Application Note : Guide For Adding New Project On Existing SDK

AN-16063000-E1

Ver 1.0.0

2016/6/30

Brief:

This document is the user guide on how to add a new project on existing SDK via Telink IDE.

SEMICONDUCTOR

TELINK SEMICONDUCTOR



Published by Telink Semiconductor

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

© Telink Semiconductor All Right Reserved

Legal Disclaimer

Telink Semiconductor reserves the right to make changes without further notice to any products herein to improve reliability, function or design. Telink Semiconductor disclaims any and all liability for any errors, inaccuracies or incompleteness contained herein or in any other disclosure relating to any product.

Telink Semiconductor does not assume any liability arising out of the application or use of any product or circuit described herein; neither does it convey any license under its patent rights, nor the rights of others

The products shown herein are not designed for use in medical, life-saving, or life-sustaining applications. Customers using or selling Telink Semiconductor products not expressly indicated for use in such applications do so entirely at their own risk and agree to fully indemnify Telink Semiconductor for any damages arising or resulting from such use or sale.

Information:

For further information on the technology, product and business term, please contact Telink Semiconductor Company (<u>www.telink-semi.com</u>).

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

telinkcnsales@telink-semi.com

telinkcnsupport@telink-semi.com



Revision History

Version	Major Changes	Date	Author
1.0.0	Initial release	2016/6	W.S.H., Cynthia



Table of contents

1	Brief Introduction	5
2	Create A New Configuration "8267_ble_demo"	6
3	Modify Critical Compiling Parameters For "8267_ble_demo"	9
4	Copy Working File To Vendor	17
5	Add "app_config.h" To SDK	18
6	Replace Old Compiling Symbols By "8267_BLE_DEMO"	19
7	Configure Compiling Exclusion	20
8	Compile New Project	24
9	Write Your Own Code	25

•



Table of figures

Figure 1	Existing projects and corresponding working files5
Figure 2	"Properties for ble_sdk_lt_release" window6
Figure 3	"ble_sdk_lt_release: Manage Configurations" window7
Figure 4	"Create New Configuration" window7
Figure 5	Return to "ble_sdk_lt_release: Manage Configurations" window8
Figure 6	New configuration "8267_ble_demo"8
Figure 7	Set "Configuration" as "8267_ble_demo"9
Figure 8	Set bootloader compiling configurations10
Figure 9	"-DMCU_CORE_8267" flag in cstartup_8267.S11
Figure 10	Set project compiling symbol12
Figure 11	Choose lib13
Figure 12	Rename compiling result list file14
Figure 13	Rename compiling result binary file15
Figure 14	Confirm and apply parameter modifications16
Figure 15	Copy working file for new project17
Figure 16	New working file "8267_ble_demo" under "vendor"17
Figure 17	Add "8267_ble_demo/app_config.h" to SDK18
Figure 18	Modify old compiling symbols19
Figure 19	Open "Exclude from build" window for "8267_ble_demo" directory
	20
Figure 20	Configure compiling exclusion for "8267_ble_demo"21
Figure 21	Open "Exclude from build" window for "8267_ble_remote"
direc	22
Figure 22	Configure compiling exclusion for "8267_ble_remote"23
Figure 23	Compiling result for new project24



1 Brief Introduction

This document presents how to add a new project (e.g. "8267_ble_demo") on existing SDK (e.g. Telink 8267 BLE SDK "ble_sdk_lt_release") by using Telink IDE based on Eclipse platform.

New project is not created by clicking File/New menu from the very beginning. Actually it's copied from the existing demo project, but critical settings and configurations should be modified for the new project.

As shown in Figure 1, click the hammer icon, there are 5 existing projects that can be compiled, and corresponding working files are contained under the "vendor" directory. In the demonstration, we need to add a new project "8267_ble_demo" by copying from an existing project "8267_ble_remote", and add its working files under the "vendor" directory.



Figure 1 Existing projects and corresponding working files





2 Create A New Configuration "8267_ble_demo"

Click "Project" menu -> "Properties" (or right click on the Project Explorer window and select "Properties") to open the "Properties for ble_sdk_lt_release (i.e. SDK name)" window.

As shown in Figure 2, select "C/C++ Bulid"->"Settings" on the left side to open the "Settings" option, then click the "Manage Configurations" button to open the "ble_sdk_lt_release: Manage Configurations" window.



Figure 2 "Properties for ble_sdk_lt_release" window



As shown in Figure 3, configurations corresponding to the existing 5 projects are available on the window. The "Description" column is not essential and it just shows details about the configuration. Click the "New" button to open the "Create New Configuration" window.

	ble_sdk_lt_release: Manage Co	nfigurations		×	J	
	Configuration	Description	Status			
	8261_ble_remote					
	8266_ble_master_kma_dongle	ble master dongle fo				
	8267_beacon	duplicated from ble_r				
	8267_ble_master_kma_dongle					
	8267_ble_remote	remote with MIC	Active			
8	Set Active Ne	w Delete	,	Rename		
			ОК	Cancel		

Figure 3 "ble_sdk_lt_release: Manage Configurations" window

As shown in Figure 4, set the "Name" as any new configuration name (e.g. "8267_ble_demo"); for "Copy settings from" option, select "Existing configuration" -> "8267_ble_remote". For the "Description", you can add simple details or just skip it. Click "OK" to return to the "ble_sdk_lt_release: Manage Configurations" window.

Name: <u>8267_ble_de</u>	mo
Description:	
Copy settings from	
Existing configuration	8267_ble_remote(remote with MIC)
Default configuration	8267_ble_remote(remote with MIC) 8266_ble_master_kma_dongle(ble master dongle for keyboard/mouse/audio)
Import from projects	8267_beacon(duplicated from ble_remote) 8267_ble_master_kma_dongle
Import predefined	8261_ble_remote

Figure 4 "Create New Configuration" window



As shown in Figure 5, the new configuration "8267_ble_demo" is available on the "ble_sdk_lt_release: Manage Configurations" window. Click "OK" to return to the "Properties for ble_sdk_lt_release" window.

ble_sdk_lt_release: Manage Cor	figurations		
Configuration	Description	Status	
8261_ble_remote			
8266_ble_master_kma_dongle	ble master dongle for k		
8267_beacon	duplicated from ble_re		
8267_ble_demo			
8267_ble_master_kma_dongle			
8267_ble_remote	remote with MIC	Active	
Set Active New	Delete	Rename	
	ОК	Cancel	

Figure 5 Return to "ble_sdk_lt_release: Manage Configurations" window

Then click "OK" to close the "Properties for ble_sdk_lt_release" window.

As shown in Figure 6, the new configuration "8267_ble_demo" is available when clicking the hammer icon.



Figure 6 New configuration "8267_ble_demo"



3 Modify Critical Compiling Parameters For "8267_ble_demo"

After the new configuration "8267_ble_demo" is created as shown in **Section 2**, since all compiling parameters are copied from the selected source project "8267_ble_remote", it's need to modify all critical compiling parameters for the new project.

Open the "Properties for ble_sdk_lt_release" window again, select "C/C++ Bulid" -> "Settings", and then set the "Configuration" option as "8267_ble_demo".

Properties for ble_sdk_lt_rel	ease				
type filter text	Settings				¢
Resource			_		
Builders C/C++ Build	Configuration:	8267_ble_demo		,	Manage Confi
Build Variables Discovery Options	ND To al Carrier	8261_ble_remote 8266_ble_master_kma_dongle 8267_beacon	5		
Environment Logging	Additio	8267_ble_demo 8267_ble_master_kma_dongle 8267_ble_rameta_[Active]			
Tool Chain Editor C/C++ General	ல TC32 0 20 Ger	[All configurations] [Multiple configurations]			
Project References	🖄 Pat 🔊 Del	hs			

Figure 7 Set "Configuration" as "8267_ble_demo"



Please follow the steps below to modify relevant parameters.

1) Set bootloader compiling configurations.

As shown in Figure 8, select "TC32 CC/Assembler" -> "General" -> "Other GCC Flags". It's needed to define "-DMCU_CORE_8267" and delete the following "-D_LOAD_RAM_SIZE__=0x16".

ilter text	Settings	
esource iilders C++ Build Build Variables Discovery Options Environment	Configuration 8267_ble_demo	▼ Manag
Settings	Additional Tools in Toolchain Suppress warnings (-W)	
Tool Chain Editor	TC32 CC/Assembler (-Xassembler) options	ą
oject References in/Debug Settings isk Repository link Tools ikiText	 Paths Debugging S TC32 Compiler Symbols Warnings Debugging Optimization Language Standard Miscellaneous S TC32 C Linker General S TC32 Create Extended Listing General S TC32 Create Flash image General S TC32 Create Flash image General S Trisze General 	
	Other GCC Flags -DMCU CORE 8267 -D LOAD RAM SIZE =0x16	

Figure 8 Set bootloader compiling configurations



Actually, the "-DMCU_CORE_8267" flag is used in cstartup_8267.S for

bootloader compiling, as shown in Figure 9.



Figure 9 "-DMCU_CORE_8267" flag in cstartup_8267.S



2) Set project compiling symbol.

As shown in Figure 10, select "TC32 Compiler" -> "Symbols".

User needs to click the "Edit" icon, and set the compiling symbol as

"___8267_BLE_DEMO___" to replace the old "___PROJECT_BLE_REMOTE___".

Then click "OK" to confirm the setting.

Properties for ble_sdk_lt_rel	ease	
type filter text	Settings	$\varphi \star \varphi$
Resource Builders C/C++ Build Build Variables Discovery Options Environment Logging Settings Tool Chain Editor C/C++ General Project References Run/Debug Settings Task Repository Telink Tools WikiText	Configuration 8267_ble_demo Tool Settings Build Steps 2 E Additional Tools in Toolchain TC32 CC/Assembler General Paths Debugging TC32 Compiler Directories Symbols Warnings Dobugging Optimization Language Standard Miscellaneous TC32 C Linker	Wanage Configurations

Figure 10 Set project compiling symbol





3) Choose the lib.

As shown in Figure 11, select "TC32 C Linker" -> "Libraries". The current library is "lt_8267" which is used for 8267 16M system clock application. Since 16M system clock applies to the new project, it's not needed to change the lib.

If 32M or 48M system clock applies to your application, just click the "Edit" icon to set the lib as "It 8267 32m" or "It 8267 48m" correspondingly.



Figure 11 Choose lib



4) Rename the compiling result list file.

As shown in Figure 12, select "TC32 Create External Listing" -> "General".

The current name for the compiling result output list file is

"8267_remote.lst". User needs to rename it as "8267_demo.lst".



Figure 12 Rename compiling result list file



5) Rename the compiling result binary file.

As shown in Figure 13, select "TC32 Create Flash image" -> "General".

The current name for the compiling result output binary file is

"8267_remote.bin". User needs to rename it as "8267_demo.bin".



Figure 13 Rename compiling result binary file



After all the parameters are modified as shown above, user also needs to click "Apply" and "OK" (as shown in Figure 14) so that all the modifications will take effect.



Figure 14 Confirm and apply parameter modifications



4 Copy Working File To Vendor

Copy the whole folder "8267_ble_remote" under the "vendor" directory, and rename it as "8267_ble_demo".

The new project will first use the same code copied from the 8267_ble_remote to pass the compiling. Then user can write his own code for the new project to replace the old code.

修改日期
2016/6/29 17:16
2016/6/29 17:16
2016/6/29 17:16
2016/6/29 17:16
2016/6/29 17:16
2016/6/29 20:15
2016/6/29 17:16



After the SDK is refreshed, the new working file "8267_ble_demo" is contained

under the "vendor" directory, as shown Figure 16.

€ C/	·C++ -	ble_sdk_l	t_release/p	proj/mcu_sp	ec/cstart	up_820	o7.5 - Eclip	se		
File	Edit	Source	Refactor	Navigate	Search	Proje	ect 🛭 🏶 Te	link Tools	Run	Wind
	•	6	🗟 💋	× 💼	- 🚳 -	C .	· 🞯 -		•	☆ -
6 F	roject	Explorer	x			🔝 cst	tartup_826	7.S 🕱 🔪	_	
4 5	S ble	sdk lt re	ease	E \$	69 ▽	1 2	#ifdef	MCU_CORI	2_8267	,
		Binaries	remote			3	#ifndef	LOAD_	RAM_S	IZE_
		8266_ble_ 8266_ble_ 8267_bce	master_km	a_dongle		5	#define #endif	ELOAD	_RAM_S	SIZE_
		8267_bea 8267_ble_ 8267_ble	con master_km	a_dongle		8	#ifndef #define	IRQ_	STK_SI STK_SI	ZE
		8207_bie_ ble :	remote			10 11	#endif .co	de 16		
	▷ (2⇒ ▷ (2⇒	proj proj_lib				12 13 14	@***** @ . @*****	******	*****	****
	• <u>€</u> ⊳[> 8267_l	ole_demo	>		15	includ	le "vers:	ion.in	. "
		🔁 comm	on			17 18	.eq	u MODE_I	BITS,	@ Mo
		→ 8261_I → 8266_I	ble_remote ble_master	_kma_dongl	e	19 20 21	.eq	ru IRQ_MO ru SVC_MO	DDE, DDE, TK ST7	T
		≥ 8267_I ≥ 8267_I	beacon ble_master	_kma_dongl	e	22	.eq	u _LOAI	C_RAM,	,

Figure 16 New working file "8267_ble_demo" under "vendor"

AN-16063000-E1



5 Add "app_config.h" To SDK

As shown in Figure 17, open vendor -> common -> user_config.h on the "Project Explorer" window.

Click the hammer icon and select the "8267_ble_demo" project, its status will turn to "Active" automatically. Correspondingly the macro "__8267_BLE_DEMO__" is enabled and "vendor/8267_ble_demo/app_config.h" is included by the SDK, while "app_config.h" under other projects are excluded.

#elif (__PROJECT_BLE_REMOTE__)

#include "../8267_ble_remote/app_config.h"

#elif (___8267_BLE_DEMO___)

#include "../8267 ble demo/app config.h"

File Edit Source Refactor Navigate Search Project 🏶 Telink Tools Run Window Help



Figure 17 Add "8267_ble_demo/app_config.h" to SDK



6 Replace Old Compiling Symbols By "__8267_BLE_DEMO__"

In some files of the new project "8267_ble_demo", old compiling symbol "___PROJECT_BLE_REMOTE__" may still exist, which will cause all the files are copied from the old project "8267_ble_remote".

User needs to use global search to find all old compiling symbols and replace them by "__8267_BLE_DEMO__".

As shown in Figure 18, it's needed to replace the two old symbols in main.c and ble_remote.c by the "__8267_BLE_DEMO__".



Figure 18 Modify old compiling symbols



7 Configure Compiling Exclusion

In order to eliminate the interference from other compiling projects, user needs to exclude the files under the "8267_ble_demo" directory from other projects.

As shown in Figure 19, select the "8267_ble_demo" directory, right click the directory and select "Resource Configurations" -> "Exclude from Build" to open the "Exclude from build" window.



Figure 19 Open "Exclude from build" window for "8267_ble_demo" directory



As shown in Figure 20, tick projects other than the new project "8267_ble_demo", and then click "OK". The setting will exclude files under the new project directory from other projects. Thus the compiler can't see all files under the "8267_ble_demo" directory when compiling other projects.

Exclude from build	
Exclude object(s) from build in the following configurations	
 8267_ble_remote 8266_ble_master_kma_dongle 	
 8267_beacon 8267_ble_master_kma_dongle 	
 ✓ 8261_ble_remote ☑ 8267 ble demo 	
Select All Decelect All	
OK Cancel	

Figure 20 Configure compiling exclusion for "8267_ble_demo"



Since the "8267_ble_demo" project is copied from the "8267_ble_remote", it's also needed to exclude the "8267_ble_demo" in the "8267_ble_remote".

As shown in Figure 21, select the "8267_ble_remote" directory, right click the directory and select "Resource Configurations" -> "Exclude from Build" to open the "Exclude from build" window.



Figure 21 Open "Exclude from build" window for "8267_ble_remote" directory



As shown in Figure 22, since the "8267_ble_remote" is only excluded from other 4 projects, user needs to tick the "8267_ble_demo" and click "OK".

Exclude from bui	d		×
Exclude object(s) f	om build in the follow	wing configuratio	ons
8267_ble_rem	iote		
✓ 8266_ble_mas	ter_kma_dongle		
✓ 8267_beacon			
▼ 8267_ble_mas	ter_kma_dongle		
▼ 8261_ble_rem	iote		
8267_ble_der	no		
Tick "8267 b	le demo"		
		Decelect All	
	Select All	Desciect All	
?	ОК	Cancel	

Figure 22 Configure compiling exclusion for "8267_ble_remote"



8 Compile New Project

After all the settings above are done, user can carry out compiling for the new project.

Figure 23 shows the compiling result for the new project "8267_ble_demo".

名称	修改日期	类
퉬 8261_ble_remote	2016/6/29 17:16	Ŕ
퉬 8266_ble_master_kma_dongle	2016/6/29 17:16	3
퉬 8267_beacon	2016/6/29 17:16	3
퉬 8267_ble_demo	2016/6/29 21:18	Ż
🐌 8267_ble_master_kma_dongle	2016/6/29 17:16	3
8267_ble_remote	2016/6/29 17:20	3
📔 ble	2016/6/29 17:16	3
Diroi	2016/6/20 17:16	7
磁盘 (E:)) 8267_SDK) ble_sdk_lt_release	▶ 8267_ble_demo ▶	
		_
浸▼ 新建文件夹		
名称	修改日期	
퉬 ble	2016/6/29 21:16 文件夹	
🌗 proj	2016/6/29 21:15 文件夹	
🍌 vendor	2016/6/29 21:15 文件夹	
8267_demo.bin	2016/6/29 21:18 BIN 文件	
8267_demo.lst	2016/6/29 21:18 LST 文件	
ble_sok_it_release.elf	2016/6/29 21:18 ELF 文件	
aiv_mod.o	2010/0/29 21:15 0 文件	
objects mk	2010/0/29 21:10 又件 2016/6/29 21:18 MK 文件	
sources.mk	2016/6/29 21:18 MK 文件	
subdir.mk	2016/6/29 21:18 MK 文件	



9 Write Your Own Code

After the new project passes compiling, user can write his own code under the "8267_ble_demo" directory. New head files and C files can also be added under this directory.