3. Go to USER tab and place these fromelf commands in "RUN THESE COMMANDS AFTER BUILD" section.
  - a. fromelf -cvf Objects/lab.axf --vhx -32x1 -o lab.hex
  - b. fromelf -cvf Objects/lab.axf -o disasm.txt

ARM University Program

Copyright © ARM Ltd 2013

More information about fromelf utility can be found here: http://www.keil.com/support/man/docs/armutil/armutil\_caccdhia.htm



-Run User Prog	rams Before Compilation of a C/C++ File
🔲 Run #1:	🗖 DOS16
	Stop Build/Rebuild #1 on Exit Code: Not Specified
🔲 Run #2:	🗖 DOS16
	Stop Build/Rebuild #2 on Exit Code: Not Specified
Run User Progr	ams Before Build/Rebuild
- Run #1:	🗖 DOS16
🔲 Run #2:	🗖 DOS16
- Run User Prog	rams After Build/Rebuild
Run #1:	fromelf -cvf lab.axf -vhx32x1 -o lab.hex
🔽 Run #2:	fromelf -cvf lab.axf -o disasm.txt
_	

## BUILD THE PROJECT

ARM University Program

Copyright © ARM Ltd 2013

- 1. Build Target (Project  $\rightarrow$  Build Target)
- 2. Analyze disasm.txt and lab.hex file generated in your project directory and compare with the source file cm0dsasm.s



🕼 disasm.txt (~\Desktop\Wo	rkshop\USB\Full Mate	erial - V2\P	6\Software)	- GVIM1				
File Edit Tools Syntax	Buffers Window I	Help						
9 8 <b>6</b> 8 6 8 6 8	X 🗉 🛍 🖏	Ð. 🔁	<b>å</b> 🕹 🖇	\$ ግ <b>ଘ⊡ ?</b> ዪ		Code.hex (~\D	Software) - GVIM	
RESET						File Edit Tools	Syntax Buffers	Window Help
Vectors								
0×0000000:	000003fc		DCD	1020			96196	🖽   😘 💁 🕄
0x0000004:	00000081		DCD	129		ดดดดดดดดดดดดดดดดดดดดดดดดดดดดดดดดดดดดดดด		
0×0000008:	00000000		DCD	0		66666666		
0x000000c:	00000000		DCD	0		6666666		
0×0000010:	00000000		DCD	0		000000000		
0x0000014:	00000000		DCD	0		000000000		
0x0000018:	00000000		DCD	0		តតតតតតត		
0x000001c:	00000000		DCD	0		0000000		
0×0000020:	00000000	••••	DCD	5		000000000		
0×0000024:	00000000	••••	DCD	<b>U</b>		000000000		
0x00000028:	00000000		DCD	U		000000000		=
0x000002c:	00000000		DCD	9 0		0000000000		
0×0000030:	00000000	••••	DCD	8		0000000000		
UX UUUUU034:	00000000		DCD	8		0000000000		
0×00000038:	00000000		DCD	8		000000000		
0×0000003c:	00000000		DCD	0		000000000		
0x0000040:	00000000	••••	DCD	0		000000000		
0x00000044:	00000000	••••	DCD	0		000000 <mark>00</mark>		
0x 00000048 :	00000000	••••	DCD	8		000000000		
UX UUUUUU4c :	00000000		DCD	8		000000 <mark>00</mark>		
0×00000050:	00000000		DCD	0		000000000		
0x00000054:	00000000		DCD	0		000000000		
0x00000058:	00000000	••••	DCD	0		000000 <mark>00</mark>		
0x000005c:	00000000		DCD	8		000000 <mark>00</mark>		
0×00000060:	00000000		DCD	8		000000 <mark>00</mark>		
0X 00000064:	00000000		DCD	0		000000 <mark>00</mark>		
0×00000068:	00000000		DCD	0		000000 <mark>00</mark>		
0×0000000c:	00000000		DCD	0		000000 <mark>00</mark>		
0200000070:	00000000	••••	DCD	0		000000 <mark>00</mark>		
0200000074:	00000000		DCD	0		000000 <mark>00</mark>		
0X 00000078:	00000000	••••	DCD DOD	0		48074906		
0X0000007C:	99999999		DCD	U		48 07 6 0 <mark>88</mark>		
şi tovt						D1FD1E <mark>40</mark>		
Pocot Handlor						48 06 49 03		
0v00000000	40.06	т	LDR	и1 [nc #24] • [8v0c] - 8v	5000000	48 046 0 <mark>08</mark>		-
020000000000000000000000000000000000000	4900		LDR	$r_{0}[pc, #24]$ , $[0x_{2}c] = 0x_{2}$	255		11,1	33%
0x 00000002.	6008		STR			<u> </u>	11	
0x 00000004.	4807	.н	LDR	r0.[nc.#28] : [0xab] = 0	*2fffff			
0x00000000	1648	a.	SUBS					
Ax AAAAAAAA	dlfd		BNF	Ax88 : Reset Handler + 8				
9×9999986	49 03		I DR	r1.[nc.#12] : [Ax9c] = Ax	x50000000			
0x0000008e:	4806		LDR	r0.[pc.#24] : [0xa8] = 0	xaa			
0x 00000000 0 :	6998		STR	r0.[r1.#0]				
0×00000092	48.04	.н	LDR	r0.[nc.#16] : [0xa4] = 0	x2fffff		-	
				-, [kolino] , [oug4] W		78.51 27%		

# USING THE SIMULATOR

**ARM University Program** 

Copyright © ARM Ltd 2013

- 1. After you have compiled go to Debug  $\rightarrow$  Start/Stop Debug Session
- 2. Ignore the warning message
- 3. Note that PC is already pointing to the Reset\_Handler (This matches the entry point flag set during compilation)
- 4. Goto Debug → Memory Map and add the LED peripheral Memory information. Note the region should be Read and Write.



File Edit View	Project Flash	Debug Peripherals Tools SVCS Window Help
🗋 🞽 🛃 🧭	👗 🖻 🖺 🛛	Start/Stop Debug Session Ctrl+F5
RST 🗄 🚳 🗍	P ⊕ ⊕ +0	Reset CPU 🔜 👻
Registers		≣↓ Run F5
Register	Value	Stop
Core	0.0000000	
RU R1	0x00000000	🔂 Step Over F10 r
	0x00000000	{} Step Out Ctrl+F11
R3	0x00000000	*{} Run to Cursor Line Ctrl+F10
R5	0x00000000	✤ Show Next Statement
R6	0x00000000	Province Chilling
R7	0x00000000	EA  .
	0x00000000	Insert/Remove Breakpoint     P9
R10	0x00000000	C Enable/Disable Breakpoint Ctrl+F9 DC
R11	0x00000000	
R13 (SP)	0x000003FC	Ctrl+Shift+F9
R14 (LR)	0xFFFFFFFF	OS Support
TR15 (PC)	0x00000080 0x01000000	Execution Profiling
± Banked		Memory Map
System		Inline Assembly
Mode	Thread	Function Editor (Open Ini File)
Stack	MSP	BS Loo
States	0	Debug Settings
360	0.0000000	68 LDR
		70 STR
Memory	Мар	×
Curren	it Mapped	
000:	0x0000000 - 0	0x0007FFFF exec read
001:	0x20000000 - 0	0xE000FFFF read write
		-
		Kill Selected Range
- Man	Bange - Examr	ple: 0x4000000 0x4000FFFF
		2020102
	0000000. 0x500	
Ox5		
0x5	Read	
Dx5	Read	
0x5	✓ <u>Read</u> ✓ Write	

ARM University Program Copyright © ARM Ltd 2013



5. View the contents of at 0x5000\_0000 in Memory1 Window.

C:\Users\karshi	01\Desktop\Workshop\U	ISB\Full Materia	al - V2\P2\Lab0\Softwar	Lab0.uvpr	oi - uVision	internal li	-	Read Red		-			- 0 ×
File Edit View	Project Flash Debu	ug Peripheral	s Tools SVCS Wind	low Help									
🗋 😂 🖬 🥔	X 16 😤   9 0	-	P 作作作 [ ] 第	運 //: //;	SysTick_Config	💌 🗟 🥔	۹ 🜔	o 🔗 🍓 🔳 🔦					
👫   🗄 🛞	♦   0° ⊕ ¶	DQ 🖬	= 💽 - 🖾 🧐 =	🤉 - 🔳 -	💷 • 📓 • 🔆 •								
Registers	÷ 🗎	Disassembly											# 🔛
Register	Value	58:	AGAIN	1	LDR R1, =0x500	00000		;Write to LED with value	e 0x55				*
Core		59:	1080 4906 LL	IK I	r1,[pc,#24] ; @UX LDR	R0. =0x5	5						
R0	0x0000000	0x00000	082 4807 LE	R	r0,[pc,#28] ; @0x	00000000							
R2	0x00000000	60:			STR	R0, [R1]							*
R3	0x0000000												,
85	0x0000000	cm0	ldsasm.s										<del>▼</del> ×
RG	0x0000000	50			DCD 0								^
R7	0x0000000	51		AREA I	.text], CODE, READ	ONLY							
H8	0x0000000	53	;Reset Handler										
R10	0x00000000	54	Reset_Handler	PROC	Baarda Maadalaa								
R11	0x0000000	56		ENTRY	Reset_nandier								
R12 (SP)	0x00000000	57											
R14 (LR)	DxFFFFFFF	58	AGAIN	LDR	R1, =0x50000000	;	Write t	D LED with value 0x55					
R15 (PC)	0×0000080	59 60		STR	RO, =0x55 RO, [R1]								
E Reeled	0x01000000	61											
E System		62											
E-Internal		63		LDR	RO =0x2FFFFF		Delav						
Mode	Thread	65	Loop	SUBS	R0,R0,#1		Deray						
States	0	66		BNE Lo	qo								
Sec	0.00000000	67		TIDE	P1 =0*5000000		Write t	TED with value 0x22					
		69		LDR	RO, =OXAA		WIICE C	DED WICH VALUE DARA					
		70		STR	R0, [R1]								=
		71		TIDD			Delew						
		72	Loopl	SUBS	R0, R0, #1	,	Deray						
		74	-	BNE Lo	op1								
		75		P ACAT	NT.								
Desired III De				B AGAI	a								-
and Project and Re	gisters	<b></b>											
Command		0.000 0.000	0-0- 0/ T/-/-				4 🖬	Memory 1					4 🖬
*** Currently	ed version with 3 v used: 172 Bytes	32/68 Byte 3 (0%)	Code Size Limit				Â	Address: 0x5000000					🖬 🍙
							-	0x50000000: 00 00 00 00 00 0	00 00 00 00	00 00 00 00 0	0 00 00 00 00 00	00 00 00 00	00 00 00 00 00 00
<			m				F.	0x5000001C: 00 00 00 00 00 0	00 00 00 00	00 00 00 00 0	00 00 00 00 00 00	00 00 00 00	00 00 00 00 00 00
>								UX50000038: 00 00 00 00 00 0	00 00 00 00	00 00 00 00 0	00 00 00 00 00 00	00 00 00 00	00 00 00 00 00 00
ASSIGN Break	Disable BreakEnab	le BreakKi	ill BreakList Bre	akSet B	reakAccess COVERAG	E DEFINE DIR		Call Stack + Locals   III Memory 1					
									Simula	tion	t1: 0.00000000 sec	L:58 C:1	CAP NUM SCRL OVR R /W
- C-	🤌 🦂 N	🔁 Sticky	👩 Using 🛐 Sky	e 🚺 li	nbox 🔼 🔎 P	👔 👔 Softwa	Downl	. 📑 Docu 🕌 PSN 🛛 🐨 Gettin.	😨 C:\Use	Code.h	disas	100%) @	∽ 🏡 afi 10:14

- 6. Execute the image using Single Step and watch the memory contents change
- 7. Close the simulation using Debug  $\rightarrow$  Close simuation



ARM University Program Copyright © ARM Ltd 2013