

1

NGX parallel port JTAG



How to Install H-JTAG:

User can download the latest version (the document is in reference with the H-JTAF V0.9.2) installation file from www.hjtag.com. Normally, the downloaded file is zipped and need to be unzipped first. After unzipped, user can get the installation exe file h-jtag.exe. Correct setting is done by referring the User manual that is given in installation path (C:\Program Files\H-JTAG\DOC). Select lpc2148 as target for blue board.

How to install KEIL:

User can download the latest version (the document is in reference with the mdk380a) installation file from <u>http://www.keil.com</u> and install it.

Configuration of H-JTAG in KEIL:

This section introduces how to configure the keil to work with the H-JTAG. The H-JTAG V0.9.2 uses the AGDI interface under keil.

First, run TOOLCONF.EXE located under the installation direction of H-JTAG or the icon which you can see on the desktop as in figure 1.



Figure 1

The purpose is to update the configuration file (TOOS.INI) of KEIL. In TOOLCONF.EXE, use the browse button to locate the configuration file TOOLS.INI, which can be found under the installation direction of KEIL, as shown in below. Then click the config button to perform the update as in figure 2. After the update is completed successfully, click Exit to exit TOOLCONF.EXE.

ToolConf for KEIL		
KEIL TOOLS.INI		
C:\Kei\TOOLS.INI		—
	Config	Exit

Figure 2



Next start KEIL and open a project. Then click Project \rightarrow options for target as shown in figure 3

🕎 buzzer - µVisi	on3 - [E:\Ravi\Keil_Related_Work\Buzzer\Buzzer\main	.c] 🔲 🗖 🔀
Eile Edit View	Project Debug Flash Peripherals Tools SVCS Window Help	_ & ×
 ▲ ← → (2) ▲ ▲ ● ● Project Workspace 	New µýlsion Project New Project <u>W</u> orkspace Import µVision1 Project Open Project Glose Project	_ شم باط files
Target 1	Manage Select Device for Target 'Target 1' Ramove Item Octions for Target 'Target 1'	Function*******
	Clean target Build target Rebuild all target files	×
<pre>x compiling m linking Program Siz FromELF: cr y buzzer.axf</pre>	Batch Build Translate E:{RavilKell_Related_Work}Buzzer/Buzzer/main.c Stop build	6
Set Project options	1 E:\Ravi(Keil_Related_Work)Buzzer/Buzzer/buzzer.Uv2 2 C:\Keil/ARM(Boards)WRP(S)A2510(Bilnky/bilnky.uv2 3 C:\Keil/ARM(Boards)WRP(LH79524)Bilnky/bilnky.uv2	er . AXF"

Figure 3

Then a dialog of options will appear as in figure 4.

ptions for	Targe	t 'Target 1	(
Device Ta	arget 0	utput Listing	User C/C	++ Asm	Linker	Debug I	Jtilities		
NXP (found	led by Pl	nilips) LPC214	8						
			∐tal (MHz):	2.0	ABM	-Mode	•		
Operating	g system:	None		•	L Di	se Cross-N	1odule Optimiza	tion	
-		,			L Di	se MicroLl	B ľ	Bjg Endian	
- Read/Or default o	nly Memo off-chip ROM1:	ory Areas Start	Size	Startup	F U: Read/ default	se Link-Tir Write Men off-chip RAM1:	me Code Gener nory Areas Start	ation Size	Nolnit
	ROM2:			C	Г	RAM2:			
	ROM3:			0	Г	RAM3:			
	on-chip ROM1: ROM2:	0x0	0x80000	- e - c	<u>ح</u> ا	on-chip IRAM1: IRAM2:	0x40000000	0×8000	
		,	1	-1	1		1		-11
			ОК	Car	ncel	Defa	ults		Help

Figure 4.



Active the device page in the option dialog. Select the target processor which you are using (here in this case select LPC2148) as in the figure 5.

Options for Target 'Target 1 Device Target Dutput Listing Database: Generic CP Vendor: NXP (founded by Philip Device: LPC2148 Toolset: ARM	User C/C++ Asm Linker Debug Utilities J Data Base s)	
C LPC2131/01 LPC2132 LPC2132 LPC2132 LPC2132 LPC2134 LPC2134/01 LPC2136 LPC2136 LPC2138 LPC2138 LPC2138 LPC2138 LPC2142 LPC2142 LPC2144 LPC2144	ARM7TDMI-S based high-performance 32-bit RISC Microcontroll 512KB on-chip Flash RUM with In-System Programming (ISP) and Two 10bit ADCs with 14 channels, USB 2.0 Full Speed Device Two UARTs, one with full modem interface. Two 12C serial interfaces, Two SPI serial interfaces Two 32-bit timers, Watchdog Timer, PWM unit, Real Time Clock with optional battery backup, Brown out detect circuit General purpose I/O pins. CPU clock up to 60 MHz, On-chip crystal oscillator and On-chip F	er with Thu All All All All All All All All All Al
	OK Cancel Defaults	Help

Figure 5.

Active the debug page in option dialog. In the Debug page, click the drop-down button, two debug drivers, H-JTAG ARM and H-JTAG CORTEX-M3, can be found on the list, as in figure 6. User should select one of the drivers' accordingly. Here user should select H-JTAG ARM.

				o
Limit Spee	d to Real-Time		AG AHM IK ARM Debugger	Settings
 Load Appli Initialization Fil 	cation at Startup 🔽 Run to main() e:	Initializatic Lumi	Interface Driver a Blaster Cortex Debugge inary Eval Board um Systems JTAGjet	er o main()
Restore Deb	ug Session Settings oints IV Toolbox points & PA y Display	Restore Corts	ex-M3 J-LINK ink Debugger AG ARM AG CORTEX-M3 opints Display	
CPÚ DLL:	Parameter:	Driver DLL:	Parameter:	
SARM.DLL	-cLPC2100	SARM.DLL		
Dialog DLL:	Parameter:	Dialog DLL:	Parameter:	
DARMP.DLL	-pLPC2148	TARMP.DLL	-pLPC2148	

Figure 6.



This should look as in the figure 7.

Options for Ta	rget 'Target 1'		
Device Target	Output Listing User C/C++ Asm	Linker Debug	Utilities
C Use <u>S</u> imula □ Limit Speed	tor Settings I to Real-Time	i	AG ARM
✓ Load Applic Initialization File	cation at Startup 🔽 Run to main()	Load Applic	ation at Startup 🔽 Run to main()
	Edit		Edit
Restore Debu Breakpo Watchp Memory	ug Session Settings pints I⊽ Toolbox noints & PA Display	Restore Debu Breakpo Watchpo Memory	ig Session Settings ints I⊄ Toolbox oints Display
CPU DLL:	Parameter:	Driver DLL:	Parameter:
SARM.DLL	-cLPC2100	SARM.DLL	
Dialog DLL:	Parameter:	Dialog DLL:	Parameter:
DARMP.DLL	J-pLPC2148	TARMP.DLL	PLPC2148
	ОК Са	ancel De	sfaults Help

Figure 7.

Active the utilities in option dialog. In the utilities page click the drop down button, two debug drivers, H-JTAG ARM and H-JTAG CORTEX-M3, can be found on the list, as in the figure 8. User should select one of the drivers' accordingly. Here user should select H-JTAG ARM.

	H-JTAG ARM	✓ Settings	I Update Target before Debugging
Init File:	ULINK ARM Debugger ULINK Cortex Debugger		Edit
C Use Exter	Luminary Eval Board		
Command	J-LINK / J-TRACE		
Arguments:	ST-Link Debugger		
	HJTAG CORTEX-M3		

Figure 8



This should look as in figure 9.

Configure Flash C Use Targe Init File: Use Exterr Command: Arguments:	Menu Command t Driver for Flash Programming H-JTAG ARM ULINK ARM Debugger ULINK Cortex Debugger RDI Interface Driver Luminary Eval Board Signum Systems JTAGjet J-LINK / J-TRACE Cortex-M3 J-LINK ST-Link Debugger H-JTAG ARM X-M3 H-JTAG CORTEX-M3	Settings VUpdate Target before Debugging	
	ОК	Cancel Defaults Help	



Then, click OK to complete the configuration.

Then click on the debug \rightarrow start/stop debug session as shown in figure 10.

🕎 buzzer - µVision3 - [C:W	EILVARM\INC\PHILIPS\LPC2	14X.H]		
Eile Edit Yiew Project Det	oug Flash Peripherals Iools S	VCS <u>W</u> indow	_ Help	_ & ×
🎦 😂 🖬 🎒 🐰 🖻	Start/Stop Debug Session	Ctrl+F5	· · · ·	
← → \1 ♣ @	. <u>R</u> un	F5		
🕸 🖭 🕮 🗶 😫 Pi	Step	F11	E 👷	
Project Workspace	Step Over	F10	(*((volatile unsigned char *)	OXEODSCO-
🖃 🔂 Target 1	Step Out of current Function	Ctrl+F11	(*((volatile unsigned char *)	OxE005C0
E Source Group 1	} Run to ⊆ursor line	Ctrl+F10	(*((volatile unsigned char *)	OxE005C0
⊡ 🛣 startup.s 🛛	E Stop Running		(*((volatile unsigned char *) (*((volatile unsigned short*)	OxE005C0
🖵 📄 lpc214x.ł			(*((volatile unsigned short*)	OxEOO5CO
بت ا	Breakpoints	Ctrl+B	(*((volatile unsigned char *)	OxE005C0
	Insert/Remove Breakpoint	F9	eral Interface 0) */	
The second se	Enable/Disable Breakpoint	Ctrl+F9	(*((volatile unsigned short*)	0xE00200 -
	Disable <u>A</u> ll Breakpoints			
	Kill All Breakpoints Ctrl-	+Shift+F9	C214X	
× assembling Startı 🖒	Show Next Statement			-
compiling main.c.				
3 Program Size: Coc	Debug Settings		ta=1256	
FromELF: creating	Enable/Disable Trace Recording			
S Duzzer.axi" - 0	View Trace Records	Ctrl+T		- I. I. I. I.
	Execution Profiling	•		
Start/Stop Debug Session	11		H-JTAG ARM	

Figure 10.



Then a dialog of H-Flasher will appear as shown in figure 11

e	LPC2148	
	Program or Skip ?	Size = 832 B



Click on the program tab to program the target board. The cursor will point to the main as in figure 12.

🕎 buzzer 🕞 µVision3 - [E: RaviVkeil_Related_Work\Buzzer\Buzzer\main.c]	
Ele Edit View Project Debug Flash Peripherals Iools SVCS Window Help	_ & ×
1111日間 2111年年10%% 10% 1111日 111111	
않 🗉 🚳 🖓 🖓 🔶 앞 않 🙊 🖉 😸 🔳 🗄 😨 🕫 🔈 🗡	
Project Workspace • × 19 (
Berixter Val A 20 // for loop variable declaration	
E Current 21 int j;	
R0 0x4 22 // P1.25 output	
$R1 0_{x4}$ $rac{1}{2}$ IODIR1 = 0x02000000;	
R2 0x4 24 // endless loop to toggle the Buzzer Beep	
R3 0x4 25 while (1)	
R4 0x0 26 (
R5 0x4 27 #if 1	-1
for (j = 0; j < 1000000; j++)	
	<u> </u>
Sumbole E Startup.s E main E LPC214X R Disassembly	
X AGDI: \\buzzer\main A X Name X Add A X Stack Frames X Fibe	er: Execution All
AGDI: \\buzzer\Star	er: Executionsai
AGDI: \\buzzer\Star	me # Addres 🔔
S ASSIGN	
	-
For Help, press F1 H-JTAG ARM	t1: 0.0000000

We are now all set to debug the program with NGX parallel port JTAG module.