

Telink IoT Studio User guide

AN-22063002-E1

Ver1.0.10 2025.02.21

Keyword

Telink, IoT Studio

Brief

This document is to guide how to use Telink IoT Studio.



Published by Telink Semiconductor

Bldg 3, 1500 Zuchongzhi Rd, Zhangjiang Hi-Tech Park, Shanghai, China

© Telink Semiconductor All Rights Reserved

Legal Disclaimer

This document is provided as-is. Telink Semiconductor reserves the right to make improvements without further notice to this document or any products herein. This document may contain technical inaccuracies or typographical errors. Telink Semiconductor disclaims any and all liability for any errors, inaccuracies or incompleteness contained herein.

Copyright © 2023 Telink Semiconductor (Shanghai) Co., Ltd.

Information

For further information on the technology, product and business term, please contact Telink Semiconductor Company www.telink-semi.com

For sales or technical support, please send email to the address of:

telinksales@telink-semi.com

telinksupport@telink-semi.com



Version

Revision	Date	Author	Change log
V1.0.0	2022-06	Xiongjun.He	First release
V1.0.1	2022-07	Xiongjun.He	Add more known issues
			Add IDE setting chapter
V1.0.2	2022-09	Xiongjun.He	Select version with chip type
V1.0.3	2022-11	H.X.J,W.F	Add Jtag debugging section
V1.0.4	2023-03	H.X.J,W.F	Telink IDE is renamed to Telink IoT Studio
			Add usage of some tools
V1.0.5	2023-08	H.X.J,W.F	Update toolchain to V5.1.2
			Add description about building V3.2.3 and V5.1.2 projects/
			configurations simultaneously
			Add installation and uninstallation in linux
V1.0.6	2024-03	H.X.J,W.F	Add usage of Smart Build Button and Telink ICEman GDB
			Debugging
			Add explanation for andes doc and different versions of
			ICEman
V1.0.7	2024-04	H.X.J	Add usage of Telink Formatter
V1.0.8	2024-06	H.X.J,W.F	Add usage of Telink IoTStudio to CMake and update Telink menu pictures
			Delete all content related to build tool
V1.0.9	2024-09	H.X.J,W.F	Modify the description of toolchains and ICEman Add
		·	instructions on how to compile in the command line Add
			methods to enable Windows OS case sensitivity
V1.0.10	2025-02	H.X.J,W.F	Modify the usage of TelinkFormatter Add the usage of SVD
			Support

Contents

Ver	sion		3
1	Des	ktop shortcuts	8
2	Diff	erent IDEs	8
3	loT	Studio or integrated tools	9
4	Ргој	ject property difference between Telink RDS and Telink IoT studio	9
	4.1	Ext HW DSP option	9
	4.2	-msave-restore and -msmall-data-limit	10
5	Tool	Ichain	11
•	5 1	Toolchain information	11
	5.1	5.1.1 Toolchain location	11
	52	Select a different toolchain for TLSR9 projects	12
	5.2	5.2.1 Advanced usage: toolchain setting according to path	12
6	Imo	ort and build projects	14
Ŭ	6.1		1 <u>4</u>
	6.2		15
7	Tolir	nk menu and toolbar entries	15
•	7 1	RDS to loT Studio converter	15
	,	7.1. Using converter as a standalone utility	18
		712 Notice	19
	72		19
	73		20
	7.0	7 3.1 Parameters	20
		7.3.2 Use ICEMan in Itan Burn dialon	21
		7.3.3 ICEMan in Jtag Burn ELF setting	21
		7.3.4 Burn Program	21
		7.3.5 Security Download	22
	7.4	Telink links	23
	7.5	libusb version BDT	24
	7.6	Toolchain shell or ICEMan shell menu entry	25
	7.7	Windows BDT	26
	7.8	Open artifact path	26
		7.8.1 Notice	27
	7.9	Copy artifact path	27
		7.9.1 Notice	27
		7.9.2 In other places	27
	7.10	Search on DocSite	28
8	Соп	nmon compiling error	29
	8.1	Can't find the link script file	29
9	Oth	er plugins or functions	31
	9.1	Telink Formatter	31
		9.1.1 Use as formatter	31
		9.1.2 Trigger with saving event(recommanded method)	34
		9.1.3 Known issue	35
		9.1.3.1 Save button is disabled at the first time	35

	9.2	Easy Shell	35
	9.3	ECalculator	35
	9.4	Terminal in Eclipse	36
	9.5	Binary Viewer	37
10	FAQ		38
	10.1	Error 127 when compiling the code	38
	10.2	Not found make in PATH or nothing to build when building	38
	10.3	Orphaned configuration and no options on Settings tab	38
	10.4	How to verify the current used toolchain gcc version for TLSR9	39
	10.5	How to verify current used make tools	41
	10.6	The permission error occurred while opening the Telink TC32 console	43
	10.7	' How to enable case sensitivity feature on Windows OS	43
11	Kno		44
	11.1	Save button is disabled at the first time	44
	11.2	IoT Studio exits if users click the Open artifact path multiple times	44
	11.3	IoT Studio will block for a while after importing project and changing toolchain	44
	11.4	The size of generated artifacts are zero	44
	11.5	Clicking the telink menu is invalid	46
	11.6	The problem that the source file cannot be found in debugging $\ldots \ldots \ldots \ldots \ldots \ldots \ldots$	46
12	IoT S	Studio settings	46
	12.1	Hide print margin	46
13	IoT S	Studio Debug Tool For TLSR9	48
	13.1	Build debuggable programs	48
	13.2	Burn program	50
	13.3	Telink ICEman GDB Debugging (Recommended)	51
	13.4	View peripheral registers through SVD file	53
	13.5	Breakpoint	55
14	Inde	ependent use of some tools in the IoT Studio	55
	14.1		55
	14.2		56
15			56
	15.1		50
16	Dec	s for AndeSight	57
10		r to compile lotStudio projects in the command line	50
17	HUW		30



List of Figures

Figure 2.1	Telink IoT studio Icon	8
Figure 2.2	Telink Old IDE for TLSR8 Icon	8
Figure 2.3	Telink RDS IDE for TLSR9 Icon	8
Figure 4.1	DSP option in Telink IoT studio	9
Figure 4.2	Add DSP libraries in Telink IoT studio	10
Figure 4.3	RISC-V specific option on Telink IoT studio	11
Figure 5.1	Select Toolchain	12
Figure 5.2	Toolchain setting	13
Figure 5.3	Set Toolchain path	13
Figure 6.1	Import	14
Figure 7.1	Telink Menu	15
Figure 7.2	Linux Select cproject	16
Figure 7.3	Telink project converter	17
Figure 7.4	Converter result	18
Figure 7.5	Telink loTStuio to Cmake	19
Figure 7.6	Telink loStudio to Cmake UI \ldots	20
Figure 7.7	Jtag burn dialog	21
Figure 7.8	Jtag burn	22
Figure 7.9	Jtag Burn Encrypt	23
Figure 7.10	Telink Links	24
Figure 7.11	Telink tools launcher	24
Figure 7.12	Open libUSB BDT with artifact Path	25
Figure 7.13	Open libUSB BDT without artifact Path	25
Figure 7.14	Shells	26
Figure 7.15	Open artifact path	26
Figure 7.16	Copy artifact path	27
Figure 7.17	Copy artifact path on Toolbar	27
Figure 7.18	Copy artifact path on pop-up menu	28
Figure 7.19	keyword	28
Figure 7.20	doc site	29
Figure 8.1	Link error	29
Figure 8.2	Link file path	30
Figure 8.3	Link file select	31
Figure 9.1	Formatter setting	32
Figure 9.2	TelinkFormatter setting	33
Figure 9.3	Formatter hint	33
Figure 9.4	Formatter console output after formmtting	34
Figure 9.5	Telink Formatter Preference	34
Figure 9.6	Formatter console output after saving	35
Figure 9.7	Easy Shell	35
Figure 9.8	ECalculator	36
Figure 9.9	Terminal Selection	36
Figure 9.10	Terminal for Serial	37
Figure 9.10	Rinary Viewer	יב, דר
. igoic 2.11		57

Figure 10.1	Error127	38
Figure 10.2	Orphaned configuration	39
Figure 10.3	GCC version setting in Post command	40
Figure 10.4	Verify the GCC version	41
Figure 10.5	Make tool setting in Post command	42
Figure 10.6	Verify the make tool path \ldots	42
Figure 10.7	Permission problem	43
Figure 10.8	Enable WSL	43
Figure 11.1	Artifact size is 0	45
Figure 11.2	loT Studio is busy	45
Figure 11.3	locate	46
Figure 12.1	Print margin in text editor	47
Figure 12.2	Configure the print margin	47
Figure 13.1	Jtag Burn err	48
Figure 13.2	syntax err	49
Figure 13.3	delete option	49
Figure 13.4	delete_mabi	50
Figure 13.5	Jtag burn	50
Figure 13.6	Select ELF	52
Figure 13.7	debug	53
Figure 13.8	ICEman interface	53
Figure 13.9	SVD Tab	54
Figure 13.10	View Peripheral	54
Figure 14.1	converter cmd	55
Figure 14.2	ICEman defalut port	56
Figure 15.1	Installation	57
Figure 15.2	Post of installation	57
Figure 15.3	Uninstall	58
Figure 17.1	select toolchain shell	59
Figure 17.2	make in shell $($	59



1 Desktop shortcuts

The Telink IoT studio installer will create 2 shortcuts on the desktop:

- Telink IoT studio : used to launch the IoT Studio.
- Telink TC32 console : used to launch the development console for TLSR8, users can input commands like tc32-elf-gcc and make in it.

2 Different IDEs

The following 3 different IDEs will be referred to in this user guide:

- Telink IoT studio : the IDE contains this document.
- Telink old IDE : the IDE list on this wiki page (IDE for TLSR8 Chips) for TLSR8 chips.
- Telink RDS IDE : the IDE list on this wiki page (IDE for TLSR9 Chips) for TLSR9 chips.

The above 3 IDEs have the following different desktop shortcut icons:

Telink IoT studio:



Figure 2.1: Telink IoT studio Icon

Telink old IDE:



Figure 2.2: Telink Old IDE for TLSR8 Icon

Telink RDS IDE:



Figure 2.3: Telink RDS IDE for TLSR9 Icon

3 IoT Studio or integrated tools

- Build TLSR8 Projects and SDK
- Import and build TLSR9 Telink RDS Projects and SDK
- Telink Windows BDT used to flash image for both TLSR8 and TLSR9
- Telink libUsb version BDT used to flash image for both TLSR8 and TLSR9, still in beta phase.
- Jtag_Burn used to flash image for TLSR9 with JTAG

4 Project property difference between Telink RDS and Telink IoT studio

4.1 Ext HW DSP option

This option is used to configure whether to use the DSP and its libraries:

Properties for B93_Driver_Demo

type filter text	Settings
 Resource Builders C/C++ Build Build Variables Environment Logging 	Configuration: D25_Enable_N22_Demo [Active] Tool Settings Toolchains Devices Container Settings Build Steps
Settings Tool Chain Editor > C/C++ General Git Linux Tools Path Project Natures Project References RISC-V Toolchains Paths Run/Debug Settings Task Tags TC32 Loader > Validation WikiText	 Target Processor Optimization Warnings Debugging Sove temporary files (save-temps Use with cau Verbose (-v) Other compiler flags -mext-dsp -mabi=ilp32f -c Support ANSI Programs, same as RDS (-ansi) Code Coverage, same as RDS (-coverage) Apply HW DSP options(-mext-dsp) Miscellaneous Source Coverage Optimization Warnings Includes Optimization Warnings Miscellaneous Miscellaneous

Figure 4.1: DSP option in Telink IoT studio

This option would add -mext-dsp flag to cc and ld , and also it would add -lm and -ldsp flags to linker in Telink RDS IDE. While the Telink IoT studio would only add -mext-dsp flag to cc and ld without the -lm and -ldsp extra libraries flags. This makes the configuration more flexible.

Users can add the math and dsp libraries to linker specifically on the linker library setting page:

Telink



Properties for B93_Driver_Demo

Figure 4.2: Add DSP libraries in Telink IoT studio

4.2 -msave-restore and -msmall-data-limit

TheTelink RDS IDEuses the IoT Studio built-in (default) value for both-msave-restoreand-msmall-data-limitwithout options on configuration page, while theTelink IoT studiohas a moreflexible configuration for these options.

Right now, users should use the Use toolchain default value option for TLSR9 chips.



Figure 4.3: RISC-V specific option on Telink IoT studio

5 Toolchain

5.1 Toolchain information

The Telink IoT studio contains several toolchains for TLSR8 and TLSR9:

- Telink TC32 Cross Compiler with GCC version 4.5.1 (tc32-elf) for TLSR8 which is the same as the Telink old IDE.
- Cross Compiler with GCC version 7.4 from Telink RDS IDE V3.2.3 for TLSR9 (D25F and N22 architecture) chips
- Cross Compiler with GCC version 10.3 from Telink RDS IDE V5.1.2 for TLSR9 (D25F and N22 architecture) chips
- Cross Compiler with GCC version 12.2 from Telink RDS IDE V5.3.x for TLSR9 (D25F and N22 architecture) chips

5.1.1 Toolchain location

We assume that the Telink IoT studio is installed at location \$IoTStudio :

- Telink TC32 Cross Compiler with GCC version 4.5.1 (tc32-elf-1.5) for TLSR8: \$IoTStudio/opt/tc32/bin
- Cross Compiler with GCC version 7.4 from Telink RDS IDE V3.2.3 for TLSR9 (D25F and N22 architecture) chips
- N22: \$IoTStudio/RDS/V3.2.3/toolchains/nds32le-elf-mculib-v5
- D25F: \$IoTStudio/RDS/V3.2.3/toolchains/nds32le-elf-mculib-v5f
- Cross Compiler with GCC version 10.3 from Telink RDS IDE V5.1.2 for TLSR9 (D25F and N22 architecture) chips
- N22: \$IoTStudio/RDS/V5.1.2/toolchains/nds32le-elf-mculib-v5
- D25F: \$IoTStudio/RDS/V5.1.2/toolchains/nds32le-elf-mculib-v5f
- Cross Compiler with GCC version 12.2 from Telink RDS IDE V5.3.x for TLSR9 (D25F and N22 architecture) chips



- N22: \$IoTStudio/RDS/V5.3.x/toolchains/nds32le-elf-mculib-v5
- D25F: \$IoTStudio/RDS/V5.3.x/toolchains/nds32le-elf-mculib-v5f

5.2 Select a different toolchain for TLSR9 projects

As only one toolchain is included for TLSR8, this section is only applied to TLSR9 chips.

To change the toolchain of a project, open the project properties, navigate to C/C++ Build -> Settings -> Toolchains tab, and select the different one from the drop-down list:

Settings		Ç • ⊂) •
Configuration: UA	RT_Demo [Active] / 1	Manage Configurations
Tool Settings	🛞 Toolchains 📕 Devices 📳 Devices 🕕 Container Settings 🎤 Build Steps 🈤 Build Artifact 🔛 Binary Parsers 🥝 Error Parsers	
Name:	TL32 ELF MCULIB VSF-GCC122 (risco32-elf-gco)	~
Architecture:	T122 EF MCULB V5F GC122 (mov2z-ef-gec) T123 EF MCULB V5F GC122 (mov2z-ef-gec)	
Prefix:	T122 EF MCUIB VFF GOCID3 (mixe32=hif-spec)	
Suffix:	Lizz ED WIGOLB VI POLCA (MIGOZA) Higgel Telink TLSB VESF GCZ / (MIGOZA) Higgel	
C compiler:	Telink TLSR 025F GCC 10 (riscv2a-elf-spc) Deak GNU 18/GVC Humbddde DEC (riscv-none-embed-gcc)	
C++ compiler:	RISC-V CCC/New/B (risco4-unknown-Hirg.cc) RISC-V CCC/New/B risco4-unknown-Hirg.cc)	
Archiver:	RISC-V GC/RTEMS (rikes4-unknown-tems-gcc)	
Hex/Bin converter:		
Listing generator:	bbjdump	
Size command:		
Bemove command:		
Toolchain path:	\$(ecilipse_home)/RD5/V5.3.x/toolchains/nds32le-elf-mculib-v5f/bin	
	(to change it use the <u>global</u> or <u>workspace</u> preferences pages or the <u>project</u> properties page)	
Build tools path:	Stellipse home//RDS/V/S1.2/c/gwin/bin	
	(to change it use the <u>diobal</u> or <u>workspace</u> preferences pages or the <u>project</u> properties page)	
🗹 Create flash ima	age	
Create extended	d lising	
Print size		



The listed toolchains are:

- TL32 ELF MCULIB V5F GCC12.2 : for D25F architecture chips, from RDS V5.3.x
- TL32 ELF MCULIB V5 GCC12.2 : for N22 architecture chips, from RDS V5.3.x
- TL32 ELF MCULIB V5F GCC10.3 : for D25F architecture chips, from RDS V5.1.2
- Telink TLSR9 D25F GCC 10 : same as TL32 ELF MCULIB V5F GCC10.3
- TL32 ELF MCULIB V5F GCC7.4 : for D25F architecture chips, from RDS V3.2.3
- Telink TLSR9 D25F GCC 7 : same as TL32 ELF MCULIB V5F GCC7.4

5.2.1 Advanced usage: toolchain setting according to path

Normally, users shouldn't set the toolchain by changing toolchain path. Instead, a toolchain should be set with the method mentioned in the previous section: Select a different toolchain for TLSR9 projects

This section is for advanced users with special requirements.

Open the project property dialog by clicking the Property menu item on right and click the pop-up menu on the project:



Switch to the C/C++ Build -> Settings -> Toolchains tab:

Probetnes for pag_priver_p	remu I
type filter text	Settings
 Resource Builders C/C++ Build Build Variables Environment Settings Tool Chain Editor C/C++ General Git Linux Tools Path MCU Project Natures Project References Run/Debug Settings Task Tags TG2 Loader TeinkFormatter Validation 	Configuration: UART_Demo [Active]
	Tool Settings Tool Settings Tool Settings Tool Settings Tool Settings Telink TLSRP D25F GCC 10 (riscv32-elf-gcc) Archivecture: RiSC-V Prefix: riscv32-elf- Suffix: C compiler: gcc C++ compiler: g++ Archiver: ar Hex/Bin converter: ebjcopy Listing generator: objdump
Wiki lext	Size command: size Build command: make Remove command: m Toolchain path: \$(eclipse home)/RDS/V5.1.2/toolchains/nds32le-elf-mculib-v5f/bin
	(to change it use the global or workspace preferences pages or the project properties page) Build tools path: Steclipse_home)/RDS/V5.12/cygwin/bin (to change it use the global or workspace preferences pages or the project properties page)
	 ✓ Create flash image ✓ Create extended listing ✓ Print size
	Figure 5.2: Toolchain setting

Click the global link text which is surrounded in red square block.

T Preferences				×
type filter text	Global RISC-V Toolch	hains Paths	<- ▼ <>	₩ 00
✓ MCU A Global Arm Toolcha	Configure the locatior redefined more speci	ns where various GNU RISC-V toolchains are installed. The values are stored within ifically, they are used for all projects in all workspaces.	Eclipse. Unle	ss
Global Build Tools Global OpenOCD P	Default toolchain:	- Global default		~
Global pyOCD Path	Toolchain name: Tel	link TLSR9 D25F GCC 10		
Global QEMU Path	Toolchain folder: \${	eclipse_home}/RDS/V <u>5.1.2/</u> toolchains/nds32le-elf-mculib-v <u>5f/</u> bin Browse	xPack	c



Normally, users would only be required to change the version and the suffix which are marked with red lines.

The first red line marks the toolchain base version string (in the image it is **V5.1.2**) of the toolchain, users can change it to one of the following items:

- V3.2.3 : from Telink RDS IDE V3.2.3
- V5.1.2 : from Telink RDS IDE V5.1.2
- V5.3.x : from Telink RDS IDE V5.3.x

The second red line indicates the version of the toolchain variants string (in the image it is nds32le-elf-mculib-v5f) of the toolchain, users can change it to one of the following items:

Set the path string in the line edit.

- nds32le-elf-mculib-v5f : for Telink TLSR9 D25F chips
- nds32le-elf-mculib-v5 : for Telink TLSR9 N22 chips

Once the toolchain path variable has been changed, click the **apply and close** button to save it.

6 Import and build projects

6.1 TLSR8

Telink IoT studio supports SDK or projects import function like the Telink old IDE.

```
Users can use the File -> Import... to import SDK or projects:
```

T Import	
Select Create new projects from an archive file or directory.	Ľ
<u>S</u> elect an import wizard:	
type filter text	
 ✓ Seeneral Image: Archive File Image: Existing Projects into Workspace Image: File System Image: Projects from Folder or Archive Image: Projects from Folder or Archive Image: C/C++ Image: C/C++ Image: C/C++ Image: Project from Folder or Archive Image: Pr	
< <u>B</u> ack <u>N</u> ext > <u>Finish</u>	Cancel
org.eclipse.embedcat 2022/6/21 N+12:09	

Figure 6.1: Import

Users can refer to this Telink wiki page.

6.2 TLSR9

For Telink RDS IDE format projects(for example, the SDKs for TLSR9 chips at Telink official wiki page), it must be converted to Telink IoT studio format before importing it into the workspace.

Users can refer to the following RDS to IDE converter section on Telink menu chapter to view how to do it.

7 Telink menu and toolbar entries

7.1 RDS to IoT Studio converter

If you get the SDK or projects for TLSR9 chipB91from Telink official wiki page, they are inTelink RDS IDEformat, you can use this function to convert it toTelink IoT studioformat before importing it into TelinkIoT studio, and SDK or projects ofB92do not require conversion.



Figure 7.1: Telink Menu

Click the Select .cproject file button in the dialog to choose the Telink RDS IDE format .cproject file.

In Linux OS, users can use Ctrl + h or other shortcut keys to show the hide files (.cproject), and the filter should be changed to * :

	Telink Project converter: RDS to IDE				
Converter setting					
Converter util location: //home/hexiongjun/Projects/CDT_Telink_IDE/eclipse/tools/Converter/TelinkRDS2IDE Select TelinkRDS2IDE exe/ELF					
RDS .cproject					
.cproject file location: Click the select button	on right to select .cproject file>>>	Select .cproject file			
Verbose Log Convert					
Log and info					
2022-07-11 11:17:12 CST					
Platform Info: Linux amd64					
Usage Steps: 1. Select the RDS format '.cproject' by clicking t 2. Click the 'convert' button to do the convert to 3. Click the 'OK' button to import the project	he 'Select a file' button DE format				
Cancel		Q Open			
🔿 Recent	Image: SDKs telink_b91m_driver_sdk-2 project tlsr_riscv BS)1 ▶			
🔂 Home	Name S	ize Modified 🔻			
🖿 Desktop	.project 7	.6 kB 27 Jan			
Documents	.cproject 6	27.4 kB 27 Jan			
🕹 Downloads					
J Music					
Pictures					
► Videos					
🖿 t rds2ide					
	*	•			
?	C	ancel OK			

Figure 7.2: Linux Select cproject

If a **.cproject** file has been selected, it will display the file path:

Telink

т

Telink

Telink Project converter: RDS to IDE	×
Converter setting	
Converter util location: C:\IDERelease\20220620_V1.0\eclipse\tools\Converter\TelinkRDS2IDE.exe	Select TelinkRDS2IDE exe/ELF
RDS .cproject	
.cproject file location: C:\Users\admin\Desktop\telink_b91m_driver_src\project\tlsr_riscv\B92\.cproje	ct Select .cproject file
Verbose Log Convert	
Log and info	
2022-06-21 11:27:26 CST	
Platform Info: Windows 10 amd64	
Usage Steps: 1. Select the RDS format '.cproject' by clicking the 'Select a file' button 2. Click the 'convert' button to do the convert to IDE format 3. Click the 'OK' button to import the project	^
<	>
?	OK Cancel

Figure 7.3: Telink project converter

Then click the **convert** button, the result would show on the large text block:

🔨 🛛 Telink

Telink Project converter: RDS to IDE	×
Converter setting	
Converter util location: C:\IDERelease\20220620_V1.0\eclipse\tools\Converter\TelinkRDS2IDE.exe Select TelinkRDS2IDE exe	e/ELF
RDS .cproject	
.cproject file location: C:\Projects\CDT_IDE_dev\telink_b91m_driver_src\project\tlsr_riscv\B92\Backup_before Select .cproject	t file
Verbose Log Convert	
Log and info	
2022-06-21 17:47:11 CST	
Platform Info: Windows 10 amd64	
[2022/06/21T17:47:43] [NOTICE] CProject file is Telink RDS IDE format, continue [2022/06/21T17:47:43] [NOTICE] Configuration list from original project file, count:31	^
0:UART Demo 1:Debug Demo 2:TIMER Demo	
3:PWM_Demo 4:SPI_Demo 5:Flash_Demo	
6:s7816_Demo 7:ALG_REG_Demo 8:GPIO_Demo	
9:I2C_Demo 10:RF_Demo 11:TRNG_Demo	
12:USB_Demo 13:STIMER_Demo 14:AUDIO_Demo	
15:VCD_Demo 16:DHRYSTONE 17:COREMARK	
18:AES_Demo 19:PM_Demo 20:LPC_Demo	
21:HTOL_Demo 22:Test_Demo 23:EMI_BQB_Demo	
24:Auto_Test_Demo 25:Make_lib 26:PLIC_Demo	
27:ADC_Demo 28:Swire_Demo 29:PKE_Demo	
10020 C Demo	
[2022/06/211174743] [NOTICE]	
[2022/06/211174743] [NOTICE] Write target project file successfully	
[2022/06/2117/47:43] [NOTICE] Write target project resolutions and the devitation of the project resolution of the contract of	orec
[2022/06/21117:47:43] [NOTICE] Origin file path: C:\Projects\CDT IDE dev\telink b91m driver src\project\tisr risc\B92karkur	o bef
[2022/06/21T17:47:43] [NOTICE] Successful backup origin file to: C:\Projects\CDT IDE dev\telink b91m driver src\project\tlsr	riscv
[2022/06/21T17:47:43] [NOTICE] Successful generated target file : C:\Projects\CDT IDE dev\telink b91m driver src\project\tlsr	riscv 🗸
<	>
OK Ca	incel

Figure 7.4: Converter result

If it is fails, the background of log text block would be set to red. Users can copy and send the log to Telink IoT studio developer/FAE to get support.

7.1.1 Using converter as a standalone utility

The converter can be used as a standalone utility, users can use it in command line.

The converter is installed at:

- Windows: \$IoTStudio_PATH/tools/Converter/TelinkRDS2IDE.exe
- Linux: \$IoTStudio_PATH/tools/Converter/TelinkRDS2IDE

To convert a .cproject file from command line in Windows OS, users can use following the command:



\$IoTStudio_PATH/tools/Converter/TelinkRDS2IDE.exe Path/To/.cproject

In Linux OS:

\$IoTStudio_PATH/tools/Converter/TelinkRDS2IDE Path/To/.cproject

7.1.2 Notice

The converted project (.cproject) would set the toolchain to Telink TLSR9 D25F GCC by default, which will use the Telink RDS IDE 7.4 GCC Cross compiler for D25F.

7.2 Telink IoTStudio to Cmake converter

If you need to convert the Telink IoTStudio project to a cmake project, you can use the tool shown in the following figure.

First, you need to double-click to select the project you want to convert, and then click Telink IoTStudio to Cmake in Telink menu.



Figure 7.5: Telink IoTStuio to Cmake

You can choose whether to generate the settings.json file required for the VS Code project. Click Convert to obtain the CMakelists.txt and settings.json files, and the log will indicate their location.





7.3 Jtag burn menu

Telink

т

The Jtag_Burn function is similar to the Telink RDS IDE Jtag_Burn function, both use the JTAG to burn the image to TLSR9X chips.

Except for the Jtag_Burn , TLSR9 can be burned with Windows BDT software through SWS protocol with Burning EVK , which is the recommended option.

The SWS protocol requires only one GPIO to flash the binary image to TLSR8 and TLSR9 chips, while the JTAG uses 2 or 4 GPIO pins for TLSR9 chips.

7.3.1 Parameters

Once the Jtag_Burn dialog has been opened, users can fill or set parameters for Jtag_Burn and ICEMan:

🔳 Telink Jtag Burn								- 0 ×
Flashing Drivers and ICEMan								
Jtag_Burn utility Path	C:\TelinkloTStudio\RDS\V3.2.3\flash\Jtag_b	urn.exe			Sele	ct Jtag_burn		~
ICEMan exe/ELF	linkloTStudio\RDS\V5.1.2\ice\ICEman.exe			Select ICEMan		MCU Architecture:	v5	~
Flashing Image/Binary and Other ICEMan arg	uments							
r_src\project\tlsr_riscv\B92\GPIO_Demo\GP	IO Demo.bin	Select image		Programming Start Address:	0x0]	
Other ICEMan Arguments:								
✓ Target management	ICEMan/SPI_burn Host:	127.0.0.1		Bunner Port:	2354			
Target Ruro								
Finder burn								Calcut Incom
Enable target burn Target burn image:								Select image
Jtag_burn Driver Arguments ICEMan Interfac Target Chip: B92 Verification Erase All	ce cfg	Flash Controller Address: Unlock	Lock after	burn	Misc Ar ☑ Reset	guments:		
L ICEMan Interface Config :		I elnet Port:	4444					
Logging and the Rup LICEMan								
>> Elash page program at address 0x000400)		^	Start the ICEMan at: 2023-08-28 10:44:4	15			<u>^</u>
>> Hash page program at address Dx000400, >> Flash page program at address Dx000400, >> Flash page program at address Dx000400 >> Flash page program at address Dx001400 >> Flash page program at address Dx001400 >> Flash verify ok at address Dx001400 >> Flash verify ok at address Dx0004000 >> Flash verify ok at address Dx0004000 >> Flash verify ok at address Dx001400 >> Flash verify ok at address Dx001400	, 0x00000000: 6916 bytes		Ň		ap Eman.exe, -Z, v5, . 2022-04-22-14:27 ap] hart 0	b, 2354])		~
<			>	<				>

Start ICEMan before burning Stop ICEMan Burn

Telink

Figure 7.7: Jtag burn dialog

Parameters for Jtag_Burn can be set in Jtag_Burn driver Arguments .

7.3.2 Use ICEMan in Jtag_Burn dialog

ICEMan is a tool to manage the TLSR9 JTAG ICE (support 4-wire and 2-wire mode).

Users can refer to the document files in the doc directory to get more information.

7.3.3 ICEMan in Jtag_Burn ELF setting

Users can set the other ICEMan and Jtag_Burn ELF from Flashing Drivers and ICEMan section, the set/ selected value will be saved and reused in next re-opening the dialog.

The normal ELF files' names are listed for reference:

- ICEMan: ICEMan.exe in Windows, and ICEMan on Linux OS
- Jtag_Burn: Jtag_Burn.exe in Windows, and Jtag_Burn on Linux OS

7.3.4 Burn Program

When burning the program, pay attention to burner's path and chip type, and then click Start ICEman. After ICEman is ready, the telnet_port and burner_port should be consistent with ICEman's, then click Burn, as shown below:



Flashing Drivers and ICEN	fan				2						
Jta	ag_Burn utility Path	C:\TelinkloTStudio\RDS\V3.2.3\flash\J	tag_burn.exe					Select Jtag_bur	n		~
	ICEMan exe/ELF	linkIoTStudio\RDS\V5.1.2\ice\ICEman	exe			Select ICEMa	n	MCL	Architecture: v5		~
Elaching Image/Ripany and	d Other ICEMan ard	umento									
r src\project\tlsr riscv\B	92\GPIO Demo\GPI	IO Demo.bin	Select image			Programming Start Address	5:	0x0			
Other ICEMan Arguments	si	-									
Target management		ICEMan/SPI_burn H	lost: 127.0.0.1				Bunner Port	t: 2354		4	
Target Burn											
Enable target burn Tar	rget burn image:										Select image
Jtag_burn Driver Argume	nts_ICEMan Interfac	ce cfg 3									
Target C	hip: B92		- Flash Contr	oller Address:	:			Misc Arguments:			
✓ Verification	Erase All		Unlock		Lock after	ourn		Reset and Run	Encrypt		
ICEMan Interface Conf	fig :		~	Telnet Port:	4444			5			
Logging: <== Jtag_Burn	ICEMan ==>	2				Chart the ICEMan at 201	22.00.20.10.44	.45			
>> Flash page program a	it address 0x000800)				Args:[C:\TelinkloTStudio\Rl	DS\V5.1.2\ice\/	CEman.exe, -Z, v5, -b, 2354]			
>> Flash page program a	t address 0x000c00)			_	Andes ICEman (OpenOCD)	5.1.1-g3a34fc9	(2022-04-22-14:27)			
>> Flash page program a >> Flash page program a	it address 0x001000 it address 0x001400)				Burner listens on 2354 Telnet port: 4444					
>> Flash page program a	t address 0x001800)				TCL port: 6666					
>> Flash verify ok at addr	ress 0x000000					Andes AICE-MINI+					
>> Flash verify ok at addr	ress 0x000400					JTAG frequency 10.000 MH	z				
>> Flash verify ok at addr	ress 0x000c00				_	The core #0 listens on 1111	ι.				
>> Flash verify ok at addr	ress 0x001000					ICEman is ready to use.					
>> Flash verify ok at addr	ress 0x001400					hardware reset-and-hold su	uccess on [nds.	tap] hart 0			
>> Flash verify ok at addr	ress 0x001800	0v00000000 6016 butor				Stopped the ICEMan a	+ 2022 00 20 1	10:49:04			
>> MCU reset and run	r success at autress	0x00000000.0910 bytes				stopped the icclivian a	1. 2025-06-26	10.48.04			
SPI_Burn exited with code	e:0										
<					>	<					>
	1	6									
Start ICEMan before burni	ng Start ICEMan Bi	urn									

Figure 7.8: Jtag burn

Jtag_ Burn is located in the \$IoTStudio_PATH/RDS/V3.2.3/flash/directory, and its help documents are also in the same directory.

7.3.5 Security Download

If you need to use the security download function of Jtag_Burn, check the encrypt checkbox

🔳 Telink Jtag Burn									o ×
Flashing Drivers and ICEMan									
Jtag_Burn utility Path	C:\TelinkloTStudio\RDS\V3.2.3\flash\Jtag_b	Jrn.exe				Select Jtag_burn			~
ICEMan exe/ELF	linkIoTStudio\RDS\V5.1.2\ice\ICEman.exe			Select ICEMan		MCU Architecture	e: v5		~
Flashing Image/Binary and Other ICEMan arg	juments						_		
r_src\project\tlsr_riscv\B92\GPIO_Demo\GP	IO_Demo.bin	Select image		Programming Start Address:	0x0				
Other ICEMan Arguments:									
Target management	ICEMan/SPI_burn Host:	127.0.0.1		Bunner Port:	2354				
Target Burn									
Enable target burn Target burn image:								s	elect image
E chable target barn rarget barn inage.									cicceriniuge
Jtag_burn Driver Arguments _ICEMan Interfac Target Chip: 892	ce cfg	 Flash Controller Address: 				Misc Arguments:]
Verification Erase All		Unlock	Lock after	burn	6	🗸 Reset and Run 🗹 Encrypt			
ICEMan Interface Config :		Telnet Port:	4444						
Logging: <== Jtag Burn ICEMan ==>									
>> Flash page program at address 0x000400	0		^	Start the ICEMan at: 2023-08-28 10:44:4	45				^
>> Flash page program at address 0x000800	D			Args:[C:\TelinkloTStudio\RDS\V5.1.2\ice\IC	Eman.exe,	-Z, v5, -b, 2354]			
>> Flash page program at address 0x000c00	D			Andes ICEman (OpenOCD) 5.1.1-g3a34fc9 (2022-04-2	22-14:27)			
>> Flash page program at address 0x001000	0			Burner listens on 2354					
>> Flash page program at address 0x001400	0			TCI port 6666					
>> Flash verify ok at address 0x001800	0			Andes AICE-MINI+					
>> Flash verify ok at address 0x000400				JTAG frequency 10.000 MHz					
>> Flash verify ok at address 0x000800				There is 1 core in tap					
>> Flash verify ok at address 0x000c00				The core #0 listens on 1111.					
>> Flash verify ok at address 0x001000				ICEman is ready to use.					
>> Flash verify ok at address 0x001400				hardware reset-and-hold success on [nds.ta	ap] hart 0				
>> Flash verify ok at address 0x001800									
>> File download to flash success at address	s 0x00000000: 6916 bytes			Stopped the ICEMan at: 2023-08-28 10	J:48:04	-			
>> MCO reset and run									
SPI_Burn exited with code.o									
			~						\sim
<			>						>

Figure 7.9: Jtag Burn Encrypt

7.4 Telink links

Start ICEMan before burning Start ICEMan Burn

Telink

т

In this sub-menu, users can quickly visit Telink websites for development resources or information:

- Telink forum
- Telink wiki
- Telink official webpage







7.5 libusb version BDT

Telink

T

The **libusb version BDT** is a new burning tool which is similar to **Windows BDT** for Windows OS and Linux OS, but this tool requires Burning EVK with new firmware. This tool can be used to flash the image to TLSR8 and TLSR9 chips through SWS. This tool is still in the beta phase. If this tool is not working on some Windows OS, use the **Windows BDT** (mentioned in the following **Windows BDT** section) instead.

Users can launch this tool and click the help on libUSB BDT menu to learn how to use it.

Clicking this menu entry would open the libusb version of BDT and pass the artifact bin path to it if a project has been selected and the artifact has been found, so users do not require to copy the artifact path manually:

T Telink Debug Tool			- U	×
Device File Tool Help				
B91 👻 < evk 💌 🖑 sws b0 10 b0 10	▷ run	00 pause	Ø⊅ step	
Image: Bash ▼ Image: Description of the second se	Q pc	⊳ start		
🗟 read 🖻 write 🐺 download 🔶 erase 🕐 reset 😤 stive	.≣var	🗉 stall		
Image: Control of the second seco	nk utils rojects\CDT_ _btble_contro : ⓒ ▼ හ	IDE_dev\BTBL biller__prj_btbl	E_SDK\b91_bt e_controllerk ▼ : 茶 ▼ 0	ble_controller_sdk\ bin • 😢 • 🗣 •
Project Explorer X Project E				There is no active editor that provides an outline.

Figure 7.12: Open libUSB BDT with artifact Path

If no project is selected, a error dialog would show, users can close it, the libusb BDT program would be started normally as well:

🖭 Une	expected Selection Class	×
	Expected a TreeSelection but got a StructuredSelection instead.	
	Processing Terminated.	
	ОК	

Figure 7.13: Open libUSB BDT without artifact Path

7.6 Toolchain shell or ICEMan shell menu entry

This is a submenu, it has several menu entries to start the toolchain console or ICEMan console, users can input commands (for example: riscv-elf-gcc or ICEMan) in these shells.

It is worth noting that we provide two versions of ICEman. Under \${IoTStudio_Path}/tools/ice_cygwin_from_andes/ , there is the original version of ICEman, while under \${IoTStudio_Path}/tools/ice/, there is a new version of ICEman used by IoTStudio by default. It has made some modifications to match the functionality of IoTStudio and currently does not support IPv6.

/endor	/cor	nmon/subdir.mk - Telink IoT Studic							
	T	Telink RDS to IDE Converter) ▼ 9: ▼ 9: ▼ 1: ഈ - タ ▼ 1: ഈ 回					
€ F	N 	Telink IoTStudio to Cmake Windows BDT (SWS and Jtag) Jtag Burn (SPI Burn)		: ×					
ster]	٢	🔰 Telink IoT Studio User Guide	-	Console 🗙 🔲 Properties 🖷 Progress					
		Projects and Artifacts shortcuts	>						
	Ť	Toolchain Shell or ICEMan shell	> [Launch RISCV V5F(D25F) toolchain Shell (RDS 3.2.3 GCC 7.4) for TLSR9					
	(j) Q	Telink Links Search on DocSite	> 	 Launch RISCV V5(N22) toolchain Shell(RDS V3.2.3 GCC 7.4) for TLSR9 Launch RISCV V5F(D25F) toolchain Shell(RDS V5.1.2 GCC 10.3) for TLSR9 Launch RISCV V5(N22) toolchain Shell(RDS V5.1.2 GCC 10.3) for TLSR9 Launch RISCV V5F(D25F) toolchain Shell (RDS 5.3.x GCC 12.2) for TLSR9 Launch RISCV V5(N22) toolchain Shell (RDS 5.3.x GCC 12.2) for TLSR9 Launch RISCV V5(N22) toolchain Shell (RDS 5.3.x GCC 12.2) for TLSR9 Launch RISCV V5(N22) toolchain Shell (RDS 5.3.x GCC 12.2) for TLSR9 Launch RISCV V5(N22) toolchain Shell (RDS 5.3.x GCC 12.2) for TLSR9 					

Figure 7.14: Shells

7.7 Windows BDT

The Windows BDT (SWS and JTAG) menu item would open the standalone program Windows BDT . Users can use this tool to flash or debug the TLSR8 and TLSR9 chips.

Refer to this wiki page on how to use it.

7.8 Open artifact path

There's an icon on the toolbar used to open the artifact directory:



Figure 7.15: Open artifact path

It wouldn't work if the project was not properly configured (the .cproject file), this occurs in many old TLSR8 SDKs.

While most TLSR9 SDK projects work as expected.

7.8.1 Notice

The project or project file must be selected before clicking the icon, otherwise it will not work.

7.9 Copy artifact path

The Telink IoT studio added a handy menu entry to help users copy the artifact path:



Users can click this icon to copy the artifact path to the system clipboard.

7.9.1 Notice

The project or project file must be selected before clicking the icon, otherwise it will not work.

7.9.2 In other places

The Copy artifact path menu entry also shows at toolbar and right-click the pop-up menu:

 III - Telink IoT Studio

 File Edit Source Refactor Navigate Search Project Run I Telink Window Help

 III - Telink No Studio

 III - Telink No Studio

 III - Telink No Studio

 III - Telink Window Help

 III - Window Help

 IIII - Window Help

 IIII

Figure 7.17: Copy artifact path on Toolbar



Figure 7.18: Copy artifact path on pop-up menu

7.10 Search on DocSite

This tool can be used to search for specific keywords in the Telink Document Center website. Double click a keyword in the project file and click "Search on DocSite" to jump to the Document Center search page.



Figure 7.19: keyword

https://doc.telink-semi.cn/inde × +				✓ - □ >
→ C △	ex/index/search.html?search=reg_gpio_out&type=	&handbook=&product=		🖻 🖈 🖪 🛡 🔺 🖬 🔇
hrome 🧧 视频站 📕 工具 📕 搜索 📕 🗧	书签 🧧 泰凌 📕 临时资源			
र Telink र्थ	次件 硬件 产品变更通知			EN ~ 社区
		文档中心		
	reg_gpio_out		<u> </u>	
日米	搜索结果			
如何入门 ~				
开发手册~	12余元46,14日101121112系旦间的从面			
应用指南 ~	-			
			xU	

Figure 7.20: doc site

8 Common compiling error

Telink

T

8.1 Can't find the link script file

This type of error would occur when link script file path is in relative format. This wouldn't be an issue in Telink RDS IDE , but there will be an error in Telink IoT studio .

The error log would hint like the following:

	~	4
Building target: AES_Demo.elf Invoking: GNU RISC-V Cross C Linker risv32.elf, gcc =mcmodel=medium -02 -fmessage-length=0 -ffunction-sections -fdata-sections -flto -g3 -T/link/flash_boot.link -nostartfiles -Klinkergc-sections -L"C:\Projects \CDT_IDE_dev\telink b9im_driver_src\thip\B31\drivers\lib' ~NL, Hap, AES_Demo.map' -02 - fpack-struct -fshort-enums -g3 -mcpu-d25f -ffunction-sections -fdata-sections -mext-dsp -mabi=1]p32f - \CDT_IDE_dev\telink b9im_driver_src\thip\B31\drivers\lib'src/hip.GEDemo.map' -02 -fpack-struct -fshort-enums -g3 -mcpu-d25f -ffunction-sections -fdata-sections -mext-dsp -mabi=1]p32f - \/Uredor/Common/flash/puya flash_scratch.o./vendor/common/flash/puya flash_trim.o./vendor/common/falibration.o./vendor/Common/flash/puya_common.o. //vendor/AES_Demo/apino. //drivers/lib/src/hke/ke_common.o. //drivers/lib/src/hke/ke_common.o./drivers/lib/src/hke/ke.o //drivers/lib/src/hke/ke_common.o./drivers/lib/src/hke/ke.o //drivers/lib/src/hke/ke_common.o./drivers/lib/src/hke.o //drivers/lib/src/hke/ke_common.o./drivers/lib/src/hke.o //drivers/lib/src/hke/ke_common.o./drivers/lib/src/hke.o //drivers/lib/src/hke/ke_common.o./drivers/lib/src/hke.o //drivers/lib/src/hke/ke_common.o./drivers/lib/src/hke.o //drivers/lib/src/hke/ke_common.o./drivers/lib/src/hke.o //drivers/lib/src/hke/ke_common.o./drivers/lib/src/hke.o //drivers/lib/src/hke/ke_common.o./drivers/lib/src/hke.o //drivers/lib/src/hke/ke_common/drivers/lib/src/hke.o //drivers/lib/src/hke/ke_common/drivers/lib/src/hke.o //drivers/lib/src/hke/ke_common/drivers/lib/src/hke.o //drivers/lib/src/hke/ke_common/drivers/lib/src/hke.o //drivers/lib/src/hke/ke_common/drivers/lib/src/hke/ke_common/db/deb/starte_db/drivers/lib/src/hke.o //drivers/lib/src/hke/ke_common/db/deb/starte_db/drivers/lib/src/hke/ke_common/db/deb/starte_db/drivers/lib/src/hke/ke_common/db/deb/starte_db/drivers/lib/src/hke/ke_common/db/deb/starte_db/drivers/lib/src/hke/ke_common/db/deb/starte_db/drivers/lib/src/hke/ke_common/db/deb/starte_db/drivers/lib/src/hke	5	
1. State Duild Exclude 1 annual a wanning (took 21, 477s-)		
IO-94-35 BOLLD FRIED, I EFFORS, & MBHININGS, (COOK DIS-42/MS)		
I	*	1

Figure 8.1: Link error

To resolve this problem, users can open the project property, then navigate to linker's General page of Tool Settings tab as following image, then double click the link script file:



Figure 8.2: Link file path

On opened dialog, select it from workspace :



Click OK and apply and close button, and rebuild the project to check the results.

9 Other plugins or functions

9.1 Telink Formatter

This plugin is used to format the code/file with format shortcut or saving a file with clang-format.

9.1.1 Use as formatter

To use it in formatter, users should set the Telink Formatter as the default formatter. Open the setting from the menu Windows -> Preference -> C/C++ -> Code style -> Formatter , select the TelinkFormatter in Code Formatter: list, then click the Apply and Close button:

	Preferences	0 8
type filter text	Formatter	← ➡ ⇒ <
▶ General	<u>(</u>	Configure Project Specific Settings
▼ C/C++	A <u>c</u> tive profile:	
Appearance	GNU [built-in]	<u>E</u> dit <u>R</u> emove
 Autocools Build 	Ne <u>w</u> I <u>m</u> port	
CMake	Code Formatter:	
Code Analysis	TelinkFormatter	*
 Code Style Code Templates 	Note: Contributed formatters may	not respect all code style settings.
Formatter	Prev <u>i</u> ew:	
Name Style	/* * A sample source file for the	code formatter preview
Organize Includes	*/	i code formaccor protien
 Debug 	#include <main.n></main.n>	
Docker Container	<pre>class Point {public:Point(doub</pre>	<pre>vle x, double y) : x(x), y(y)</pre>
▶ Editor	<pre>double x;double y;};</pre>	
File Types Indexer	<pre>double Point::distance(const P</pre>	Point& other) const {double dx
JSON Compilation D	(Restore <u>D</u> efaults <u>Apply</u>
? > 2 (0)		Cancel Apply and Close
	Figure 9.1 : Formatter setting	

 The select Telink style in the settings of TelinkFormatter
 Open the setting from the menu
 Windows

 ->
 Preference
 ->
 C/C++
 ->
 TelinkFormatter
 , select the
 Telink
 in
 Clang-ormat fallback style:

 list, then click the
 Apply and Close
 button:
 Select the
 Select th







Afterwards users can press the shortcut keys (by default, it's Ctrl+Shift+F) on any openned source code file editor, a dialog would show, uses click the corronsponding button and option to confirm:

Select Formatting Scope						
No code is selected. Choose the scope of formatting.						
O The entire file						
O The statement on the current line						
Remember this choice and do not ask again						
Cancel OK						

Figure 9.3: Formatter hint

After the Telink Formatter has formatted the code/file, users can view the logs on the Telink Formatter output console for details (notice that the log function should be enabled in the TelinkFormatter Preference dialog, check the following guide):

Telink

т



Figure 9.4: Formatter console output after formmtting

9.1.2 Trigger with saving event(recommanded method)

Telink

т

The formatter can be triggered when a file or multiple files are saving, this requires the selection of the function in TelinkFormat setting(through the menu Windows -> Preference -> C/C++ -> TelinkFormatter):

	Preferences 🔳	0
Telink 🛛	TelinkFormatter 🗘 🖛 🖒 🖛	000
▼ C/C++ TelinkFormatter	The fallback style option would be used, if no style file is found on the path of project tree Clang-format fallback style: none Clang-format on file save Show clang-format log in console	
	Rescore Deraults Apply	
? ڬ 🖆 🔘	Cancel Apply and Clos	e

Figure 9.5: Telink Formatter Preference

After the Telink Formatter has formatted the code/file, users can view the logs on the Telink Formatter



output console for details if the log has been enabled in setting:

Figure 9.6: Formatter console output after saving

9.1.3 Known issue

9.1.3.1 Save button is disabled at the first time

The "Save" button is int disabled state when a file is modified at the first time. After save it using "CTRL+S" (or File -> Save , or with Save all menu-item/button), modify this file again, it will become enabled state.

9.2 Easy Shell

This plugin allows to open a shell window or file manager from the pop-up menu in the navigation tree or editor view. Additionally, it is possible to run selected files in the shell, copy file or directory path or run user defined external tools.

> 🖳 driver		New	>		
> 🔂 link		Go Into			
> 👝 UART		00 1110			
> 🔁 vendo	÷,	EasyShell	>	>	Open Command Prompt Here
> 💋 3rd-p		Open in New Window		\sim	Show in Explorer
📑 Backu		Show In	Alt+Shift+W >	ß	Copy Full Path to Clipboard 📡
		Show in Local Terminal	>	ß	Copy Qualified Name to Clipb \${easyshell:r
		Carry	culue.		



9.3 ECalculator

A calculator with lots of functions, including:

- Display of the calculation in base-n numbers
- Standard, scientific and trigonometric calculation
- Enter arithmetic expressions naturally



Users can open it from the menu: Window -> Show view -> other... , then input Ecalculator to search and open it:

🖹 Proble	ems 🧔	Tasks	📮 Conso	ole 🔲 Pr	operties	EC	alcula	tor X							
			D	EG RAD	GRAD	HEX	DEC	BIN	Store	Recall	Clear	Last Result	C AC	< <<	000
													25bc	: OR fe1	1 ^
															\sim
()	%	1/n	^2	sqrt		7	8		9		+			
~	^1/n	In	e^	log	10^		4	5		6					
F	ni		OR		XOR		1	2		3		-		=	
-	P	-			XOR		0			+/-		*			
sin	cos	tan	asin	acos	atan		Α	B		С					
sinh	cosh	tanh					D	E		F		/			



9.4 Terminal in Eclipse

The terminal can be used for inputting git commands and serial debugging.

Users can open it from the menu Window -> Show View -> Terminal :

🔳 Launch Termina	ıl	—		×
Choose terminal: Settings	Git Bash Git Bash Local Terminal			~
Encoding: UTF-	SSH Terminal Serial Terminal Telnet Terminal			
?	ОК		Cance	el

Figure 9.9: Terminal Selection

Choose term	inal: Serial Terminal		~
Settings			
Serial port:	a		~
Baud rate:	115200		~
Data size:	8		\sim
Parity:	None		\sim
Stop bits:	1		~
Encoding:	Default (ISO-8859-1)		~
?	ОК	C	ancel

Figure 9.10: Terminal for Serial

If your serial debugging log output frequency is very fast, use other dedicated tools instead.

9.5 Binary Viewer

This plugin is used to open the binary files:

🎦 Project Ex 🗙 🗖 🗖	p10 UART_Demo.bin ×		
🖻 🔩 🍸 🕴	🔄 🦘 🥟 👖 HEX 🚽 🗄 🖸 🔞		_
> 📂 b85m_ble_single_conne	00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F	Rinary	-
✓ 💏 > B91_Driver_Demo (in	0000 25 A0 00 00 00 00 00 00 00 00 00 00 00 00		
> 🗊 Includes	0010 00 00 00 00 00 00 00 D8 34 00 00 00 00 00 00 4		
> ᇋ AES_Demo	0020 4B 4E 4C 54 00 00 3B 17 97 01 08 E0 93 81 81 7D KNLT ; }	Byte	
> 🚘 boot	0030 B7 02 10 80 93 82 02 70 13 03 00 04 A3 80 62 50 p bP	37	
> 🚘 common	0040 97 02 0A E0 93 82 02 FC 16 81 97 02 00 E0 93 82	Word	=
> 🚘 drivers	0050 62 FB 73 90 02 80 99 62 F3 A2 02 30 73 10 30 00 b s b 0s 0	0632	-
> 🚘 link	0060 97 22 00 E0 93 82 02 7A 73 90 52 30 73 60 01 7D zs R0s }	9032	_
∽ ≽ UART_Demo	0070 B7 02 00 E4 09 43 23 A0 62 00 F3 22 A0 7C 93 E2 C# b "	Integer	
> 👝 boot	0080 12 00 93 E2 22 00 73 90 A2 7C 0F 10 00 00 17 03 s	631242752	
> 🗁 common	0090 00 00 13 03 23 10 97 03 00 E0 93 83 A3 F6 17 0E #	Long	
> 🗁 drivers	00A0 00 E0 13 0E 6E 0B 63 F9 C3 01 83 22 03 00 23 A0 n c #	Long	
> 📂 vendor	0000 03 00 11 03 91 03 05 DF 17 05 00 00 13 03 05 23 5 1 #	2711166975677038592	
🗋 makefile	0000 93 03 80 13 1, 02 00 20 13 02 42 02 03 F9 03 01	Float	
objects.mk	00R0 00 00 13 03 A3 25 97 03 00 R0 93 83 A3 11 17 38 %	2 77555765-16	- 1
osources.mk	00F0 00 E0 13 0E 4E 81 63 F9 C3 01 83 22 03 00 23 A0 N c " #	2.11333762-10	
⁰¹⁰ UART_Demo.bin	0100 53 00 11 03 91 03 C5 BF 17 33 00 00 13 03 83 39 S f 3 9	Double	
UART_Demo.map	IITE 8 A 13 529 (0)		
> 🗁 vendor	017-6************************************	- <u> </u>	UVA
> 🗭 3rd-party	🖹 Problems 🧔 Tasks 📮 Console 🗙 🔲 Properties 📃 ECalculator 🤨 Error Log	🗙 🕂 🗘 😫 🌆	a :

Figure 9.11: Binary Viewer





Users can double-click a binary file to open and view it in IoT Studio.

10 FAQ

10.1 Error 127 when compiling the code

This error would give the Error 127 at console window:

👔 Problems 🧟 Tasks 📮 Console 🗙 🔲 Properties
CDT Build Console [B92_Driver_Demo]
riscv32-elf-gcc -mcmodel=medium -mcmodel=medium -02 -fmessage-length=0 -ffunction-sections -fdata-sections -flto -g3 -DMC \telink_b9lm_driver_src_CI\chip\B92\drivers" -I"E:\SDKs\telink_b9lm_driver_src_CI\demo\vendor\common\B92\calibration" -I"E \telink_b9lm_driver_src_CI\demo\vendor\common\common" -mext_dsn -mabisiln32fcc_fmessage_length=0 _fomit_frame-pointer
zero-variadic-macro-arguments -fpack-struct -fshort-enums -fno-jump-tables -MMD -MP -MF"drivers/lib/src/pke/ecdh.d" -MT"dr "E:/SDKs/telink_b91m_driver_src_CI/chip/B92/drivers/lib/src/pke/ecdh.c"
Building file: E:/SDKs/telink_b9lm_driver_src_CI/chip/B92/drivers/lib/src/pke/pke.c
make: *** [drivers/lib/src/pke/subdir.mk:55: drivers/lib/src/pke/ecdsa.o] Error 127 make: *** Waiting for unfinished jobs
make: *** [vendor/common/subdir.mk:30: vendor/common/printf.o] Error 127
Building file: E:/SDKs/telink_b9lm_driver_src_CI/chip/B92/drivers/lib/src/pke_prime.c
make: *** [vendor/common/calibration/subdir.mk:20: vendor/common/calibration/calibration.o] Error 127
make: *** [[vendor/common/subdir.mk:23: vendor/common/plic_isr.o] Error 127
make: *** [drivers/lib/src/pke/subdir.mk:41: drivers/lib/src/pke/eccp_curve.o] Error 127
make: *** [vendor/UART_DEMO/subdir.mk:40: vendor/UART_DEMO/main.o] Error 127
make: *** [vendor/UART_DEMO/subdir.mk:33: vendor/UART_DEMO/app_dma.o] Error 127
make: *** [vendor/UART_DEMO/subdir.mk:26: vendor/UART_DEMO/app.o] Error 127
Invoking: GNU RISC-V Cross C Compiler

Figure 10.1: Error127

This issue is caused by the incorrect setting of the build tool version, setting the correct build tool version would fix this problem.

10.2 Not found make in PATH or nothing to build when building

These problems are normally caused by importing the Telink RDS IDE format projects to Telink IoT studio directly without converting.

To resolve this problem, remove the project and convert it to Telink IoT studio before importing.

10.3 Orphaned configuration and no options on Settings tab

When this occurs, the properties of the project would be like the following:



Properties for B91_Driver_Demo





This problem is caused by the same reason as previous FAQ, converting Telink RDS format project before importing it.

10.4 How to verify the current used toolchain gcc version for TLSR9

Users can set the following command to post command in project properties:

• Telink

Properties for B91_Driver_Demo

	Settings
 Resource Builders C/C++ Build Build Variables Environment Logging 	Configuration: UART_Demo [Active] ~
Settings Tool Chain Editor C/C++ General Git Linux Tools Path Project Natures Project References	Pre-build steps Command: Description:
RISC-V Toolchains Paths Run/Debug Settings Task Tags TC32 Loader Validation WikiText	Post-build steps Command: riscv32-elf-gcc -v Description:

Figure 10.3: GCC version setting in Post command

Then build the project, the console will output the GCC version:



10.5 How to verify current used make tools

Users can set following command (echo "\${PATH}") to post command in project properties:

Properties for B93_Driver_I	Demo	— 🗆 X		
type filter text	Settings	<> → <> → 8		
 Resource Builders C/C++ Build Build Variables 	Configuration: N22_Test_Demo [Active] ~	Manage Configurations		
Environment Logging Settings Tool Chain Editor > C/C++ General Git Linux Tools Path Project Natures Project References RISC-V Toolchains Paths Run/Debug Settings Task Tags TC32 Loader > Validation WikiText	Build Steps Build Artifact Binary Parsers & Error Parsers Pre-build steps Command: Description:			

Figure 10.5: Make tool setting in Post command

Then build the project, the console will output the path that the first one is the make tool path:

🖹 Problems 🧔 Tasks 📮 Console 🗙 🔲 Properties										
CDT Build Cor	nsole [B9	3_Driver_	Demo]							
text	code ro	odata	data	bss	dec	hex filename				
6520	6072	448	37	60	6617	19d9 N22_Test_Demo.elf				
Finished bu Finished bu	ilding: ilding:	N22_Tes N22_Tes	st_Demo.s st_Demo.l	iz st						
<pre>rinisned building: N22_Test_Demo.1st echo "C:\TelinkIDE\RD\$\V3.2.3\cygwin\bin;C:\TelinkIDE\RDS\V3.2.3\toolchains\nds32le-elf-mculib- v5\bin;C:/TelinkIDE//plugins/org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.2.v20220201- 1208/jre/bin/server;C:/TelinkIDE//plugins/org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.2.v \system32;C:\Windows;C:\Windows\System32\Wbem;C:\Windows\System32\WindowsPowerShell\v1.0\;C:\Windows\Syste \AppData\Local\Microsoft\WindowsApps;;C:\Users\admin\AppData\Local\Programs\Microsoft VS Code\bin;C:\TelinkID C:\TelinkIDE\RDS\V3.2.3\cygwin\bin;C:\TelinkIDE\RDS\V3.2.3\toolchains\nds32le-elf-mculib-v5\bin;C:/TelinkI 1208/jre/bin/server;C:/TelinkIDE//plugins/org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.2.v \system32;C:\Windows;C:\Windows\System32\Wbem;C:\Windows\System32\WindowsPowerShell\v1.0\;C:\Windows\Syste \AppData\Local\Microsoft\WindowsApps;;C:\Users\admin\AppData\Local\Programs\Microsoft VS Code\bin;C:\TelinkID \AppData\Local\Microsoft\WindowsApps;;C:\Users\admin\AppData\Local\Programs\Microsoft VS Code\bin;C:\TelinkID \AppData\Local\Microsoft\WindowsApps;;C:\Users\admin\AppData\Local\Programs\Microsoft VS Code\bin;C:\TelinkID \AppData\Local\Microsoft VS Code\bin;C:\TelinkID</pre>										
15:10:23 Build Finished. 0 errors, 31 warnings. (took 7s.458ms)										

Figure 10.6: Verify the make tool path

In the above example result, the make tool path is C:\TelinkIDE\RDS\V3.2.3\cygwin\bin with base version V3.2.3 .

10.6 The permission error occurred while opening the Telink TC32 console



Figure 10.7: Permission problem

The user should open Telink TC32 console with the privileges of administrator .

10.7 How to enable case sensitivity feature on Windows OS

On the Windows OS, by default, file names are not case sensitive. For example, when you have a header file name of **app.h** and another file name of **App.h**, by default, Windows cannot distinguish them correctly. If you need to enable case sensitivity, you can use the following methods.

Firstly, you need to enable WSL:

Telink

T

👩 > 控制面板	> 程序	
iet		程序和功能 卸数程序 ● 启用或关闭 Windows 功能 查看已安装的更新 运行为以前版本的 Windows 编写的程序 如何安装程序 默认程序 更改媒体或设备的款认设置
		🐻 Windows 功能 — □ ×
		启用或关闭 Windows 功能 ご 君要自用一种功能,请选择其复选框。若要关闭一种功能,请清除具复选框。 机 填加的理题示仪屈用规议加助一部分。 福祉 東加的理题示仪屈用规议加助一部分。 B ● Windows Process Activation Service Mindows Process Activation Service Mindows Process Activation Service Windows Process Activation Service Mindows Process Activation Service Mindows TFI Filter Windows IFIF Filter Windows 虚拟机监控理学平台 Windows 虚拟机监控理学平台 B ● 打印和文件報告 C 「在文件来告予論 面備單 TCPIP 服务(即 echo、daytime 等) B ● 目版组件 C 盧囲手 Linux 的 Windows 子系统 正理差分压缩 API 支持





Then, open Windows PowerShell as an administrator and navigate to the directory you need to use, and execute the following command:

(Get-ChildItem -Recurse -Directory).FullName |
ForEach-Object {fsutil.exe file setCaseSensitiveInfo \$_ enable}

11 Known issues

11.1 Save button is disabled at the first time

Use the Ctrl+S or menu item File -> Save , or click the Save All button/menu-item or use Ctrl + Shift +S . The save button would become enabled state in the following modifications.

This is caused by Telink Formatter plugin.

11.2 IoT Studio exits if users click the Open artifact path multiple times

This occurs in that the IoT Studio cannot find the artifact path. Users should avoid clicking the toolbar icon **Open artifact path** multiple times in a short time.

11.3 IoT Studio will block for a while after importing project and changing toolchain

This is a normal case as the IoT Studio would take a while to do the **code indexing** and scan compiler files (for example the toolchain system header files and directories).

This procedure would take a few seconds. Users should wait for this job done before other operations (eg: building). This is a one-shot action, it will happen on toolchain changing settings or importing projects.

11.4 The size of generated artifacts are zero

This issue will occur if users start to build a configuration when the project is just imported and the makefile hasn't been generated fully.

When this happens, a log similar to the following example would display on the console log window:

riscv32-elf-gcc -mcmodel=medium -mcmodel=medium -02 -fmessage-length=0 -ffunction-sections -fdata-sections -flto -g3 -T ../boot.link -nostartfiles -Xlinker --gc-sections -L"E:\TelinkIDE\[Src]telink_eagle_ble_multi_connection_src\B91_ble_multi_conn_src\algorithm\lc3" -Wl,-Map,"B91_demo.map" -g3 -mcpu=d25f -ffunction-sections -fdata-sections -fmessage-length=0 -fno-builtin -fomit-frame-pointer -fno-strict-aliasing -fshort-wchar -fuse-ld=bfd -fpack-struct -O2 -o "B91_demo.elf" ./vendor/common/hci_transport/hci_dfu.o ./vendor/common/hci_transport/hci_dfu_port.o ./vendor/common/hci_transport/hci_h5.o ./vendor/common/hci_transport/hci_slip.o ./vendor/common/hci_transport/hci_tr.o ./vendor/common/hci_transport/hci_tr_h4.o ./vendor/common/hci_transport/hci_tr_h5.o ./vendor/common/blt_common.o ./vendor/common/blt_fw_sign.o ./vendor/common/blt_led.o ./vendor/common/blt_soft_timer.o ./vendor/common/common_dbg.o ./vendor/common/custom_pair.o ./vendor/common/device_manage.o ./vendor/common/flash_fw_check.o ./vendor/common/simple_sdp.o ./common/tl_queue.o ./common/utility.o -lm -ldsp /cygdrive/d/TelinkIDE/RDS/V3.2.3/toolchains/nds32le-elf-mculib-v5f/bin/../lib/gcc/riscv32-elf/7.4.0/ ../../../riscv32-elf/bin/ld.bfd: warning: cannot find entry symbol _RESET_ENTRY; defaulting to 0000000020000000 Finished building target: B91_demo.elf Invoking: GNU RISC-V Cross Create Flash Image Invoking: GNU RISC-V Cross Create Listing Invoking: GNU RISC-V Cross Print Size riscv32-elf-objcopy -O binary "B91_demo.elf" "B91_demo.bin" riscv32-elf-size "B91_demo.elf" riscv32-elf-objdump --source --all-headers --demangle --line-numbers --wide "B91_demo.elf" > "B91_demo.lst" text code rodata data bss dec hex filename 0 B91_demo.elf Θ Θ Θ Θ Θ Θ

Figure 11.1: Artifact size is 0

To resolve this problem, users should wait a few seconds then build the configuration again.

To avoid this issue, users should wait for the makefiles to be generated and the indexing to be finished. The process can be viewed at the right bottom of the IoT Studio:

Telink_WS_N22 - Telink IDE					_		×
<u>F</u> ile Edit <u>S</u> ource Refac <u>t</u> or <u>N</u> av	<i>i</i> igate Se <u>a</u> rch <u>P</u> roject	<u>R</u> un 🔳 Telink	<u>W</u> indow <u>H</u> elp				
i 📩 🗕 🔚 🕼 🗞 🖛 🍕 🖬 🔝	🗉 🐁 🔍 🖉 😓 🖷	i 😻 🔳 📸 🔻	· 🚳 🕶 💽 🕶 🎯	* * • • • •	} - 🍋 -	19 🔗	•
BP ■ ¶ B ▼ P ▼ P → P → P	>> - ⊡					Q : E	\$ 🖬
Project Explorer 🗙 🗖 🗖				- 8	₽o×	»3 3	
E					There is that prov	no active vides an o	editor outline.
	🖹 Problems 🧔 Tasks	📮 Console 🛛	Properties		🌒 🛃 🔁 ·	- 📬 -	- 8
	Eclipse Embedded CDT C	MSIS Packs con	sole				
	4				-		•
🕞 B93_Driver_Demo		Disc	over compiler bui	ttings: (64%)	b i -7	🗇 🖻	20

Figure 11.2: IoT Studio is busy

11.5 Clicking the telink menu is invalid

This issue will occur if users do not select any project. To avoid this issue, select a project before clicking the telink drop-down menu.

11.6 The problem that the source file cannot be found in debugging

In the process of debugging, there may be an error that the source file cannot be found. At this time, click Edit Source Path... , as shown below:

l main.c l app_dma.c	(×)= Va × ⁰₀ Br 🤄	<pre></pre>	🔀 Pe 🔓 Pe 📮 🗖
Can't find a source file at			ڭ 🖆 🖻 🗠 🐛
"/cygdrive/c/Users/wf_15/Documents/telink_b91m_driver_sdk/demo/vendor/UART_DEMO/app_dma.c"	Name	Туре	Value
Eocate the file of edit the source lookup path to include its location.			
View Disassembly			
Locate File			
Edit Source Lookup Path			
Configure when this editor is shown Preferences			
			^
			~
	<		>
Console × 🎟 Registers 🔮 Problems 📀 Executables 💀 Debugger Console 🔋 Memory	× .) () (\$) 🖬 🖬 🖷 🛼 (🛃 🚽 🖢 🕶 🖻
CDT Build Console [B91_Driver_Demo]			
11:11:15 **** Incremental Build of configuration UART_Demo for project B91_Driver_Demo ****			^
make -j8 all Invoking: GNU RISC-V Cross Print Size			
riscv32-elf-size "UART_Demo.elf"			
text code rodata data bss dec hex filename			
Finished building: UART_Demo.siz			
11:11:15 Build Finished, 0 errors, 0 warnings, (took 290ms)			
			~
<			>

Figure 11.3: locate

Then follow the instructions in Chapter Edit Path Mapping to configure path mapping

12 IoT Studio settings

12.1 Hide print margin

The print margin will show in default, which is a vertical line in the code editor:

Telink

Image: Image	
Image: Control of the source file for B9lm Image: Control of the source file for B9lm<	1
<pre>1@ /************************************</pre>	
<pre>2 * @file dbgport.c 3 * 4 * @brief This is the source file for B9lm 5 * 6 * @author Driver Group 7 * @date 2019</pre>	
<pre>3 * 4 * @brief This is the source file for B9lm 5 * 6 * @author Driver Group 7 * @date 2019</pre>	1
4 * @brief This is the source file for B9lm 5 * 6 * @author Driver Group 7 * @date 2019	1
5 * 6 * @author Driver Group 7 * @date 2019	
6 * @author Driver Group 7 * @date 2019	
7 * @date 2019	
8 *	
9 * @par Copyright (c) 2019, <u>Telink</u> Semiconductor (<u>Shanghai</u>) <u>Co</u> ., <u>Ltd</u> . ("TELINK")	
10 * All rights reserved.	
11 *	
<pre>12 * Licensed under the Apache License, Version 2.0 (the "License");</pre>	
13 * you may not use this file except in compliance with the License.	
14 * You may obtain a copy of the License at	
🖳 Brahlana 💼 Taula 🔲 Casaala X 🔲 Branadia	

Figure 12.1: Print margin in text editor

If you want to hide it, you can uncheck the Show print margin option on the Text Editors page of preferences.

T Preferences					×
type filter text	Text Editors		¢	• 🔿	▼ 00
 General Appearance Compare/Patch Content Types Editors Autosave File Associations Large File Associ Structured Text E Text Editors Accessibility Annotations Hyperlinking Linked Mode Quick Diff Spelling WikiText Globalization Keys 	Displayed tab width: Insert spaces for tabs Remove <u>multiple</u> spaces on backspace/delete Highlight current line Show print margin Print margin colu <u>m</u> n: Allow <u>e</u> ditors to override the margin column Show line num <u>b</u> ers Show line num <u>b</u> ers Show c <u>u</u> rsor position in the status line Show selection size in the status line Show selection size in the status line Show range indicator Show whitespace characters (<u>configure visibility</u>) Show affordance in hover on how to make it sticky When mouse moved into hover: Enable drag and drop of text Warn before editing a derived file	4 80 Enrich after delay	~		
Link Handlers > Network Connectio	Show code minings <u>f</u> or problem annotations:	None		~	



13 IoT Studio Debug Tool For TLSR9

Note: This jtag debug tool is only used for TLSR9 series chips

13.1 Build debuggable programs

To use the debugging function, when building an executable file, add the -g option to the compilation options, and pay attention to check the compiler and linker parameters, and delete all the -O options.

And the optimization option should be selected as -Og, and cannot be set to -OO, because in the test, it is found that the program compiled with the -OO option cannot run normally after downloading, and the following error will occur when trying to download again:

Logging: <= #Iag_Burn ICEMan ==> option <i, image=""> para<clubers b92="" delldesktop="" gpio_dem<="" gpio_demo="" project="" telink_b91m_driver_src="" th="" tlsr_riscv="" wks=""><th>Args:(C:\TellinkloTStudio\ROS(V5.1.1\u00edce\L'Eman.exe, -Z. v5, -b, 2354) Andes (L'Eman (OpenOCD) 5.1.1-g3a34fc9 (2022-04-22-14:27) Burner listens on 2354 Telnet port: 4444 TCL, port: 6666 Andes AICE-MINI+ The core #0 listens on 1111. (Efman is ready to use. hardware reser-and-hold success on [nds.tap] hart 0 < Unable to halt [nds.tap] hart 0 -> dmcontrol=8x000430ca2 < Unable to halt [nds.tap] hart 0 -> dmsturs =0x00439ca2</th></clubers></i,>	Args:(C:\TellinkloTStudio\ROS(V5.1.1\u00edce\L'Eman.exe, -Z. v5, -b, 2354) Andes (L'Eman (OpenOCD) 5.1.1-g3a34fc9 (2022-04-22-14:27) Burner listens on 2354 Telnet port: 4444 TCL, port: 6666 Andes AICE-MINI+ The core #0 listens on 1111. (Efman is ready to use. hardware reser-and-hold success on [nds.tap] hart 0 < Unable to halt [nds.tap] hart 0 -> dmcontrol=8x000430ca2 < Unable to halt [nds.tap] hart 0 -> dmsturs =0x00439ca2
	· · · · · · · · · · · · · · · · · · ·

Figure 13.1: Jtag Burn err

In the process of building the program, you may encounter syntax errors like the one shown in the figure below. The solution is to delete options such as -Wall -Werror of the compiler (also pay attention to checking the parameters of assembler and linker):



Figure 13.3: delete option

One thing to note is that when building the V5.1.2 N22 program, because of the tool chain, when using the GCC 10 version of the N22 tool chain, delete the -mabi=ilp32f option (also pay attention to check the parameters of assembler and linker).

Telink

T

type filter text	Settings	\$ * \$ * 1
 Resource Builders 		
▼ C/C++ Build	Tool Settings Toolchains	ûContainer Settings →Build Steps 😤Build Artifact 📾Binary Parsers ΘError Parsers
Build Vallables Environment Logging Tool Chain Edit -> C/C++ General Linux Tools Path -> MCU Project Natures Project Reference Run/Debug Settii Task Tags TC32 Loader -> Validation WikiText	 Target Processor Optimization Warnings Debugging S GNU RISC-V Cross Assembler Preprocessor Includes Warnings Miscellaneous S GNU RISC-V Cross C Compiler Preprocessor Includes Optimization Warnings Miscellaneous S GNU RISC-V Cross C Linker General Libraries Miscellaneous S GNU RISC-V Cross Crabe Flash Image General S GNU RISC-V Cross Create Listing General S GNU RISC-V Cross Print Size General 	Cenerate assembler listing (-Wa,-adhins="\$@.ist") Save temporary files (-save-temps Use with caution!) Verbose (-v) Other compiler flags (-mext-dsp imabialip32f) - c-fmessage-length=0 -fomit-frame-pointer -fno-strict-aliasing -fshort-wchar -fuse-ld=bfd -std=c99 -Wno-gnu-ze Opport ANSI Programs, same as RDS (-b, si) Code Coverage, same as RDS (-coverage) Apply HW DSP options(-mext-dsp) delete it
?		Cancel Apply and Close
		Figure 13.4: delete_mabi

13.2 Burn program

Telink

т

When burning the program, pay attention to burner's path and chip type, and then click Start ICEman. After ICEman is ready, the telnet_port and burner_port should be consistent with ICEman's, then click Burn, as shown below:

🔝 Telink Jtag Burn											- 0 ×
Flashing Drivers and IC	EMan				2						
	Jtag_Burn utility Path C:\TelinkloTStu	dio\RDS\V3.2.3\flash\Jtag_bu	ırn.exe					Select Jtag_burn			~
	ICEMan exe/ELF !linkloTStudio\F	RDS\V5.1.2\ice\ICEman.exe				Select ICEMa	n	MCU Archi	tecture: v5		~
Flashing Image (Pinan	and Other ICEMan arguments										
r src\project\tlsr risc	v\B92\GPIO_Demo\GPIO_Demo.bin		Select image			Programming Start Address	.: 0x0				
Other ICEMan Argume	ents:		3			5 5					
☑ Target managemen	t	ICEMan/SPI_burn Host:	127.0.0.1				Bunner Port: 2354			4	
Target Burn											
Enable target burn	Target burn image:										Select image
Itag hum Driver Argu	mentr. ICEMan Interface cfr	3									
- Jtag_burn briver Argu				=							
Targe	et Chip: B92		Flash Controller	Address:				Misc Arguments:			
Verification	Erase All		Unlock		Lock after	burn		Reset and Run Encr	ypt		
ICEMan Interface C	onfig :		⊻Te	Inet Port: 4	444			5			
Logging, car Itog R	m III (CEMan ==)										
Logging: <== Jtag_Bu	rn ICEMan ==>				^	Start the ICEMan at: 20	23-08-28 10-44-45				<u>^</u>
>> Flash page progra	n at address 0x000800					Args:[C:\TelinkloTStudio\RI	DS\V5.1.2\ice\ICEman.e	xe, -Z, v5, -b, 2354]			
>> Flash page progra	n at address 0x000c00				_	Andes ICEman (OpenOCD)	5.1.1-g3a34fc9 (2022-0	4-22-14:27)			
>> Flash page progra	n at address 0x001000					Burner listens on 2354					
>> Flash page progra	n at address 0x001400					Teinet port: 4444					
>> Flash verify ok at a	ddress 0x000000					Andes AICE-MINI+					
>> Flash verify ok at a	ddress 0x000400					JTAG frequency 10.000 MH	z				
>> Flash verify ok at a	ddress 0x000800					There is 1 core in tap					
>> Flash verify ok at a	ddress 0x000c00					The core #0 listens on 1111					
>> Flash verify ok at a	ddress 0x001000					hardware reset-and-hold si	score on Inde tan) hart	0			
>> Flash verify ok at a	ddress 0x001400					naruware reservanu-noid se	iccess on [nds.tap] nare	0			
>> File download to f	ash success at address 0x00000000: 6	916 bytes				Stopped the ICEMan a	t: 2023-08-28 10:48:04-				
>> MCU reset and run											
SPI_Burn exited with c	ode:0										
					~						~
<					>	<					>
	1 6										
Start ICEMan before bu	rning Start ICEMan Burn										

Figure 13.5: Jtag burn

Jtag_ Burn is located in the \$loTStudio_PATH/RDS/V3.2.3/flash/directory, and its help documents are also in the same directory.

13.3 Telink ICEman GDB Debugging (Recommended)

Before starting debugging, you must ensure that the elf file you want to use is selected, as shown in the figure.



Figure 13.6: Select ELF

Then click the drop-down arrow of the Debug icon on the toolbar and select Debug Configuration.





Double clicking on Telink ICEMan GDB Debugging, and a default debug configuration will be generated. You can choose whether you need an ICEman interface in the Debugger tab, as shown in the figure.

Create, manage, and run configurations		
🖸 🖻 🎭 🗎 🗶 🖻 🏹 🔹	Name: UART Demo	
Verifier tet with the second sec	Name: LART: Queon Marke: Debugger: Startup: Startup: Attual executable: Clubreriver: CUbreriver: Startup: Startup: Startup: OBB port: 1111 Telete port: 444 Burineer port: 2254 Config options: -2: v3 Start CEMan Allocate console for ICEMan CBB session -2: v3 Startup: Startup: Startup: Startup: Atlucate console for ICEMan Allocate console for the televet connection CBB clien: Startup: Startup: Startup: Atlucate console for ICEMan Allocate console for the televet connection CBB clien: Storup: Commandis:	Browse Variables
Filter matched 15 of 15 items		激活 Windows Apply
0		转到"设置"以激活 Windows。 Debug Close



13.4 View peripheral registers through SVD file

Users can select the correct chip type in the SVD tab, and the IDE will find the required SVD file on its own. Users can also customize the path of SVD files, but it is not recommended.

Name: TL_PLATFORM_SDK_721X UART_Demo								
Chip :	TL721X							
Custom file path		Browse						

Figure 13.9: SVD Tab

In the Debug view, users can select the peripherals they want to view through the Peripheral tab.

ið 🕏 🗵 🕹 🕹 i	+ 弐 三 乙 乙 5 〒 5 毎 目 Q 5 キ + 0 + ┗ - > ② 9 米 - 5 三 2 5 日 + 句 + ウ - ウ - > 1 1								
app.c ×	- 8	🗱 Variables 💁 Breakpoin	🕸 Expressio 🛛 🕆 Periphera	al × 🔀 Peripheral 👘 🗖					
1 [©] /********	***************************************			* • • ·					
2 * @file 3 *	app.c	Peripheral	Address Descri	ption					
4 * @brief	This is the source file for <u>Telink</u> RISC-V MCU		0x80140						
5 *	Driver Group		0x80140						
7 * @date	2019		0x80140						
8 *		E MSPI	0x80140						
9 * @par	Copyright (c) 2019, <u>Telink</u> Semiconductor (<u>Shanghai</u>) <u>Co</u> .,		0,001+0						
10 *	licensed under the Anache License, Version 2.0 (the "Lice	No details to display for the	he current selection.						
12 *	you may not use this file except in compliance with the l								
🗳 Console 🕮 Regi	sters 🗈 Problems 📀 Executables 🗟 Debugger Console 🛛 Memory 🗙		📑 🛃 📑	📮 🔩 📲 🔹 🗖 🗖					
Monitors	💠 💥 🞇 🔀 I2C: 0x80140280 🛛 🕂 New Renderings								
• I2C	Register	Addres	s Val	ue					
	▼ # 12C	0x8014	0280						
	iiii l2CSP	0x8014	0280 0x0	0305C1F					
		0x8014	0281 0x0	0003050					
		0x8014	0201 0x0	4000030					
		0x80140		1440000					
		0x80140	0285 080	1440000					
		0x80140	0284 0X0	0014400					
	► ### I2CTRIG	0x80140	0285 0x9	5000144					
	I2CLEN	0x80140	0286 0x0	1950001					
	IN SLAVE_STRECH_CTRL	0x80140	0287 0x8	A019500					
	► III I2C_DATA_BUF0	0x80140	0288 0xA	18A0195					
	► 🔤 I2C_DATA_BUF1	0x80140	0289 0x0	0A18A01					
	► ₩ I2C_DATA_BUF2	0x80140	028A 0x0	000A18A					
	► ₩₩ I2C_DATA_BUF3	0x80140	028B 0x0	10000A1					
	→ ## I2C_BUFCNT	0x80140	028C 0x0	0010000					
	► III I2C STATUS	0x80140	028D 0x1	F000100					
	► IIII I2C IRO STATUS	0x8014	028E 0x5	C1F0001					
		0.0014	0.00	0504500					
	Writable	Smart Insert 1:1:0							

Figure 13.10: View Peripheral

13.5 Breakpoint

At present, TLSR9 series SoC supports up to two hardware breakpoints. When you find that the starting address of your program is 0x20000000, it means that it is running in flash and needs to use hardware breakpoints. Commands such as **step in** or **step over** will use a breakpoint, so users can only customize one breakpoint when debugging, otherwise an exception will occur.

When a cannot access memory at address xx exception occurs during debugging, you can use the info br command to check the number of breakpoints.

14 Independent use of some tools in the IoT Studio

14.1 Converter

The purpose of this tool is to convert the project file format of the original Telink RDS IDE to the current Telink IoT Studio project format.

To use it on the command line, you should switch to the directory of the .cproject file that needs to be converted, and executing \$(converter_path)/TelinkRDS2IDE in this directory.

The directory where the converter is located is the \$loTStudio_PATH/tools/Converter/.

For example:



Figure 14.1: converter cmd



14.2 Jtag Burn

The function of this tool is to burn and write binary files into the TLSR9 chip. Before using it, you need to first use ICEman. The Jtag Burn tool is located in the \$loTStudio_PATH/RDS/V3.2.3/flash/, and the user manual is also in this directory. The following is a example of usage:

```
./Jtag_Burn -- chip B92 -- reset- and- run -- verify -- addr 0 -- image / home/ wang/ UART_Demo.bin --
port 2354 --telnet_port 4444 --unlock
```

The parameter of --port and --telnet_port should be same with the parameter of ICEman's Burner port and telnet port. If you see the running results of ICEman indicating its Burner Port is 2354 and its telnet port is 4444 (as shown in the following figure):



Figure 14.2: ICEman defalut port

In this case, you can run the Jtag_Burn without --port and --telnet_port.

15 Installation and uninstallation of IoTStudio in linux

15.1 Installation

In linux, run the installation program " **Telink_IoT_Studio_**xxxx_**Installer.run** " to install the IoTStudio. You can give it executable permissions and run it. During the execution, the user needs to enter the installation path. Be sure to enter the absolute path, as shown in the figure below.





After the program execution ends, you need to execute the command given at the end of the program to complete the installation.





15.2 Uninstallation

In the IoTStudio installation directory, there is an **uninstall.sh** file that can be used to uninstall IoTStudio. You can give it executable permissions and run it. Users need to pass the path of IoTStudio's .desktop file as a parameter to uninstall.sh. Note that there may be a TelinkIoTStudio.desktop file in the installation directory, do not use it as a parameter. You should use the .desktop file with the version number and you can refer to the following example.

./uninstall.sh /home/wang/IDE2308/TelinkIoTStudio_2023.8.desktop

wang@wang-ASUS:~/IDE2308\$ ls							
artifacts.xml	features	p2	SPI_Burn_chip_para.csv	TelinkICE.desktop	TelinkIoTStudio.desktop	uninstall.sh	
configuration	onfiguration icon.xpm plugins tc32 Telink.ico TelinkIoTStudio.ini version.csv						
doc	install_linux_package.sh	RDS	Telink.cfg	TelinkIoTStudio	tools		
dropins	notice.html	readme	TelinkICE_2023.8.desktop	TelinkIoTStudio 2023.8.desktop	udev		
wang@wang-ASUS	:~/IDE2308\$ chmod a+x uning	stall.sh					
wang@wang-ASUS	:~/IDE2308\$./uninstall.sh	/home/war	ng/IDE2308/TelinkIoTStudio	2023.8.desktop			
Found the laun	cher:/home/wang/IDE2308/Te	linkIoTStu	udio_2023.8.desktop				
Exec: /home/wa	ng/IDE2308/TelinkIoTStudio						
Parse install	dir:/home/wang/IDE2308						
Found the inst	alled file						
Installation p	ath is:/home/wang/IDE2308						
UninstallRemoving launcher							
Removing install dir							
done							
wang@wang-ASUS	wang@wang-ASUS:~/IDE2308\$						

Figure 15.3: Uninstall

16 Docs for AndeSight

We have provided the relevant documents for AndeSight, which include the usage documents for Andes Riscv toolchain and other tools. You can find them in the path below according to your needs.

- \${IoTStudio_Path}/doc/Andes_V323_doc/ : Docs for AndeSight V3.2.3
- \${IoTStudio_Path}/doc/Andes_V512_doc/ : Docs for AndeSight V5.1.2
- \${IoTStudio_Path}/doc/Andes_V532_doc/ : Docs for AndeSight V5.3.2

17 How to compile IoTStudio projects in the command line

If you want to compile IoTStudio projects in the command line, you can refer to the following content.

Firstly, you need to build this project in IoTStudio, and then a folder containing makefiles and compilation results will be generated in the project directory, which is usually in the same directory as .cproject

Then open the shell of the toolchain that matches the project in Telink ->Toolchain Shell or ICEman Shell, The example toolchain in the figure is V5.1.2 GCC10.3 V5F toolchain Shell

tun	٣	Telink Window Help		
; ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;	T	Telink RDS to IDE Converter		O ▼ 🚱 ▼ 🤷 ▼ : 😂 🖉 ▼ : 🗟 🗉 🖷 : ½ ▼ 🖓 ▼ ∜⊃ 라 ↔ → ↔ ▼ 📷
3 7		Telink IoTStudio to Cmake		k ×
	2	Windows BDT (SWS and Jtag)		
ter]	4	Jtag Burn (SPI Burn)		-d / / /d//h.7± 1=dd/h.dz=
_con terl	Concola X III Properties III Programs			
		Projects and Artifacts shortcuts	>	
	Ŧ	Toolchain Shell or ICEMan shell	>	Launch RISCV V5F(D25F) toolchain Shell (RDS 3.2.3 GCC 7.4) for TLSR9
	\bigcirc	Telink Links	>	Launch RISCV V5(N22) toolchain Shell(RDS V3.2.3 GCC 7.4) for TLSR9
	ã	Search on DocSite		Launch RISCV V5F(D25F) toolchain Shell(RDS V5.1.2 GCC 10.3) for TLSR9
	_		_	Launch RISCV V5(N22) toolchain Shell(RDS V5.1.2 GCC 10.3) for TLSR9
				Launch RISCV V5F(D25F) toolchain Shell (RDS 5.3.x GCC 12.2) for TLSR9
				Launch RISCV V5(N22) toolchain Shell (RDS 5.3.x GCC 12.2) for TLSR9
				Launch ICEman Shell (V5.3.x) for TLSR9

Figure 17.1: select toolchain shell

Then jump the directory to the folder containing the makefile and execute make clean make all



Figure 17.2: make in shell

/vendor/common/B92/calibration/calibration.c" ished building: ../../../demo/vendor/common/B92/calibration/calibration.c