

知从木牛配置工具产品手册

ZC.MUNIU CONFIGURATION TOOL PRODUCT MANUAL

知从木牛基础软件平台 ZC.MuNiu Basic Software Platform





知从木牛配置工具产品手册

ZC.MUNIU CONFIGURATION TOOL PRODUCT MANUAL

知从木牛基础软件平台

ZC.MuNiu Basic Software Platform

1 功能概述 FUNCTIONAL OVERVIEW

知从木牛配置工具基于最新的 ARTOP 架构开发,支持最新的 AUTOSAR R21-11 标准的 基础软件配置工具,提供全套的 BSW 配置和 RTE 生成等功能。目前知从 AUTOSAR 标准的 BSW、复杂驱动、SafetyFrame、CryptoLibrary 等基础软件产品都采用该配置工具。该方案兼 容业界主流的标准,包括系统功能设计,ECU 功能映射和组件配置,运行时环境自动代码生 成等一系列的工具套件,提供了一套经过实践验证的、开发可重用 ECU 应用软件的基础,可 以同 Vector,EB 等工具交叉配合使用。

ZC.MuNiu configuration tool is developed based on the latest ARTOP architecture and supports the latest AUTOSAR 4.4.0 standard for basic software configuration tools, providing a full set of BSW configuration and RTE generation functions. Currently, ZC Technology's AUTOSAR standardized BSW, complex drivers, SafetyLibrary, SafetyFrame, CryptoLibrary, and other basic software products all use this configuration tool. The solution is compatible with mainstream industry standards, including system functional design, ECU function mapping and component configuration, and a suite of tools for automatic code generation in the runtime environment. It provides a set of practical and verified foundations for developing reusable ECU application software, which can be used in conjunction with tools from Vector, EB, and others.

M科技		<mark>知从科</mark> Easy to l	技易 know Eas
😽 workspace - Classic Designer -	ZC MuNiu v4.4_07SWMID010		
File Edit Navigate Search Pr	ject Model Form Editor Validate Run Window Help		
📬 🗝 📓 🕼 🕼 🖬 🧣			Quick Access
Configuration Editors	🗝 🗖 🐺 Basic Editor 🔀		- 8
text to filter			
Base Services	× Models		
Communication	V Can		
Diagnostics	😵 🦊 Canlf		
🐺 <u>Basic Editor</u>	CanTp CanTrov Com Com Com Com Com Com Com Com		
Properties 🔀	😑 📋 🦧 AUTOSAR Validation 🧟 Tasks 🦹 Problems 📮 Console 🕏 Navigator 🕐 BSW Para	ameter Help	
t≣‡≯ II.	ng マ 0 errors 0 warnings 0 infos	Configur	e Filters 🗸
roperty value	Description	Ођ	ect

知从木牛配置工具产品特点:

Features of ZC.MuNiu Configuration Tool Product:

基于 Eclipse 平台框架,采用插件化和模块化的开发思想,具有跨平台运行,组件化 \geq 设计.良好的可扩展性。

Based on the Eclipse platform framework, it adopts the development philosophy of plugin-based and modular design, offering cross-platform operation, componentbased architecture, and excellent scalability.

- \geq 提供完整的集成开发环境,支持智能代码编辑、编译、链接和调试功能。 Provides a complete integrated development environment, supporting intelligent code editing, compilation, linking, and debugging functions.
- ▶ 支持 AUTOSAR 开发方法和过程,提供全套 BSW 配置和 RTE 生成等工具。 Supports the AUTOSAR development methodology and process, offering a full set of BSW configuration and RTE generation tools.
- \geq 基于最新 ARTOP 架构开发,支持最新 AUTOSAR R21-11 标准。 Developed based on the latest ARTOP architecture, it supports the latest AUTOSAR 4.4.0 standard.



▶ 支持 BSW 一致性和完整性验证。

Supports BSW consistency and integrity verification.

▶ 支持 DBC、LDF、ODX、DEXT、OIL 格式文件导入。

Supports the import of DBC and ODX format files.

▶ 支持各种主流的汽车 ECU 芯片,如下表:

Compatible with various mainstream automotive ECU chips,

	主流芯片支持							
Ma	Mainstream Automotive ECU Chips							
序号	芯片型号	厂家						
No.	Model	Manufacturer						
No.1	S32K1XX	NXP						
No.2	S32K3XX	NXP						
No.3	MPC5744P	NXP						
No.4	MPC5746C	NXP						
No.5	MPC5748G	NXP						
No.6	TC23X	Infineon						
No.7	TC27X	Infineon						
No.8	TC37X	Infineon						
No.9	TC39X	Infineon						
No.10	TMS320f280048	ΤI						
No.11	AWR1642	TI						
No.12	SPC58NN	ST						
No.13	StellaE1	ST						
No.14	KF32A152	ChipOn						
No.15	FC4150	Flagchip						
No.16	AC7801x	Autochips						
No.17	AC781x	Autochips						



2 应用领域 APPLICATION FIELD

知从木牛配置工具,给 ECU 控制器软件开发提供友好的人机界面。可以支持标准的 AUTOSAR 基础软件代码模块的配置,以及复杂驱动的配置界面开发。目前主要应用于如下场 景:

ZC.MuNiu Configuration Tool offers a user-friendly interface for the development of ECU (Electronic Control Unit) controller software. It supports the configuration of standardized AUTOSAR basic software code modules and the development of configuration interfaces for complex drivers. It is currently mainly applied in the following scenarios:



> 知从木牛基础软件平台标准 AUTOSAR 模块配置

ZC.MuNiu Basic Software Platform Standard AUTOSAR Module Configuration

> 知从木牛基础软件平台复杂驱动模块配置

ZC.MuNiu Basic Software Platform Complex Driver Module Configuration

- ♦ SAFETY FRAME
- CRYPTO LIBRARY
- ♦ BCCIC
- SBC
- ▶ 同芯片企业合作,提供 MCU MCAL 的配置工具

Collaborate with chip companies to provide configuration tools for MCU MCAL.



3 开发背景 DEVELOPMENT BACKGROUND

随着汽车工业的飞速发展,汽车电子已经成为汽车技术发展过程中的革命性标志。为了 应对汽车电子软件开发中不断提高的系统复杂程度,改善软件的一致性和重用性,AUTOSAR 标准在 2003 年被提出并且迅速在产业内取得了广泛认同。在业内的开发工具提供商纷纷推出 了符合 AUTOSAR 标准的开发工具链和解决方案后,部分开发商成立了 Artop 组织,解决不同 厂商的开发工具之间的数据交互和 AUTOSAR 元模型的多版本的问题,旨在更好的推进 AUTOSAR 开发方法在汽车电子软件开发中的实现。

With the rapid development of the automotive industry, automotive electronics have become a revolutionary sign in the process of automotive technology development. To cope with the increasing system complexity in automotive electronic software development, improve software consistency and reusability, the AUTOSAR standard was proposed in 2003 and quickly gained widespread recognition in the industry. After the development tool providers in the industry launched development tool chains and solutions that conform to the AUTOSAR standard, some developers established the ARTOP organization to solve the problem of data interaction between development tools from different manufacturers and the multi-version issues of the AUTOSAR meta-model, aiming to better promote the implementation of the AUTOSAR development method in the development of automotive electronic software.

其中,一些国际知名公司基于自己原先的开发工具产品提供了面向 AUTOSAR 开发工具链, 在业内首先推出了 Artop 的架构的 AUTOSAR Builder。这些开发工具集中都包含了 ECU 配置的 功能,但是,这些 ECU 配置的功能大部分都仅针对这些公司的基础软件模块实现提供支持, 并不能够支持其他公司的基础软件实现的配置和代码生成。

Among them, some internationally renowned companies have provided AUTOSAR development tool chains based on their original development tool products, and were the first in the industry to launch the Artop architecture of AUTOSAR Builder. These development tools all include the function of ECU configuration, but most of these ECU configuration functions only provide support for the implementation of the company's own basic software modules, and cannot support the configuration and code generation of the basic software implemented by other companies.



知从科技易知简从 Easy to know Easy to do

与国际大公司的解决方案相比,国内符合 AUTOSAR 开发方法的软件开发平台和 ECU 配置 工具非常少,基础软件的开发能力也很薄弱。因此,研发和实现一套符合 AUTOSAR 标准开发 流程的开发工具,提供一个支持多厂商多平台的基础软件实现的 ECU 配置环境,对于我国的 汽车电子软件发展具有重要的意义。

Compared with the solutions of international giants, there are very few domestic software development platforms and ECU configuration tools that conform to the AUTOSAR development method, and the development capability of basic software is also very weak. Therefore, developing and implementing a set of development tools that conform to the AUTOSAR standard development process, providing an ECU configuration environment that supports the implementation of basic software from multiple manufacturers and platforms, is of great significance to the development of China's automotive electronic software.

知从木牛配置工具是符合国产汽车电子系统架构标准的 ECU 开发解决方案。该方案兼容 业界主流的 AUTOSAR 标准,包括从系统功能设计,ECU 功能映射和组件配置,运行时环境自 动代码生成等一系列的工具套件,提供了一套经过实践验证的,开发可重用 ECU 应用软件的 基础。

ZC.MuNiu configuration tool is an ECU development solution that conforms to the domestic automotive electronic system architecture standard. The solution is compatible with the mainstream AUTOSAR standard in the industry, including a series of tools from system function design, ECU function mapping and component configuration, to automatic code generation of the runtime environment, providing a set of practical and reusable ECU application software development foundation.



4 功能描述 FUNCTIONAL DESCRIPTION

知从木牛配置工具基于最新 ARTOP 架构,支持最新 AUTOSAR R21-11 标准所提供的基础 平台上,根据 AUTOSAR 开发方法中定义的 ECU 配置步骤,实现了从配置、验证到代码生成的 ECU 配置全流程的功能。主要优势可以总结为以下几个方面:

ZC.MuNiu configuration tool is based on the latest ARTOP architecture and supports the latest AUTOSAR 4.4.0 standard. It has realized the full process of ECU configuration from configuration, verification to code generation according to the ECU configuration steps defined in the AUTOSAR development method. The main advantages can be summarized in the following aspects:

配置、验证和代码生成全流程功能的实现,完整的实现了 AUTOSAR 开发方法中 ECU 配置 阶段的开发要求。

The implementation of the full process of configuration, verification, and code generation fully realizes the development requirements of the ECU configuration phase in the AUTOSAR development method.

支持厂商实现有关的模块信息开插拔易扩展的系统的设计与实现。不同厂商对于基础模块的实现都可以非常方便的集成到知从木牛配置工具中的 BSW Builder 中来,在 BSW Builder 启动过程中动态加载进来,具有较好的可扩展性。

It supports the design and implementation of a system that is easy to expand with pluggable module information related to the manufacturer's implementation. Different manufacturers can easily integrate the implementation of the basic module into the BSW Builder in ZC.MuNiu configuration tool, which is dynamically loaded during the startup process of the BSW Builder and has good scalability.

基于查看器框架的配置界面动态生成方法的实现。基于 EMF 框架基础功能, BSW Builder 将视图层与模型层进行了解耦,并且设计和实现了 BSW Builder 查看器,将 BSW Builder 参数 定义模型和配置值模型相结合,动态生成配置界面。



知从科技 易知简从 Easy to know Easy to do

The implementation of the dynamic generation method of the configuration interface based on the viewer framework. Based on the basic functions of the EMF framework, the BSW Builder decouples the view layer from the model layer and designs and implements the BSW Builder viewer, combining the BSW Builder parameter definition model and configuration value model to dynamically generate the configuration interface.

4.1 产品特性 Product Feature

知从木牛配置工具集成开发环境特点如下:

ZC.MuNiu configuration tool integrates a development environment with the following features:

基于 Eclipse 平台和 Artop 架构, 插件化开发, 可扩展性强
 Based on the Eclipse platform and Artop architecture, it is developed in a plug-in manner, offering strong extensibility.

▶ 支持对 AUTOSAR R21-11 标准各个模块的配置

Supports configuration for various modules of the AUTOSAR 4.4.0 standard.

▶ 支持复杂驱动配置工具定制开发

Supports customized development of complex driver configuration tools.

▶ 支持配置数据的一致性校验

Supports consistency verification of configuration data.

▶ 产品级的自动代码生成

Product-level automatic code generation.

▶ 支持 ARXML 格式数据交换

Supports data exchange in ARXML format.

▶ 可以根据客户需要灵活配置和裁减功能组件

Flexible configuration and trimming of functional components according to customer needs.

▶ 强大的错误检查和提示纠正机制

Powerful error checking and prompt correction mechanism.



▶ 支持多种格式数据文件的导入

Supports the import of various format data files.

4.2 技术架构 Technical Architecture

知从木牛配置工具基于 AUTOSAR 配置工具最新架构开发。

ZC.MuNiu configuration tool is developed based on the latest architecture of the AUTOSAR configuration tool.

- Eclipse Oxygen.3a Release (4.7.3a)
- > ARTOP-SDK-4.13.0 (supports the latest AUTOSAR releases R21-11)
- ➢ ARTEXT-SDK-1.7.2 1.6.0.
- ➢ CDT 8.1.2



木牛配置工具架构图 ZC.MUNIU CONFIGURATION TOOL ARCHITECTURE DIAGRAM

4.3 运行环境 Run Environment



5 使用介绍 INTRODUCTION TO USE

5.1 新建工程 Create a New Project

"File"菜单中选择"New"--->"Project" ---> "AUTOSAR" ---> "AUTOSAR Project"可以打 开如下界面,进行创建工程。

Select "New" in the "File" menu, then choose "Project" ---> "AUTOSAR" ---> "AUTOSAR Project" to open the following interface for project creation.

luNiuNew\S32K_MuNiu\workspace\S	Browse
Configure Works AUTOSAR 4.4.0 (http://autosar.org/scl	p <mark>ace Settings</mark> nema/r4.0)
	luNiuNew\S32K_MuNiu\workspace\S: Configure Works AUTOSAR 4.4.0 (http://autosar.org/scl

新建工程 CREATE A NEW PROJECT

5.2 选择平台 Select Platform

点击 next, 可以打开如下界面, 可以选择需要配置的平台。

Click "Next," and you can open the following interface to select the platform that needs to be configured.

View AUTOSAR Project		
latforms		
Please select platform.		
The following platforms:		
✓ Infineon		
TC275		
TC397		
TC23x		
TC377		
TC264		
✓ AutoChips		
AC7801x		
AC781x		
✓ ChipON		
KF32A152		
✓ ST		
SPC58NN		
✓ NXP		
MPC5744P		
MPC5746C		
MPC5748G		
S32K1xx		
S32K3xx		
✓ Renesas		
RH850P1M-C		
F1KM		
RH850U2A16		
✓ TI		
TMS320f280048		

创建平台 CREATE PLATFORM



在软件主界面中点击"Basic Editor"可以看到该平台下所包含的模块。

In the main interface of the software, click on "Basic Editor" to view the modules included under this platform.

在模块上面右键,选择"add"即可添加模块。

Right-click on a module and select "Add" to add a new module.



打开选择模块界面 OPEN THE MODULE SELECTION INTERFACE

界面中列出的都是该平台下支持的模块,根据需要进行选择。

In the interface, all modules supported under the platform are listed for selection based on your needs.

点击模块左边的箭头,即可看到该模包含的配置结构,在需要配置的地方右键进行添加,并单击该配置,即可看到每一个配置项。

By clicking on the arrow to the left of the module, you can view the configuration structure it contains. To add configurations where needed, right-click and select 'Add'. After clicking on a specific configuration, you can see each configuration item.





₩ workspace - Classic Designer - ZC MuNi	u v4.4_C –	
File Edit Navigate Search Project Me	odel Form Editor Validate Run Window Help	
📑 🕶 🔚 😭 🆃 🖬 🗣 🗸 🗸 🗸	堕・型・ひゃり・	Quick Access
🐺 Configuration Editors 🛛 🗖 🗆	🐨 *Basic Editor 🛛	- 8
text to filter		
Base Services ¥	Models ^ BswM	
Communication ¥	🐈 Can	
Diagnostics ¥	₩ Canif	
	T CanNm	
	✓ th CanTp	
	V 🐈 CanTp	
	□ CanTpGeneral [11] □ Add CanTpGeneral	
	Com	
	🐈 ComM	
	Pcm	
	••• Dem	
Basic Editor		
🗖 Properties 🛛 🗖 🗖	🥵 AUTOSAR Validation 🧔 Tasks 🖹 Problems 📮 Console 😘 Navigator 🛿 🕜 BSW Parameter Help	- 8
🌐 🎰 🔤 📑 🎽		□ 🗣 🔻
Property Value		

添加配置 ADD CONFIGURATION

😽 workspace - Classic Designer - ZC M	uNiu v4.4_(-		×
File Edit Navigate Search Project	Model Form Editor Validate Run Window	Help			
🖻 🝷 🔛 🔞 🕸 🖬 💀 💁 🔹 🛷	• 🖗 • 🖗 • 🍄 • ⇔ •			Quick	Access
🐺 Configuration Editors 👘	🗆 🐺 *Basic Editor 🛛				
text to filter		CanTpGeneral_0 [/AUTOSAR/CanTp/CanTpGeneral]			
Base Services	× ₩ BswM			`	^
Communication	¥ 🔮 Can	- Name	CanTpGeneral_0		
Diagnostics		CanTpDevErrorDetect	FALSE	~	
	CanSM CanTp	∑p CanTpReadParameterApi	FALSE	~	
	 ✓	🔀 CanTpChangeParameterApi	FALSE	~	
		🔀 CanTpVersionInfoApi	FALSE	~	
Basic Editor	♥ CanTrcv ♥ Com	CanTpDynIdSupport	-	~	
Properties 🛛 🗖	ComM	CanTpFlexibleDataRateSupport	-	~	
Property Value	✓ ¹ Dem ¹ EcuC ✓	CanTpGenericConnectionSupport	-	~	<i>,</i>
	🚜 AUTOSAR Validation 🖉 Tasks 🖹 Pro	oblems 🖳 Console 🛛		Ŭ ▼ '	
	No consoles to display at this time.				
	<u></u>				
+	+1				
			1		

配置项 CONFIGURATION ITEM



5.4 校验生成 Verification and Generation

针对完成配置的模块可以进行校验,以检查配置是否合理,有问题的配置项将会获得 提示。

For modules that have been configured, you can perform verification to check if the configuration is reasonable. Any problematic configuration items will be highlighted with prompts.

₩ workspace -	Classic Designer - ZC MuNit	1					-		×
File Edit Navi	igate Search Project Mo	del Form Editor Validate Ru	un Window	Help					
	🕸 💽 🚱 🗣 🛷 🕶	월 • 월 • 🏷 🔶 • 🔿 •						Quick	Access
Configuration	Editors 🗖 🗖	🐺 *Basic Editor 🛛							
text to filter				CanTpGeneral_0 [/AUT)SAR/CanTp/CanTpGeneral]				
Base Service	s ¥	Models	^						^
Communicat	ion ¥	🐈 Can		- Name		CanTpG	eneral_0		
Diagnostics	*	₩ Canlf ₩ CanNm		🔀 CanTpDevError	Detect	FALSE		\sim	
		Units Units <t< th=""><th></th><th>🌾 CanTpReadPara</th><th>ameterApi</th><th>FALSE</th><th></th><th>\sim</th><th></th></t<>		🌾 CanTpReadPara	ameterApi	FALSE		\sim	
		✓ ♥ CanTp✓ CanTpGenera	al (11)	🎉 CanTpChangeP	arameterApi	FALSE		\sim	
		⊟ CanTpGer ⊟ CanTpConfig	neral_0 [11]	🌠 CanTpVersionIr	foApi	FALSE		\sim	
Basic Edito	<u>r</u>	Units Units <t< th=""><th></th><th>🎉 CanTpDynIdSup</th><th>oport</th><th>-</th><th></th><th>\sim</th><th></th></t<>		🎉 CanTpDynIdSup	oport	-		\sim	
🔲 Properties 🛛		ComM		🌠 CanTpFlexibleD	ataRateSupport	-		\sim	
Property	ue 👫 🖾 🛃 🗸	₩ Dem ₩ EcuC <	>	🔀 CanTpGenericC	onnectionSupport	-		\sim	~
		4 AUTOSAR Validation	Tasks 🖹 Pro	blems 📮 Console					
		1 error, 0 warnings, 0 infos					Configure	e Filter	5 ▽
		Description		Object	URI	Resource	EObjectType		
		🛩 🏣 Errors (1 item)							
		OanTp must contai	in at least 1 ins	CanTpConfig	#/AUTOSAR/CanTp/Ca	BSW_CanT	EcucParamConfContain		

校验配置 VERIFY CONFIGURATION



校验完成后,可以进行代码生成步骤,可以在界面中看到生成的过程和结果。

Verification of the configuration can be carried out after the setup is complete. Once the verification is done, you can proceed to the code generation step. You can observe the process and outcome of the generation right within the interface.

<u> </u>	al 1 a 1 a a 1 1				_	~
workspace	- Classic Designer - ZC MuNii			-		×
File Edit Na	vigate Search Project Mo	odel Form Editor Validate Run Window	/ Help			
- 🖬 🖷 🔁	🕸 🖻 💽 💁 🔹 🛷 🔹	셸 - 웹 - 🏷 🗘 🖓 -			Quic	k Access
Configuratio	n Editors 📃 🗖	🐺 *Basic Editor 🛛				- 8
text to filter			CanTpGeneral_0 [/AUTOSAR/CanTp/CanTpGene	ral]		
Base Servic	es ¥	Models ^				^
Communica	tion ¥	> 🕂 Can	- Name	CanTpGeneral_0		
Diagnostics	*	> 🐈 CanIf 😍 CanNm	CanTpDevErrorDetect	FALSE	\sim	
		> 🐈 CanSM > 🐈 CanTp	🏹 CanTpReadParameterApi	FALSE	\sim	
		♀ CanTrcv ♀ Com	🔀 CanTpChangeParameterApi	FALSE	~	
		> 🐈 ComM	🔀 CanTpVersionInfoApi	FALSE	~	
Basic Edite	<u>or</u>	> 🐈 Dem > 🐈 EcuC	ि [] CanTpDynldSupport	-	~	
Properties		EcuM Fee	🔀 CanTpFlexibleDataRateSupport	-	~	
Property	🔚 🍄 🗔 🛃 🗢		CanTpGenericConnectionSupport		~	
roperty	value	< >				*
		🥵 AUTOSAR Validation 🧔 Tasks 🔝 P	Problems 📮 Console 🛛	B. 🔝 🕑 🖃 🖛	• 📑 👻	- 0
		MuNiu Studio Console				
		<pre>**** Generation started **** Generating PduR_Cfg.h Generating PduR_Cfg.h: OK. Generating PduR_Cfg.c. OK. Generating PduR_Cfg.c: OK. ***** Generation finished ****</pre>				~
		<				>

生成代码 GENERATE CODE

如下是生成的头文件示例:

Here is an example of a generated header file:

*======================================			
k RGanuninka (G) 2010 ⊨¥≣ándi	(利益方理公司)	Theorem is a marked and a marked	
ecopyright (C) 2019, 12/44943	A 44102 49 100 22 40 1	snanghai 20 Technology Co., L	ta. All rights reserved
This software i Shanghai ZC Tec All other right /	s copyright protected hnology Co., Ltd. gra s remain with Shangl	d and proprietary to Shanghai ants to you only those rights mai ZC Technology Co., Ltd.	ZC Technology Co., Ltd. as set out in the license conditions
• • • • • • • • • • • • • • • • • • • •	*****	*******	*****
@file <adctst_cfg.h></adctst_cfg.h>			
@brief <adc自检模块的配< td=""><td>置项></td><td></td><td></td></adc自检模块的配<>	置项>		
<compiler: mcu<="" multi7.1.4="" td=""><td>: SPC58NH84></td><td></td><td></td></compiler:>	: SPC58NH84>		
@Version <01.00.00>			
@author <束玉童>			
GASIL :			
@Generate Date <2023/	03/30>		
fndef ADCTST_CFG_H =fine ADCTST_CFG_H		Includes	
-lude Word manage by			
clude "Cdd_Mcu_Safety.h"			
	Ma	acro Definitions	
fine BandgapValue	(1.22f)	/* bandgap基准电压	*/
fine SARADC	(3.31) SAR_ADC_12BIT_1.:	/*Digital Bandgap [CDR[12]	Voitage */ /*ADC转换校验位*/
fine ADC_12bit_0_channel fine ADC_12bit_1_channel fine ADC_12bit_2_channel fine ADC_12bit_3_channel	SAR_i SAR_i SAR_i SAR_i	ADC_12BIT_0.ICDR[38] ADC_12BIT_1.ICDR[16] ADC_12BIT_2.ICDR[5] ADC_12BIT_3.ICDR[62]	/*aDC0使用的内部通 /*aDC1使用的内部通 /*aDC2使用的内部通 /*aDC3使用的内部通
fine ADCTST SARBIST ADC RESUL	T 0 (IoHwResultADC4[))))	

自动生成的头文件示例 AUTOMATICALLY GENERATED HEADER FILE EXAMPLE

5.5 导入 DBC 文件 Import DBC Files

(1) 如图所示,在创建工程和设置好平台后,点击 Import Communication Data 按钮并选择 Import DBC file,开始导入 DBC 文件。

As shown in the figure, after creating the project and setting up the platform, click the "Import Communication Data" button and select "Import DBC file" to start importing the DBC file.

Import Communication Data				
Import Diagnostic Data				
Import OS Data				
<filter></filter>		 	 	
FileName	Туре	ECU Instance		File I
EP37_E03_BDCS_SubCAN_Matrix_1.0_	_CS			O:/c
<				>
Messages				*
import success				^
		 	 	~
		 	 	~
TBC/LDF File importer				^
Select Communication Imp	port Type.			

P DBC/LDF File Importer		\sim
elect Communication Import Type.		
inter communication import type.		
Import DBC file		
) Import LDF file		

导入 DBC 文件 IMPORT DBC FILES

(2) 选择合适的 DBC 文件,选择需要的 ECU 结点:

Select the appropriate DBC file and the required ECU nodes.



						Sec. 27. 27. 23
₩ DBC/LDF File Importer				—		×
Select Frame/Signals to impor	t					
Provide import details.						
Which CanDB-node is this ECU	emulating?					
CS						~
Select which frames and/or sigr	nals that should be	imported.				
 >			Se	lect All	Unselect	All
?	< Back	Next >	Finish		Cancel	

选择 ECU 节点 ELECT ECU NODE:

(3) 将 COM 模块及 DIAG 模式和 CAN Controller 模块进行绑定:

Bind the COM module, DIAG mode, and CAN Controller module:



绑定 CONTROLLER BIND CONTROLLER

(4) 最后生成的效果如下图所示。

The final generated effect is shown in the figure below.

Imput Files Impact Editor 123 Itext to filter Impact Editor 123 Models Impact Editor 123 Impact Editor 123 Impact Editor 123	知简/ sy to d
text to filter Models > \$* Can1p > \$* Can1f > \$* Can1f > \$* Can1f > \$* Can1p > \$* Can1pGeneral > \$* Com > \$* Can1pUnldSupport > \$* Linif > \$* Liniff > \$* PduR > \$* PduR > \$* SecOC	
Models Image: Second state	
> + Cani Image: Name CanTpGeneral > + CanTp Image: CanTpDevErrorDetect Image: CanTpGeneral > + CanTpConfig Image: CanTpChangeParameterApi Image: CanTpChangeParameterApi > + Com Image: CanTpChangeParameterApi Image: CanTpChangeParameterApi > + Linlf Image: CanTpGenericConnectionSupport Image: CanTpChangeParameterApi > + SecOC Image: CanTpP	
> + CanNm > + CanTp > + CanTp > + CanTp > CanTpGeneral > + Com > + Com > + ComM > + ComM > + EcuC > + EcuC > + Linf > + Linf > + Linf > + Linff > + Linff > + Linff > + CanTpGenericConnectionSupport > + CanTpGenericConnectionSupport	-
✓ CanTp CanTpGeneral ✓ CanTpConfig ✓ CanTpCCnfig ◇ ♥ Com ✓ CanTpCCnfig ✓ CanTpCChangeParameterApi ✓ ◇ ♥ Com ✓ CanTpCChangeParameterApi ✓ ✓ ◇ ♥ Com ✓ ✓ ✓ ◇ ♥ Com ✓ ✓ ✓ ◇ ♥ Com ✓ ✓ ✓ ✓ ◇ ♥ CanTp ✓ ✓ ◆ ♥ CanTp ✓ ✓ ✓ ◆ ♥ Lin ✓ CanTpGenericConnectionSupport ✓ ↓ ♥ PduR ✓ ✓ ✓ ↓ ♥ Poure ✓ ✓	
> Image: CanTpConfig Image: CanTpChangeParameterApi Image: CanTpChangeParameterApi > Image: ComM Image: CanTpCresionInfoApi Image: CanTpCresionInfoApi > Image: CanTpC Image: CanTpC Image: CanTpC > Image: CanTpC Image: Can	
> Irror ComM Irror CanTpVersionInfoApi Irror CanTpVersionInfoApi > Irror Dcm Irror CanTpDynIdSupport Irror CanTpDynIdSupport > Irror Linf Irror CanTpFlexibleDataRateSupport Irror CanTpCenericConnectionSupport > Irror PduR Irror CanTpGenericConnectionSupport Irror Irror Secol Irror CanTpGenericConnectionSupport Irror	
> ↓• EcuC 「∑ CanTpDynldSupport > ↓• Lin 「∑ CanTpFlexibleDataRateSupport > ↓• LinSM > ↓• PduR > ↓• SecOC	
> Inf Image: CanTpFlexibleDataRateSupport > Image: LinSM > Image: PduR > Image: SecOC Image: LinSM	
> ₩ PduR > ₩ SecOC Image: SecOC Image: SecOC Image: SecOC Image: SecOC	
ti CanTpPaddingByte 170	
	-
Image: CanTpOsAlarmCounterRef	•

导入 DBC 文件后生成的配置 THE CONFIGURATION GENERATED AFTER IMPORTING THE DBC FILE.

5.6 导入 LDF 文件 Import LDF File

(1) 选择导入 LDF 文件

Select to import the LDF file

🗃 Input Files 🕺 🖼 Basic Editor		P 🖬
Import Communication Data		
Import Diagnostic Data	DBC/LDF File Importer DBC/LDF File Importer	
Import OS Data	Select Communication Import Type.	
- Eller	O Import DBC file	
	Import LDF file	
test.oil		
Messages		*
	Cancel	^ ^

(2) 选择文件路径

Select the file path



😽 DBC/LDF		×			
Import LDF File Import a ldf file from the local file system into the workspace					
Select File:	O:\cheese.bai\项目\	65	Brows	e	

(3) 生成的效果如下:

The generated effect is as follows:

ext to filter	₩ PduR ►		
lodels			
Can Can Canthm Canthm ComM ComM	ImplementationConfigVariant PostBuildVariantUsed SupportedConfigVariants	PduR - VARIANT-POST-BUILD, VARIANT-PRE-COMPILE	

5.7 导入 ODX 文件 Import ODX File

(1) 如图所示, 在创建工程和设置好平台后, 点击 Import Diagnostic Data 按钮并选择 Import ODX File, 开始导入 ODX 文件。

As shown in the figure, after creating the project and setting up the platform, click the "Import Diagnostic Data" button and select "Import ODX File" to start importing the ODX file.



📑 Input Files 🛛 🐺 Basic Editor Import Communication Data Import Diagnostic Data E Import OS Data <Filter> FileName ECU Instance File Path Туре EP37_E03_BDCS_SubCAN_Matrix_1.0_CS... O:/cheese.bai/项目/DPL/0603/EP37_E03_BDCS_S < > Messages \$ import success

₩ ODX/DEXT File Importer		×
Select Diagnostic Import Type.		
● Import ODX file		
○ Import DEXT file		

导入 ODX 文件 IMPORT ODX FILE

(2) 选择 ODX 文件:

Select ODX File

₩ ODX/DE	XT File Importer	_		×
Import OD)	File			
ODX Versio	n: 2.1.0			
Select File:	O:\cheese.bai\项目\DPL\0603\test.odx;		Brows	e

选择 ODX 文件 SELECT ODX FILE

(3) 选择 ECU 节点:

Select the ECU node:



₩ ODX/DEXT File Importer		×
Selection of an ODX ECU		
please choose the ODX ECU which shall be imported.		
New_ECU_1		
ESD_New_ECU_1_faultMemory		

选择 ECU 节点 SELECT THE ECU NODE:

(4) 最后, 生成配置, 效果如下图:

Finally, generate the configuration. The effect is shown in the figure below.

Input Files 🛛 🖼 Basic Editor 🛛				
text to filter	₩ Dem ₩ DemCon	<u>ifigSet0</u> + 🐈 <u>DemDTC</u> :	<u>s (16) </u>	
Models	^			
₩ Can ₩ CanNm	- Name		DTC_0x17A261	
🐓 Com 🛫 ComM	±i DemDTCFunction	nalUnit	0	
→ 🙀 Dcm	♥ DemDTCSeverity	у	DEM_SEVERITY_CHECK_AT_NEXT_HALT	~ -
✓ 😫 Dem ✓ 😾 Dem	1 DemDtcValue		1548897	
 > ⊟ DemGeneral0 > ⊟ DemConfigSet0 	👻 DemWWHOBDD	OTCClass	DEM_DTC_WWHOBD_CLASS_NOCLASS	~ -
 DemComponent [1*] DemDTCAttributes (065535) 	DemDTCAttribut	tesRef	DTC_0x17A261_Properites - /ZCPkg/Dem/DemConfigSet0/DTC_0x17A261_Properites	
> 🖨 DemObdDTC [065535] > 🖨 DemDTC [065535]	DemObdDTCRef	f	DemObdDTC_0 - /ZCPkg/Dem/DemConfigSet0/DemObdDTC_0	•
DTC_0x17A261 DTC_0x17A361				
 DTC_0x17A461 DTC 0x17BA8B 				
DTC_0x17BB07				
DTC_0x56B429				
DTC_0xC07388				
☐ DTC_0xC29387				
➡ DTC_0xD56217				
■ DTC_0xD50316	*			

导入 ODX 文件后生成配置 GENERATE THE CONFIGURATION AFTER IMPORTING THE ODX FILE.



5.8 导入 DEXT (ARXML) 文件 Import DEXT(ARXML) File

(1) 如图所示, 在创建工程和设置好平台后, 点击 Import Diagnostic Data 按钮并选择 Import Dext File, 开始导入 Dext 文件。

As shown in the figure, after creating the project and setting up the platform, click the "Import Diagnostic Data" button and select "Import Dext File" to start importing the Dext file.

📴 Input Files 🛛 🐺 Basic Editor			- 8
Import Communication Data			
<filter></filter>			
FileName	Туре	ECU Instance	File Path
EP37_E03_BDCS_SubCAN_Matrix_1.0_CS			O:/cheese.bai/项目/DPL/0603/EP37_E03_BDCS_S
<			>
Messages			*
import success			^

₩ ODX/DEXT File Importer	_	×
Select Diagnostic Import Type.		
O Import ODX file		
Import DEXT file		

导入 DEXT 文件入口

(2) 选择 DEXT 文件:

Select DEXT File:



选择 DEXT 文件 SELECT DEXT FILE



(3) 最后, 生成配置, 效果如下图:

Finally, generate the configuration. The effect is shown in the figure below.

to filter	Dem > 😌 DemConfigSet0 > 😌 DemD	<u>TCs [16]</u> ▶ ₩ <u>DTC_0x17A261</u>	
els • Can	^		
CanNm	- Name	DTC_0x17A261	
Com ComM	t DemDTCFunctionalUnit	0	
• Dcm	DemDTCSeverity	DEM_SEVERITY_CHECK_AT_NEXT_HALT	~
Dem 🐈 Dem	ti DemDtcValue	1548897	
> 🖨 DemGeneral0 > 🖨 DemConfigSet0	DemWWHOBDDTCClass	DEM_DTC_WWHOBD_CLASS_NOCLASS	~
> 🖨 DemComponent [1*] > 🖨 DemDTCAttributes (065535)	DemDTCAttributesRef	DTC_0x17A261_Properites - /ZCPkg/Dem/DemConfigSet0/DTC_0x17A261_Properites	
 DemObdDTC [065535] DemDTC [065535] 	DemObdDTCRef	DemObdDTC_0 - /ZCPkg/Dem/DemConfigSet0/DemObdDTC_0	
DTC_0x17A261			
DTC_0x17A361			
DTC_0x17BA8B			
DTC_0x17BB07			
DTC_0x56B429			
DTC_0xC07388			
DTC_0xC12887			
DTC_0xC29387			
DTC_0xD56217			
UIC_0xD56316			

导入 DEXT 文件后生成配置 GENERATE THE CONFIGURATION AFTER IMPORTING THE DEXT FILE.

- 5.9 导入 OIL 文件 Import OIL File
- (1)如图所示,在创建工程和设置好平台后,点击 Import OS Data 开始导入 OIL 文件。
 As shown in the figure, after creating the project and setting up the platform, click
 "Import OS Data" to start importing the OIL file.

🗃 Input Files 🙁 🐺 Basic Editor				- 8
Import Communication Data				
Import Diagnostic Data				
Import OS Data				
<filter></filter>				
FileName	Туре	EC	CU Instance	File Path
EP37_E03_BDCS_SubCAN_Matrix_1.0_CS				O:/cheese.bai/项目/DPL/0603/EP37_E03_BDCS_S
<				>
Messages				*
import success				^
				~





😽 File Import Wizard			\times
Import OIL File			
Import a oil file from the local file system into the work	space		
Select File: T:\Dew.Wang\test.oil		Brows	e
O Do you want to remove all configurations under the	OS module		
0	Finish	Cance	el

是否与现有模块合并 MERGE WITH THE EXISTING MODULES OR NOT

(2) 最后, 生成配置, 效果如下图:

Finally, generate the configuration. The effect is shown in the figure below.

t to filter	Us V OsApplications	121 > 🐈 OsApplication_APP >	
dels	^		
• Os			
🗸 🐈 Os	Name	UsApplication_APP	•
	VosTrusted		
Containing (U)			
Alarm APP BSW P2 5ms	OsTrustedApplication		
> Alarm APP BSW P5 5ms	WithProtection	—	
> 🖨 Alarm_APP_BSW_P6_2ms	SApplicationCoreBe	0	~
>		*	
>	S OsAppAlarmRef	Alarm_APP_USER_P1_10ms	
Alarm_APP_USER_P3_5ms			
Alarm_APP_USER_P4_2ms	SAppCounterRef	Counter_0	2
> @ OsAppMode [1.15]			
✓	OsAppIsrRef		E.
> OsApplication_APP			
> OsApplication_BSW	SAppScheduleTable	Ket	A.
> @ OsCounter [01]		ADD LICER D1 10mm	P
	E outproverer		E
OsBesource [0*]	OsRestartTask		
OsScheduleTable [0*]			
> 🖨 OsTask [0*]			
> 🖨 SystemTimer [01]			
OsTrace [01]			
/* PduR			



6 证书 CERTIFICATE



木牛软件著作权登记证书 MUNIU SOFTWARE COPYRIGHT REGISTRATION CERTIFICATE

