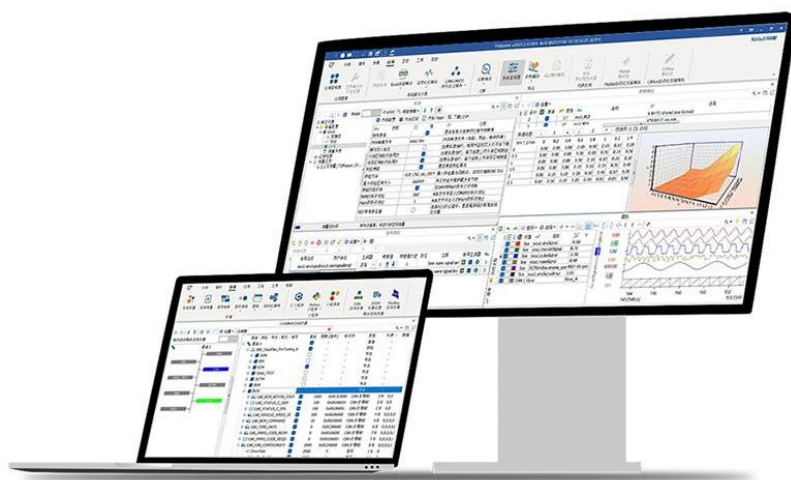




TSAutotest 测试用例编辑器用户手册



关键字 TSAutotest、测试用例编辑器

修订记录:

版本	日期	改动
V1.0.0	2026.08.06	创建文档

目录

1 介绍	5
1.1 关于手册	5
1.2 产品定位	5
1.3 术语及缩略语	6
2 功能概要	7
3 快速开始	8
4 功能说明	9
4.1 ALM 云平台	9
4.1.1 测试资源管理	9
4.1.2 文档管理	10
4.1.3 远程工程同步	11
4.1.4 编辑模式	12
4.1.5 AI 辅助生成测试用例设计	13
4.2 本地工程管理	13
4.3 用例编辑	14
4.3.1 本地映射	16
4.3.2 本地变量	17
4.4 测试库	17
4.5 信号管理	18
4.6 语言切换	19
4.7 实时检查	20
5 示例	21
5.1 配置 TSMaster 工程	21
5.2 配置 TSFMU 资源包	22
5.3 打开工程	23
5.4 新建用例文件	24
5.5 添加测试步骤	24
5.6 导入用例	25
5.7 执行用例	26
6 附录	28
6.1 读写步骤说明	28
6.1.1 读步骤	28
6.1.2 写步骤	30
6.2 测试库步骤说明	31
6.2.1 数据操作	31
6.2.2 流程控制	32
6.3 通用步骤说明	36
6.3.1 MAPort	36
6.3.2 EES Pin	37
6.4 小程序 API 说明	38
6.5 表达式说明	38
7 常见问题及解决方法	41

7.1 Git 无访问权限	41
8 联系方式	43

1 介绍

本手册用于介绍测试用例编辑器当前已实现的功能，包括用例编辑、工程和资源加载方式、用例步骤功能说明以及操作示例。

1.1 关于手册

以下表格罗列了手册中使用到的样式和符号。

样式	使用场景
【】	表示点击的界面元素、窗口和对话框的名称。
加粗	用于文本信息的强调。
超链接	超链接和引用。
<CTRL>+<S>	快捷方式。

符号	使用场景
	用于展示警告信息，以提醒用户留意。
	用于提醒静电放电可能造成损坏。
	用于展示补充/额外信息。
	用于提供示例。
	用于提供分步骤的指引。
	用于提供对文件进行编辑操作的说明。

1.2 产品定位

TSAutotest 是一款测试用例编辑器软件，核心功能为用例编辑。用户可以在图形化编辑器中组织测试步骤，维护本地映射和本地变量，并基于 TSMaster 导出的资源文件使用

信号读写和小程序 API 调用步骤。编辑后的用例文件（.cas）可导入 TSMaster，并转换为图元测试程序后执行。

搭配 ALM 系统使用时，可以作为双向可追溯性与持续集成能力的 ECU 软件自动化测试体系中的一环，可以对 ALM 系统上的测试资源进行编辑，并由 ALM 系统调度 TSMaster 执行测试用例。

1.3 术语及缩略语

术语	说明
测试用例	由多个测试步骤组成的自动化测试逻辑，当前版本以树形步骤编辑为主。
测试步骤	用例中的最小编辑单元，例如 Wait、Loop、IfThenElse、Read、Write、MiniApi、MAPort 等。
本地映射	用例内部使用的映射标签，用于将步骤中的简短名称关联到真实信号或资源路径。
本地变量	用例内部使用的变量，可用于保存 API 返回值、信号读取值或表达式计算相关数据。
TSMaster 工程	TSMaster 的 .TSProj 工程文件，当前 TSAutotest 通过临时入口读取工程目录下的导出资源。
TSFMU	TSMaster 导出的资源包，当前用于提供信号文件和小程序 API 描述文件。
ALM	Application Lifecycle Management，应用生命周期管理系统，用于管理测试资源和调度测试执行。
EES	Electrical Error Simulation，电气错误模拟，用于在测试过程中向 ECU 引脚注入电气故障。
.cas	TSAutotest 的测试用例文件格式，以 XML 形式存储测试步骤、本地映射和本地变量。

2 功能概要

当前 TSAutotest 测试用例编辑器已实现用例步骤编辑和基础工程管理等功能。编辑后的用例可导入 TSMaster，并转换为流程图程序执行测试。

主要功能包括：

- ALM 云平台
- 本地工程管理
- 用例编辑
- 信号管理
- 测试库
- 语言切换
- 实时检查

3 快速开始

1. 打开测试用例编辑器，点击【打开本地工程】按钮，选择 TSMaster 工程。
2. 在资源管理器中选中【用例】文件夹。
3. 点击右键，在菜单中点击【添加文件】，输入用例文件名称。
4. 双击新添加的用例文件。
5. 从测试库中拖拽一个步骤到用例编辑区。



图1 快速开始示例

4 功能说明

当前核心功能主要为用例编辑，其余模块如测试库、信号树等均为对用例编辑功能的增强，例如信号树为用例编辑提供了信号读写步骤，测试库提供了流程控制、数据操作等步骤，工程功能提供了多用例编辑的基础。

配合 ALM 云平台，可以实现远程工程管理、同步、AI 辅助测试用例设计等。

4.1 ALM云平台

搭配 ALM 云平台，测试用例编辑器可以从云平台上选择测试资源并拉取到本地进行编辑，编辑后可以手动同步回云平台。

4.1.1 测试资源管理

登录 ALM 云平台后，可以选择 ALM 的项目和项目下配置的台架，台架会关联一份 TSMaster 的工程，选择台架并确认后，可以通过 Git 从存储库拉取关联的 TSMaster 工程到本地进行编辑。如果选择的台架未关联 TSMaster 工程，可以选择本地已配置好的 TSMaster 工程上传到存储库并关联到台架。

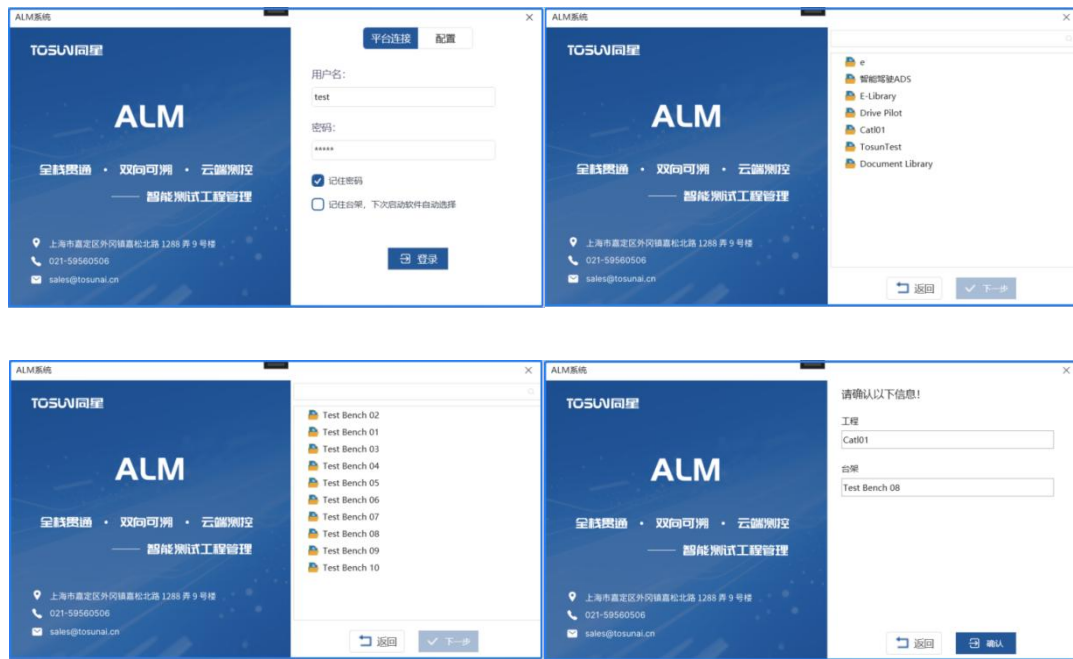


图2 测试资源管理

4.1.2 文档管理

与打开本地工程不同，从 ALM 云平台拉取远程工程时，第一次需要手动在文档工具箱中添加 TSF MU 资源包下的文件。文档工具箱分为【项目资料文件】和【映射文档】两个分类，添加方式为在对应分类节点上右键点击【导入】，在文件选择对话框中选择从 TSF MU 解压出来的 JSON 文件（如 can.json、lin.json、apis_all.json 等）。这些文件为用例步骤提供了信号读写和小程序 API 调用的数据源。

项目资料文件可以为 AI 辅助设计用例功能提供知识库，通过这些知识库，AI 可以更准确地提供用例设计。

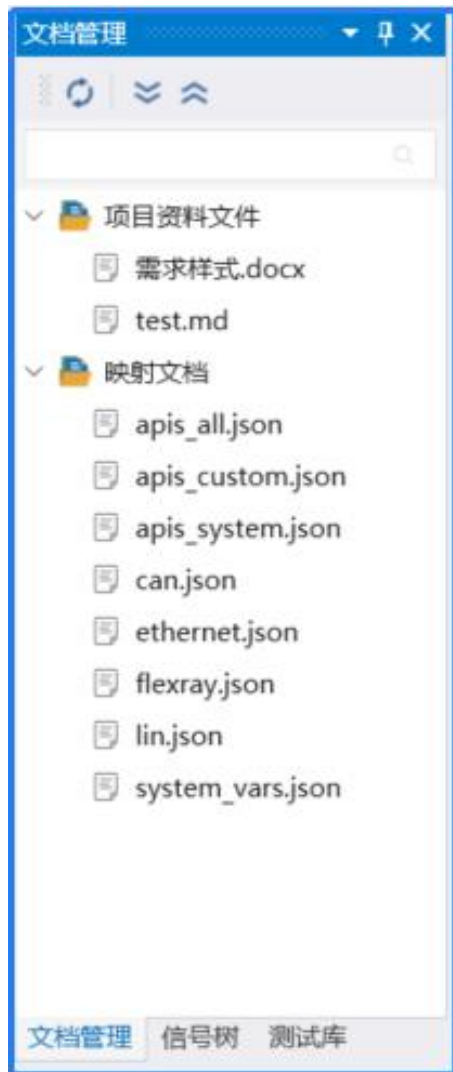


图3 文档管理

4.1.3 远程工程同步

拉取到本地的 TSMaster 工程，进行用例编辑后可以通过更新和提交进行同步，使本地的工程始终保持最新，或者修改及时推送到存储库。

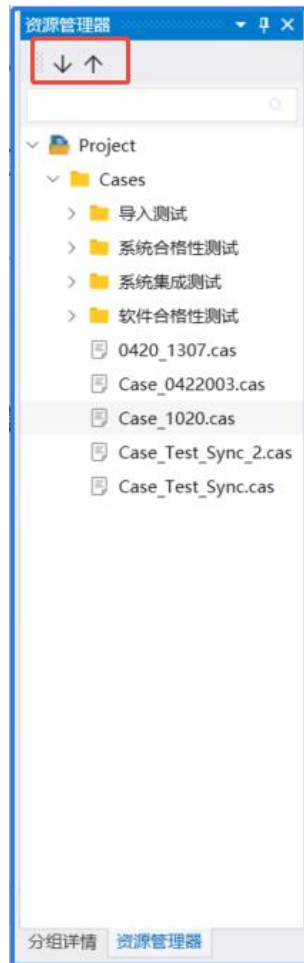


图4 远程工程同步

4.1.4 编辑模式

测试用例编辑器中支持多用户对同一个远程工程资源进行用例编辑，但是对于单个用例文件，同一时刻仅支持一个用户进行编辑，因此使用编辑模式进行锁定，默认模式为阅读模式。

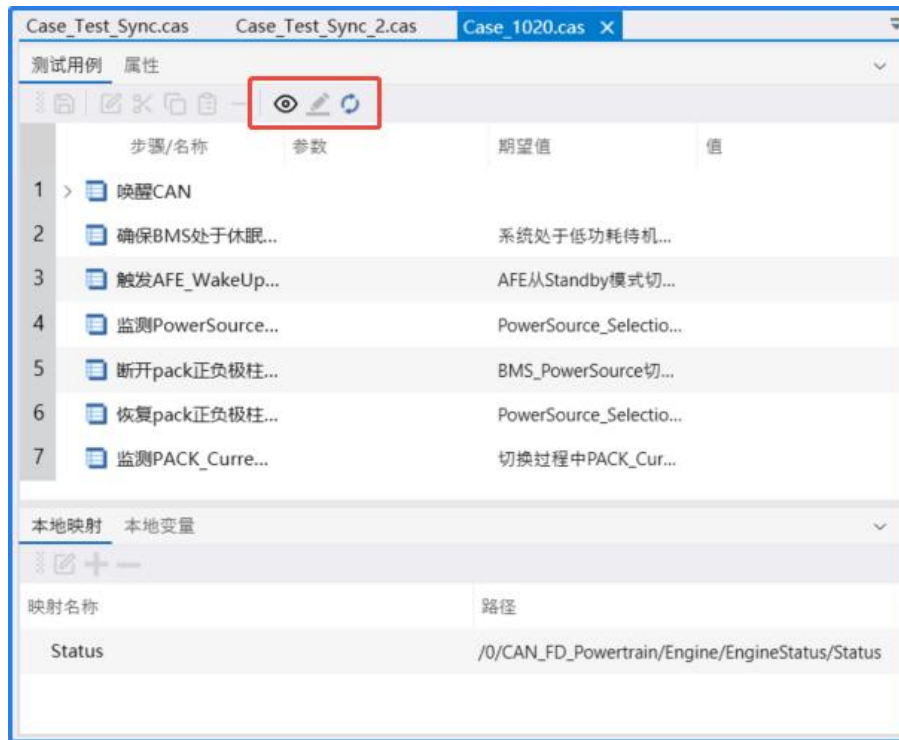


图5 阅读模式与锁定模式

4.1.5 AI辅助生成测试用例设计

AI 可以结合当前测试资源下的需求文件、信号文件，对用户提出的需求进行用例设计。当 AI 设计出用例分组后，可以应用到当前的用例文件中。

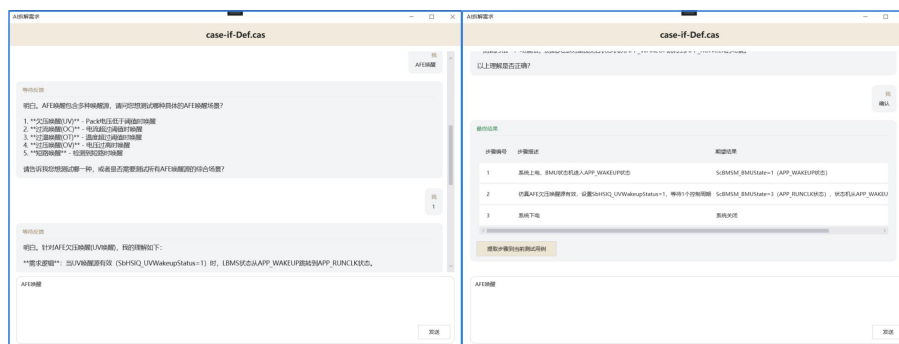


图6 AI辅助生成测试用例

4.2 本地工程管理

工程管理功能通过资源管理器模块实现，以树形结构显示工程中的文件，并通过右键菜单的方式来管理，包括对目录和用例文件的新增、剪切、复制、粘贴、删除等操作，以及从外部导入已存在的用例文件到指定目录。

通过启动页的【打开本地工程】或主界面的【打开】按钮打开 TSMaster 工程后，会从相对路径加载工程文件，路径为 ./TSTest/Project。系统会自动保存最近打开的工程记录（最多 20 条），在启动页可以直接从历史记录列表中打开之前的工程，远程工程支持自动登录。

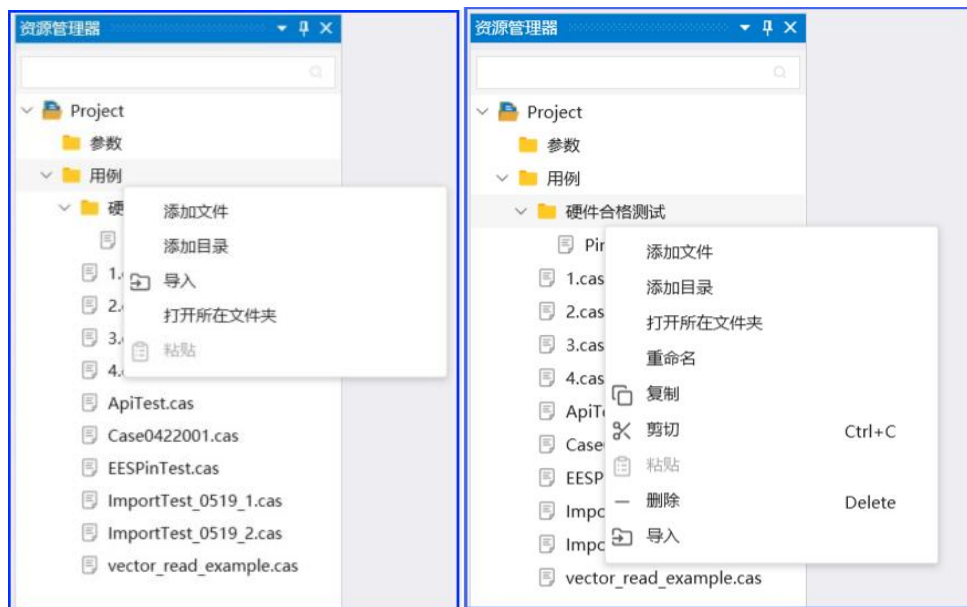


图7 工程管理

4.3 用例编辑

在资源管理器中双击需要编辑的用例文件后，会在文档选项卡中显示该用例的编辑视图。当修改用例后，标题末尾会显示 * 表示该用例已被修改，可以通过保存按钮进行用例文件保存。

用例编辑通过树的形式来显示和编辑用例步骤的顺序、位置和层级。某些步骤可以包含子步骤，例如 IfDef，当满足条件，执行段将会执行 IfDef 步骤中的子步骤。

用例编辑包含如下几种方式：

- 从测试库拖拽目标步骤到编辑区（新增测试库用例步骤）

- 在信号树上点击读或写（新增读写步骤）
- 双击目标步骤（编辑已有步骤）
- 对已有步骤拖拽到其它地方（调整步骤的位置、顺序或层级）
- 右键菜单（编辑、剪切、复制、粘贴、删除等）
- 工具栏（保存、编辑、剪切、复制、粘贴、删除）
- 快捷键（编辑：<CTRL>+<E>，剪切：<CTRL>+<X>，复制：<CTRL>+<C>，粘贴：<CTRL>+<V>，删除：<Delete>）

在远程工程模式下，用例编辑器还支持文件锁定与解锁功能。锁定后其他用户进入阅读模式，确保同一时刻仅一个用户编辑同一用例文件。工具栏中的锁定、解锁和刷新锁定状态按钮仅在远程工程模式下可用。

步骤拖拽约束：

- Break 和 Continue 步骤只能放置在 While 或 Loop 循环内部
- Then、Else、Case、Default 等分支子步骤不可拖动到其他位置
- IfThenElse 和 SwitchCase 步骤不允许拖入子步骤
- 跨用例文件粘贴时，系统会自动在目标用例中创建缺失的本地变量和本地映射，并在源用例中递减对应的引用计数

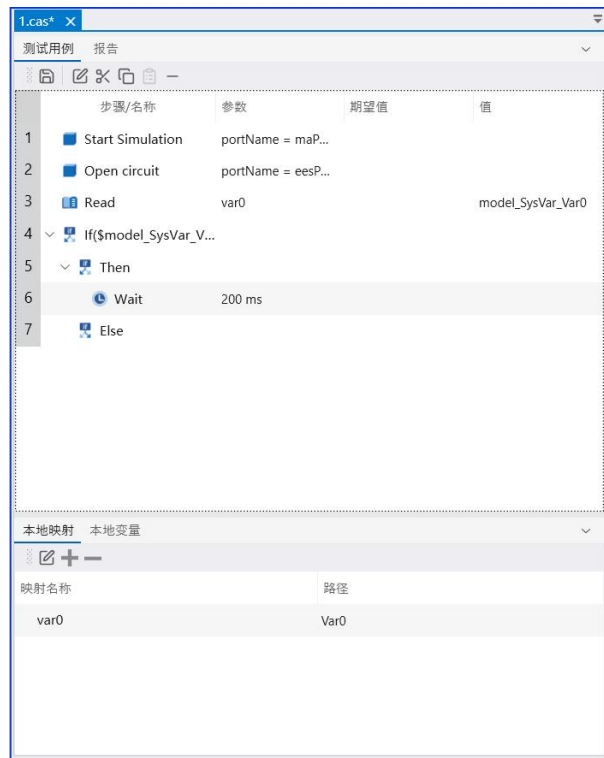


图8 用例编辑

4.3.1 本地映射

映射名称作为一个简短的别名，通过路径关联到一个具体的信号。通过该别名，即可对关联的信号进行读写。

映射名称在单个测试用例中需要具有唯一性。可以在用例编辑区域下方对已有的映射进行修改、删除和新增。除了在本地映射区域手动添加外，还可以通过信号树右键菜单中的【本地映射】选项，将选中的信号直接创建为当前用例的本地映射。编辑已有映射时，系统会递归更新所有引用该映射的 Read 和 Write 步骤。



图9 本地映射

4.3.2 本地变量

本地变量可以理解为一个在单个用例范围内的局部变量，可以由读步骤的存值产生，也可以手动新增本地变量。

本地变量可以被测试库中的 Bit extract、IfThenElse、IfDef、While、SwitchCase 等步骤使用。本地变量支持以下数据类型：Int32、UInt32、Int64、UInt64、Double、String、UInt8 Array、Int32 Array、Int64 Array 和 Double Array。



变量名称	数据类型	初始值	当前值	注释
model_SysVar_Var0	Int32	0	0	
var0	Int32	0		

图10 本地变量

4.4 测试库

测试库提供了包括流程控制、数据操作、通用步骤、小程序 API 等步骤。

- 流程控制：包括 Wait、Group、Exit、IfThenElse、IfDef、While、Loop、Break、Continue、SwitchCase、Return 等步骤，用于控制用例的执行流程或前期的用例设计等。
- 数据操作：目前提供了 Bit extract 步骤，后续还会增加其它类型的数据操作步骤，例如 Calculation 步骤。
- 通用步骤：包含 EES Pin、MAPort 下的 Start Simulation 和 Stop Simulation 步骤。
- 小程序 API 步骤：可以调用 TSMaster 提供出来的接口，和信号管理一样，可以调用的小程序 API 数据源依赖 TSMaster 导出的资源包，自定义接口也支持导出。小程序 API 库支持 JSON 格式（apis_all.json），按照 Scope、Group、API 三级结构组织。测试库支持搜索过滤功能，在搜索栏输入关键字即可过滤显示匹配的步骤。通过工具栏的刷新按钮可以重新加载 API 资源，展开和收拢按钮可以展开或折叠所有树节点。

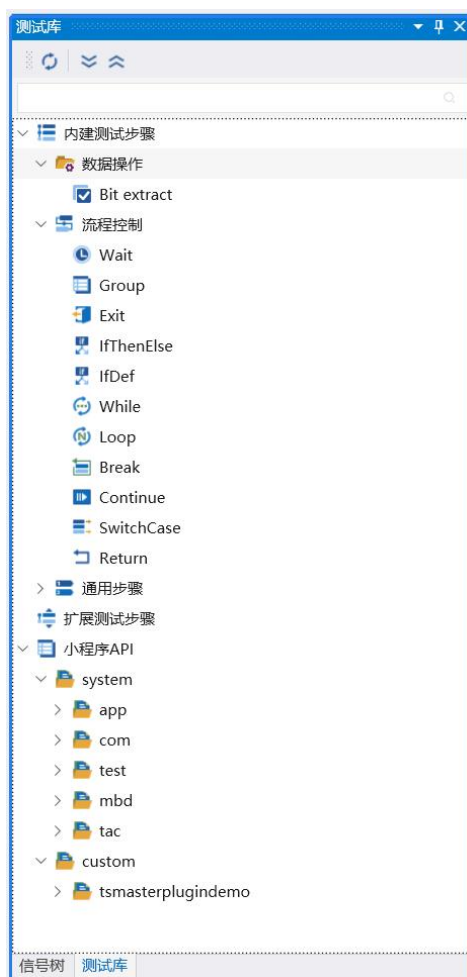


图11 测试库

4.5 信号管理

通过【信号树】模块，可以实现信号数据源显示，并提供读和写信号值的步骤。

目前信号树的数据来源依赖 TSMaster 导出的 TSFMMU 资源包。通过【打开】按钮打开 TSMaster 工程时，系统会从固定路径查找最新日期的 TSFMMU 资源包并加载，加载路径为 ./TSTest/Resources/。支持的信号类型包括系统变量（SysVar）、CAN、LIN、FlexRay 和 Ethernet 等总线类型的数据库信号。信号树按照总线类型、通道（Network）、节点（Node）、帧（Frame）、PDU、信号的层级组织。系统变量支持标量类型（int8、uint8、int16、uint16、int32、uint32、int64、uint64、float、double、string）和对应的数组类型。

在信号树中选中目标信号后，可以通过右键菜单执行以下操作：读（创建 Read 步骤）、写（创建 Write 步骤）、全局映射（添加到全局映射编辑器）、本地映射（添加到当前用例的本地映射）。信号树支持搜索过滤和展开/折叠操作。

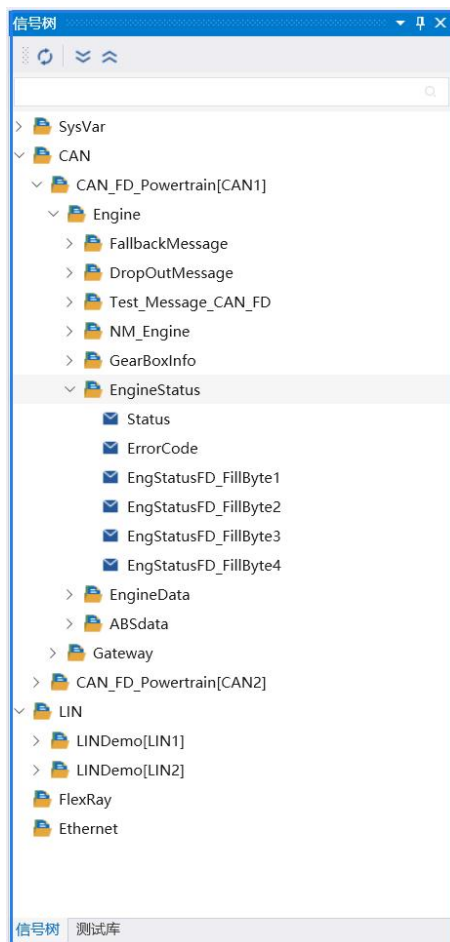


图12 信号树

4.6 语言切换

测试用例编辑器当前支持中文和英文，通过工具栏按钮【语言设置】打开设置界面进行语言切换，切换后需要重启软件生效。

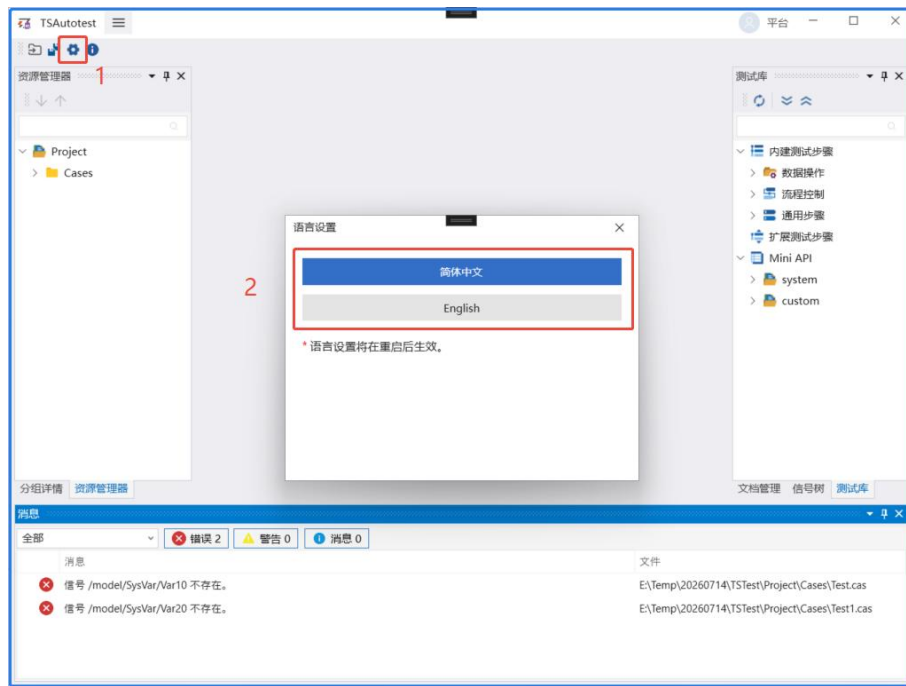


图13 语言切换

4.7 实时检查

实时检查模块在底部消息面板中显示用例文件的检查结果。系统通过文件系统监控自动检测 .cas 文件的变更，并在文件修改后自动执行检查。

检查内容包括：

- 信号是否存在
- 信号数据类型是否匹配
- Mini API 是否存在及参数是否匹配
- 本地映射是否存在
- 变量引用是否有效
- 字段是否为空等

消息按类型分为错误、警告和信息，支持按文件和类型过滤。双击消息可定位到用例编辑器中的对应步骤

5 示例

本章节提供配置资源包、新建工程、新建用例、通过用例修改 CAN 信号的值，并读取值是否符合预期的示例。

操作顺序如下：

1. 配置 TSMaster 工程。
2. 配置 TSFMU 资源包。
3. 打开工程。
4. 新建用例文件。
5. 添加测试步骤。
6. 从 TSMaster 中导入用例。
7. 执行导入的用例。

5.1 配置TSMaster工程



示例：

创建仿真工程，并导入 DBC 数据库。

- 打开 TSMaster, 点击【创建新工程】按钮
- 点击【仿真】按钮
- 点击【CAN 剩余总线仿真】
- 选择路径
- 点击【分析】->【数据库】->【显示 CAN 数据库】
- 在CAN 数据库界面中点击【通道1 [CAN]】的右键菜单下的【载入 CAN 数据库...】

选择 TSMaster 安装路径下的 CAN_FD_Demo.dbc 文件

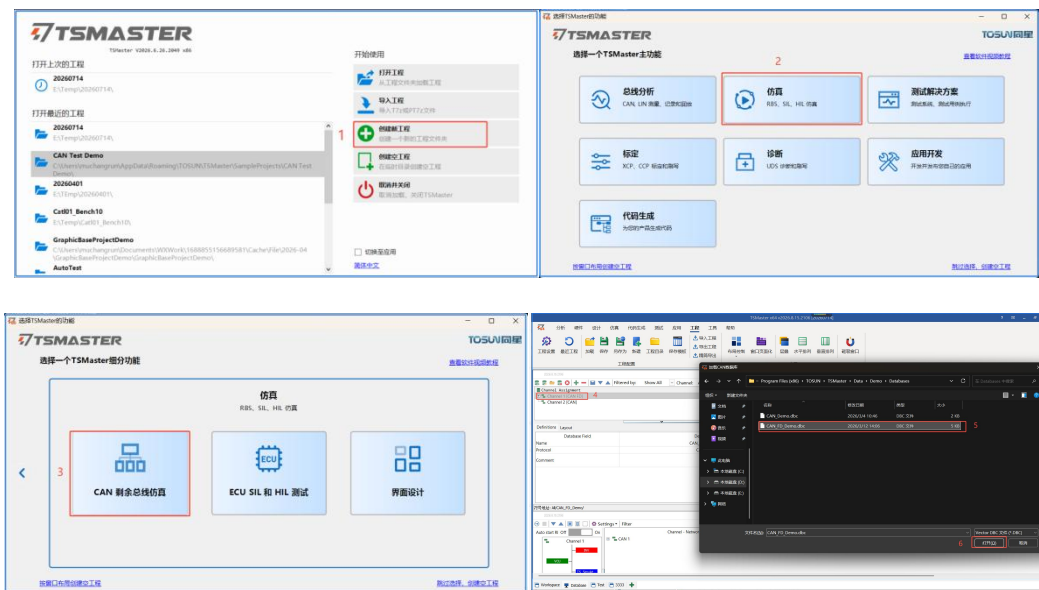


图14 新建TSMaster工程

5.2 配置TSFMU资源包

前提条件：已在 TSMaster 中打开工程、工程中已添加 CAN 数据库、并且 TSMaster 支持 TSMaster XIL 功能。

- 点击【仿真】下的【TSMaster XIL】按钮
- 点击【导出 TSFMU】选项卡
- 根据需要勾选信号和 API
- 点击【导出 TSFMU】按钮
- 在弹出的文件夹选择对话框中，选择当前工程下的 TSTest/Resources 目录（如不存在，可手动创建）
- 点击【保存】按钮

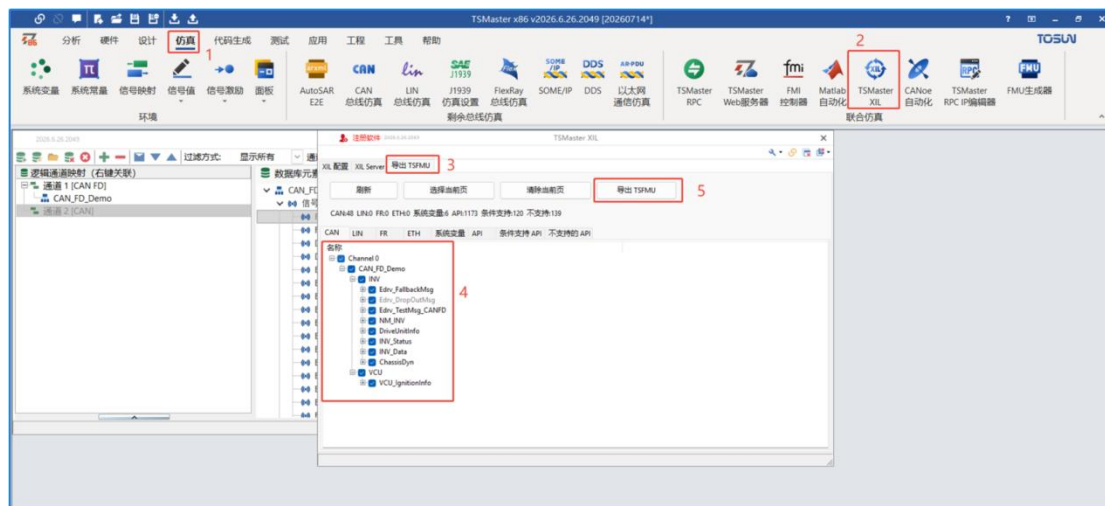


图15 配置TSFMMU资源包示例

5.3 打开工程

说明：该方案为临时方案，用于快速体验用例编辑功能。

前提条件：已将相关信号文件和 API 导出为 TSFMMU 资源包，并保存到工程的相对路径 TSTest/Resources/ 下。

- 点击 TSAutotest 中的【打开】按钮
- 在弹出的文件选择对话框中选择要打开的 TSMaster 工程，文件类型为 .TSProj
- 点击文件选择对话框中的【打开】按钮



图16 打开示例工程

5.4 新建用例文件

- 在【资源管理器】中选中【用例】节点
- 在右键菜单中点击【添加文件】,在弹出框中输入文件名称
- 点击【确定】按钮
- 双击新添加的用例文件

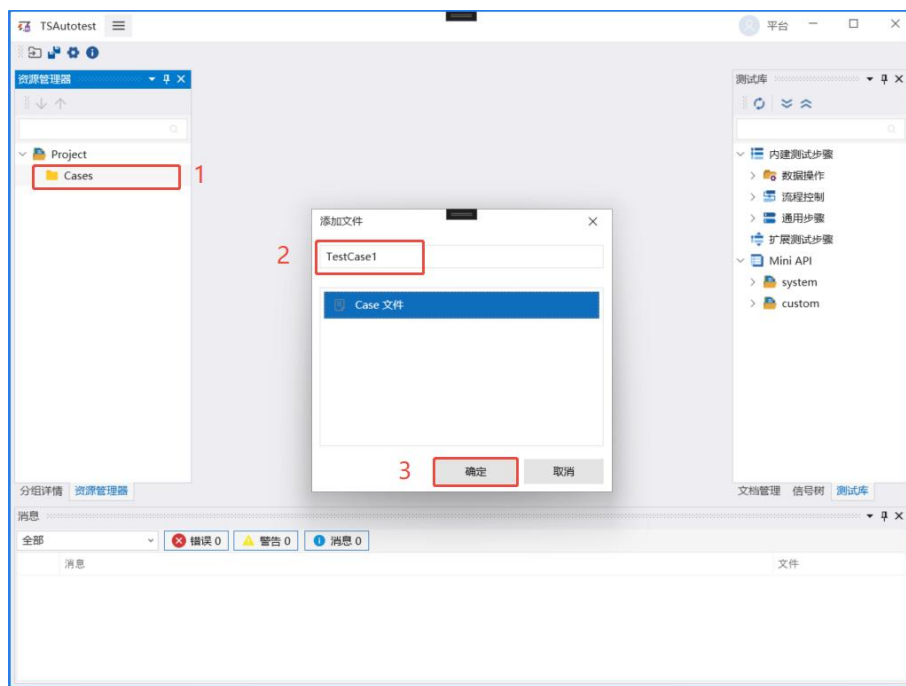


图17 添加用例文件示例

5.5 添加测试步骤

前提条件：存在已打开的用例文件，已加载 TSFMU 资源包且资源包中存在信号。

- 在【信号树】中搜索信号 IgnKey，并选中目标->点击右键菜单->点击【写】菜单
- 在【值】输入框中输入1
- 重新在 IgnKey 信号上点击【右键菜单】，并点击【读】
- 勾选上【评估】
- 【测量】输入框中输入1

- 点击【确定】按钮

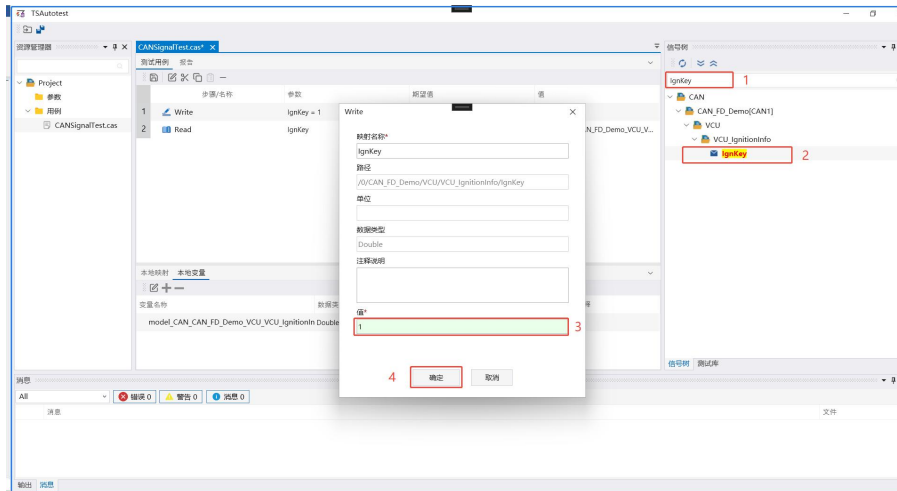


图18 添加读写步骤 1

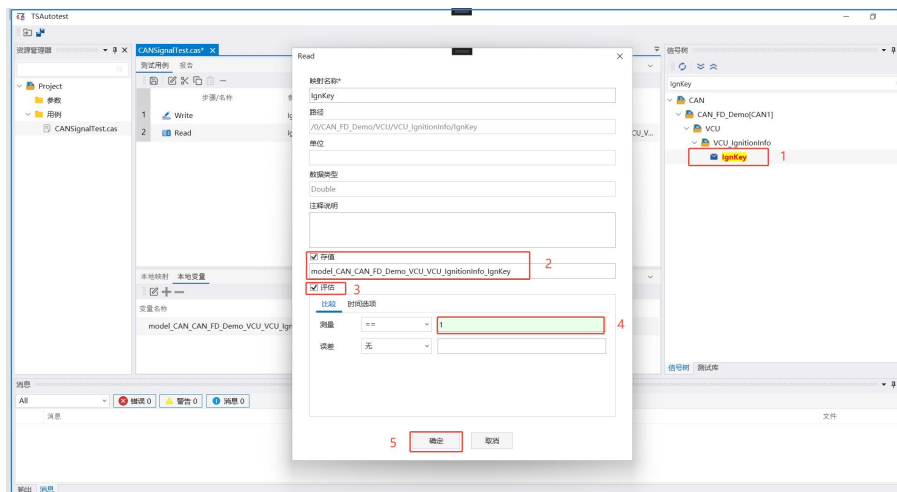


图19 添加读写步骤 2

5.6 导入用例

- 点击【设计】 -> 【流程图生成器】
- 选择【TOSUN Excel 导入器】
- 点击【Add】按钮
- 选择编辑好的 .cas 用例文件

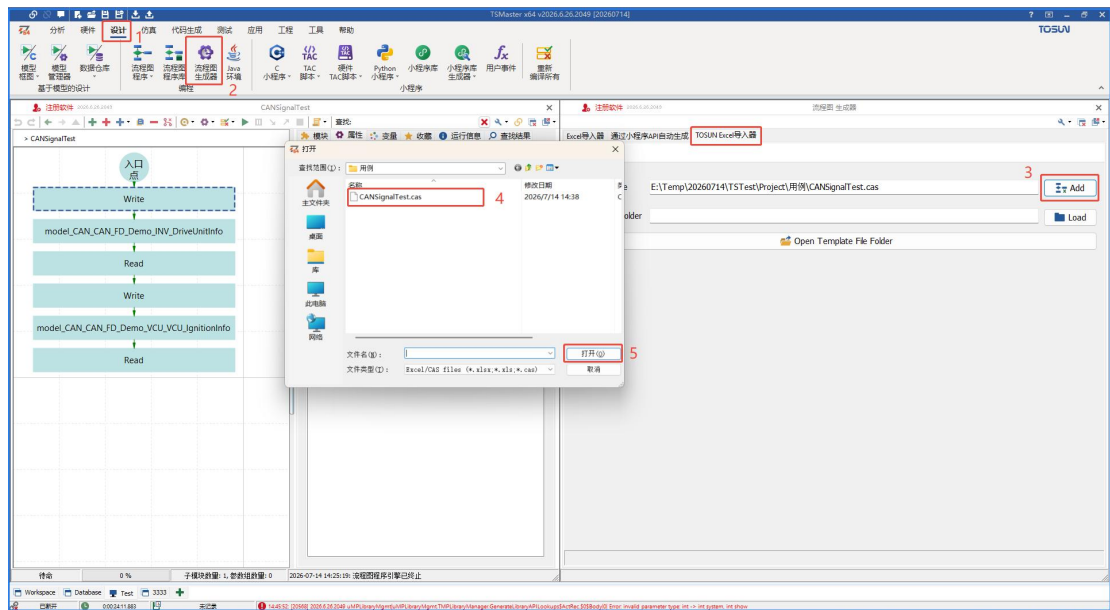


图20 导入用例

5.7 执行用例

- 点击【仿真】下的【CAN 总线仿真】
- 勾选【VCU】节点下的【VCU_IgnitionInfo】帧的激活
- 点击运行剩余总线仿真
- 点击运行当前的流程图程序
- 查看信号值是否符合预期

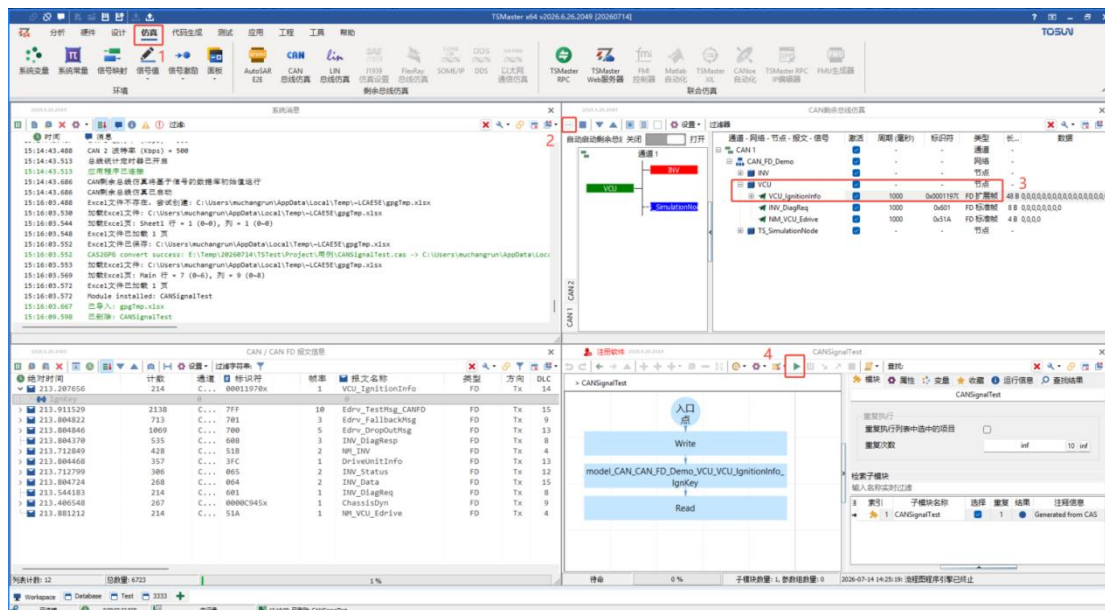


图21 执行用例步骤 1

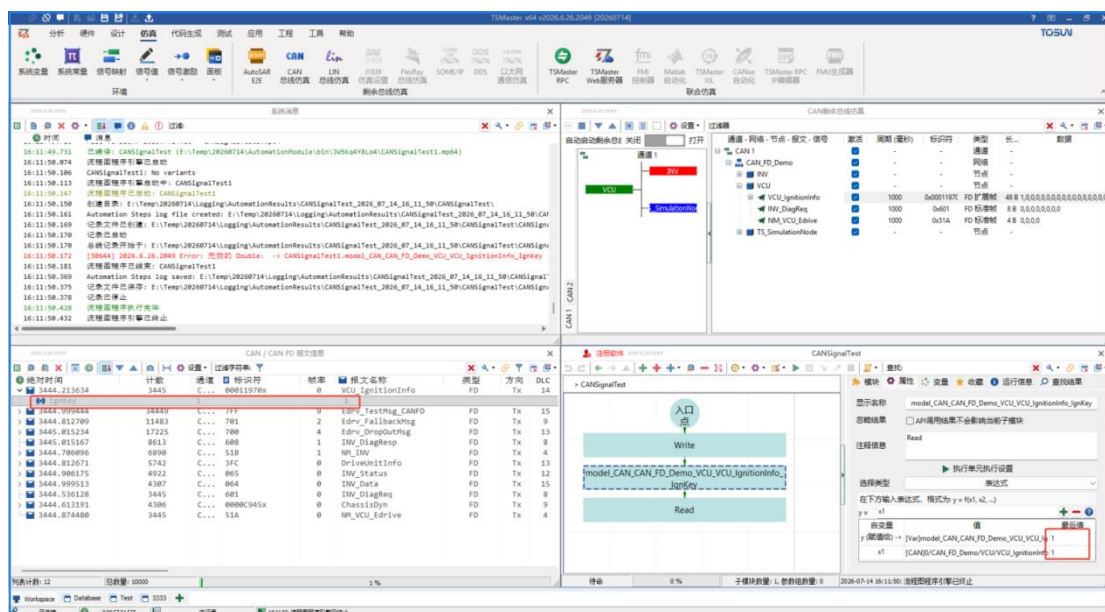


图22 执行用例步骤 2

6 附录

测试步骤的详细功能通过该章节进行说明，可以通过查看具体步骤小节来查看该步骤的功能和对应的配置项。

6.1 读写步骤说明

6.1.1 读步骤

该步骤用于读取信号值，信号类型包括 TSMaster 的系统变量和总线信号。读取后的值可以存放到本地变量中，本地变量可以用于其它类型的步骤例如流程控制中的 While、IfThenElse 等，也可以对读取到的值进行评估。

6.1.1.1 数值类型和字符串类型

Read 步骤数值类型和字符串类型中测量的比较符号支持范围不同，包含如下配置：

- 映射名称：定义一个别名，用来访问指定路径的信号
- 存值：用于控制是否保存读取到的信号值。开启后，读取值会赋值到本地变量中
- 评估：用于检查读取值是否符合预期。通过测量配置评估条件，通过误差配置误差范围，通过时间选项配置评估时间
 - 比较：包括测量（数值类型：==、!=、>=、<=、>、<。字符串类型：identical、starts with、ends with、contains、does not contain、dissimilar）和误差（数值类型：无、绝对值）
 - 时间选项：包括类型、Timeout、Minimum duration 和 Polling cycle，其中时间选项配置类型可选范围：
 - None
 - Wait until true（持续读取并评估，直到条件为真或超时）
 - True for at least（条件必须至少持续为真一段时间）

- True for at least ...within ...（在总超时（Timeout）时间内，条件必须至少持续为真一段时间）
- Synchronization only (no evaluation)

Read

映射名称*
CarSpeed

路径
/0/CAN_FD_Powertrain/Engine/ABSdata/CarSpeed

单位

数据类型
Double

注释说明

☒ 存值
carSpeedValue

☒ 评估

比较 时间选项

测量 == 60

误差 绝对值 10

确定 取消

图23 Read步骤数值类型

比较 时间选项

☐ None

☐ Wait until true

☐ True for at least

☒ True for at least ... within ...

☐ Synchronization only(no evaluation)

☐ Execute all n iterations

Timeout: 1000 ms

Minimum duration: 200 ms

☐ Polling cycle: ms

n: 1

确定 取消

图24 时间选项

6.1.1.2 数组类型

当数据类型为数组时，可以为每个元素单独设置比较规则，也可以统一设置。

当选择 Single 时，可以通过修改数组长度来设置需要比较的数组长度，并通过点击数组索引下方的单元格来设置评估规则

当选择 Common 时，通过点击【编辑】按钮来设置数组中所有元素的通用评估规则

数组类型对比数值类型多了如下配置：

- 评估类型（Single 或 Common）：当选择 Single 时，可以为数组的每个元素单独设置比较规则；当选择 Common 时，数组的所有元素共用一个规则
- 数组长度
- 数组中每个元素的评估设置

评估配置界面截图：

数组长度	0	1	2	3	4
5	1	$\$var0 + 1$	$\$var0 + 100$	$\$var0 ** 2$	$1 + 1$

评估类型：☒ Single ☐ Common [0] 编辑

图25 Read步骤数组类型评估

6.1.2 写步骤

该步骤用于写入信号值，信号类型包括 TSMaster 的系统变量和总线信号。

Write 对话框截图：

映射名称* Gear_1

路径 Gear_1

单位

数据类型 Double

注释说明

值* 120

确定 取消

图26 Write步骤

6.2 测试库步骤说明

测试库包含固定的测试步骤，包括数据操作、流程控制、通用步骤和通过 TSF MU 资源包加载的小程序 API。

测试库以树节点的形式显示，可以通过【刷新】按钮重新加载小程序 API 资源，通过展开按钮展开所有树节点，通过收拢按钮收起所有树节点。

在树节点中选中目标项并拖拽到用例编辑区后，即可添加新的用例步骤。

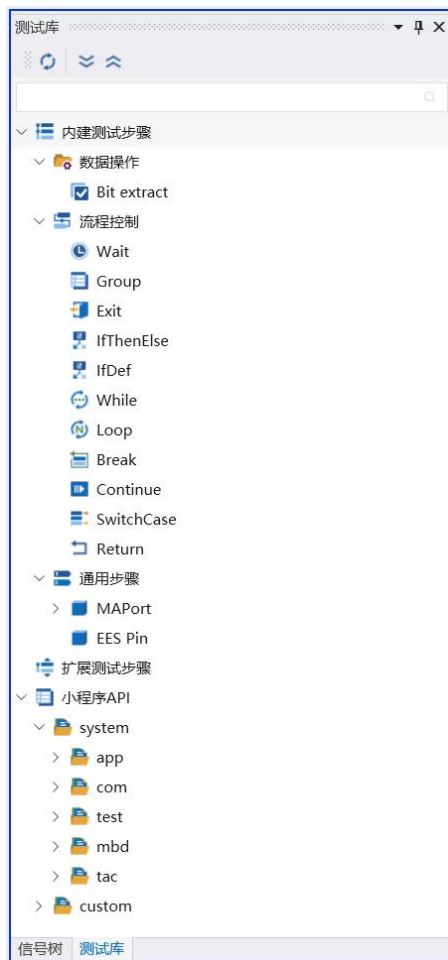


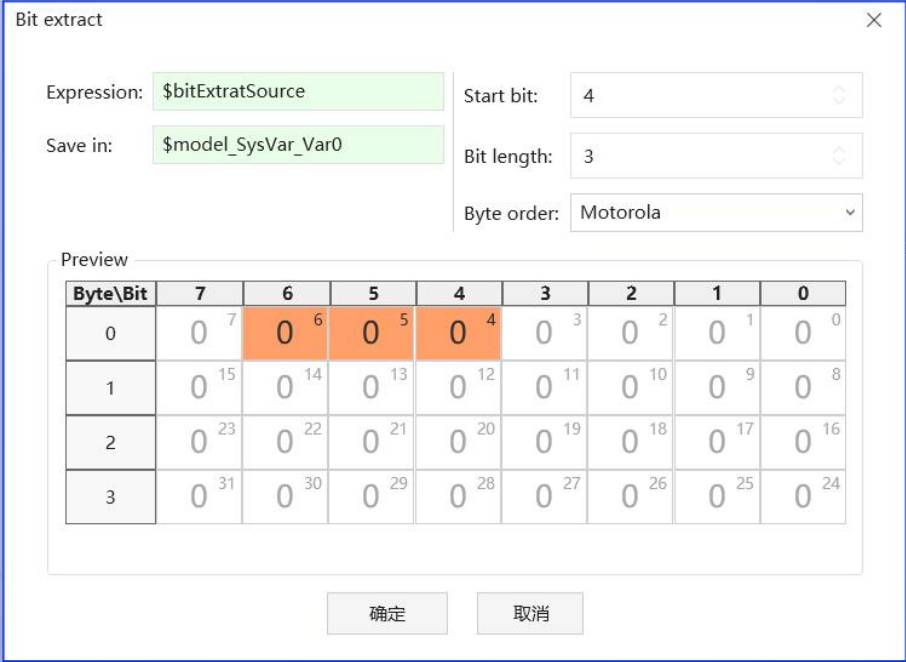
图27 测试库

6.2.1 数据操作

数据操作包含 Bit extract 步骤。该步骤用于按照配置的 Start bit、Bit length、Byte order 属性，从本地变量或常量中提取数据并赋值到另一个本地变量中；预览区会显示命中的位。

【Expression】输入框可以输入常量，也可以使用 \$ 符号引用本地变量

【Save in】输入框用于填写提取后数据存放的变量



The 'Bit extract' dialog box is used to configure bit extraction. It includes fields for 'Expression' (set to '\$bitExtratSource'), 'Start bit' (4), 'Bit length' (3), and 'Byte order' (Motorola). The 'Save in' field is set to '\$model_SysVar_Var0'. A 'Preview' section shows a 4x8 grid of bits, with the first four bits of the first row highlighted in orange.

Byte\Bit	7	6	5	4	3	2	1	0
0	0 ⁷	0 ⁶	0 ⁵	0 ⁴	0 ³	0 ²	0 ¹	0 ⁰
1	0 ¹⁵	0 ¹⁴	0 ¹³	0 ¹²	0 ¹¹	0 ¹⁰	0 ⁹	0 ⁸
2	0 ²³	0 ²²	0 ²¹	0 ²⁰	0 ¹⁹	0 ¹⁸	0 ¹⁷	0 ¹⁶
3	0 ³¹	0 ³⁰	0 ²⁹	0 ²⁸	0 ²⁷	0 ²⁶	0 ²⁵	0 ²⁴

图28 Bit extract 步骤

6.2.2 流程控制

流程控制包含 Wait、Group、Exit、IfThenElse、IfDef、While、Loop、Break、Continue、SwitchCase、Return 等步骤。

6.2.2.1 Wait

Wait 步骤包含延迟时间和单位两个参数，用于等待指定时长；单位可选范围为 ms、s、min、h。



The 'Wait' dialog box is used to set a delay. It has a '延迟时间*' (Delay Time) input field with the value '1' and a unit dropdown menu set to 'ms'. There are '确定' (OK) and '取消' (Cancel) buttons at the bottom.

图29 Wait 步骤编辑

6.2.2.2 Group

Group 步骤不会执行实际动作，仅用于用例分组和容纳子步骤。该步骤包含步骤/名称和期望结果两个重要参数，可用于描述用例名称和预期。

图30 Group 编辑

6.2.2.3 Exit

该步骤用于退出当前用例的执行。

6.2.2.4 IfThenElse

IfThenElse 为条件分支语句，包含一个条件表达式参数，以及 Then 和 Else 两个子步骤。当条件或条件结果为 True 时，执行 Then 中的步骤；否则执行 Else 中的步骤。条件表达式中可使用 \$ 符号引用本地变量。

图31 IfThenElse 编辑

6.2.2.5 IfDef

IfDef 为条件语句，包含一个表达式参数。当表达式或表达式结果为 True 时，执行 IfDef 中的子步骤。



图32 IfDef 编辑

6.2.2.6 While

While 为循环语句，包含一个循环条件参数。当循环条件或循环条件结果为 True 时，执行 While 步骤中的子步骤；可以使用 Break 步骤退出循环，或使用 Continue 步骤跳过后续子步骤并进入下一次循环。



图33 While 编辑

6.2.2.7 Loop

循环语句，包含一个循环次数参数，执行时将会按照配置的次数循环执行子步骤。



图34 Loop 编辑

6.2.2.8 SwitchCase

SwitchCase 为条件分支语句，包含一个表达式参数，用于控制分支执行。该步骤包含一个默认的 Default 分支，可以通过【+】、【-】按钮增加或删除分支，其中 Default 分支不可删除。

表达式中可以使用 \$ 引用本地变量，也可以使用常量控制分支执行。



图35 SwitchCase 编辑

6.2.2.9 Return

Return 为返回语句，包含 Rating 参数，用于控制返回状态；可选值为 NONE、SUCCESS、FAILED、ERROR。



图36 Return 编辑

6.3 通用步骤说明

6.3.1 MAPort

Start Simulation: 用于启动 XIL MAPort 的仿真，下拉框用于选择当前 TSMaster 工程中包含的 MAPort 名称。

Stop Simulation: 用于停止 XIL MAPort 的仿真，下拉框用于选择当前 TSMaster 工程中包含的 MAPort 名称。



图37 MAPort Simulation

6.3.2 EES Pin

电气错误模拟（Electrical Error Simulation, EES）是测试用例编辑器中用于模拟 ECU 线路电气故障的功能模块。EES Pin（电气错误模拟引脚）是该功能的核心概念，代表测试配置中所有错误模拟单元的逻辑引脚，是在测试用例中定义和操作电气错误的基本对象。

通过 EES Pin，可以在测试过程中向 ECU 的特定物理引脚注入各类电气故障，以验证 ECU 的诊断逻辑和故障处理能力。

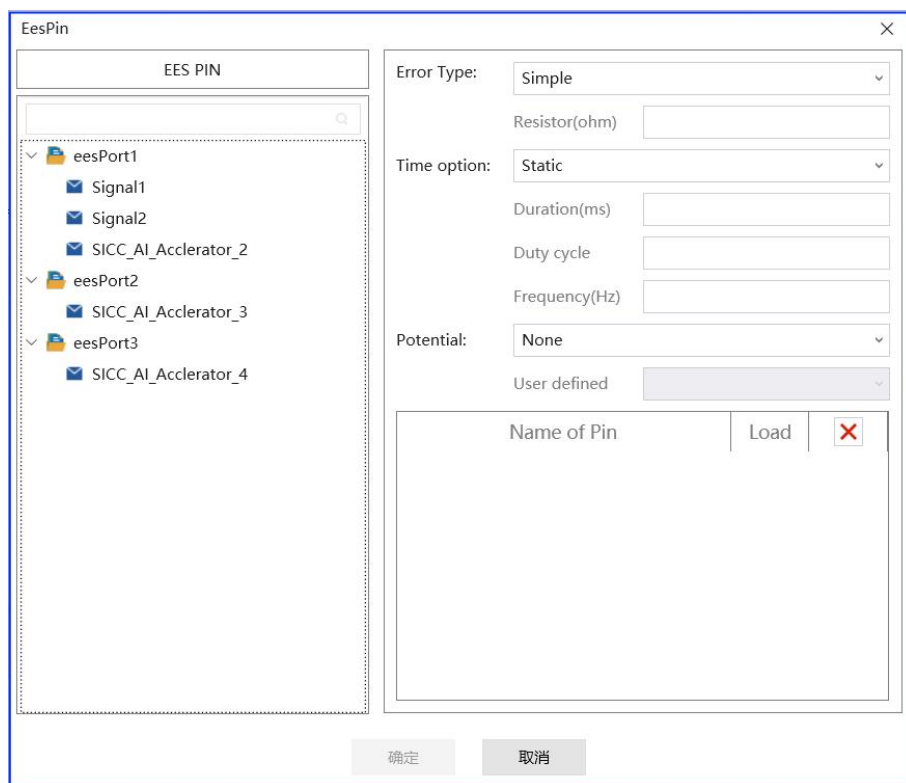


图38 EES Pin 步骤

EES Pin 步骤的配置参数：

- 错误类型（Error Type）：支持 Simple（简单错误）和 Resistor（电阻错误，需配置电阻值）两种
- 时间选项（Time Option）：支持 Static（静态持续）、Duration（持续指定时长）和 Loose contact（松接触，需配置占空比和频率）三种
- 电位类型（Potential Type）：支持 None（开路）、Ground（对地短路）、Ubatt（对电源短路）和 User defined（自定义电位）四种

信号选择从 EESPort 信号树中拖拽引脚完成，同一步骤仅能操作一个 Port。

6.4 小程序API说明

通过小程序 API 步骤，可以调用 TSMaster 提供的接口。API 方法从测试库中拖拽到用例编辑区后，会自动弹出编辑对话框。

可配置参数包括：

- 参数列表：参数名、类型以及值输入框
- 存值：可将接口执行结果保存到本地变量中
- 评估
 - 测量（用于检查接口返回值是否符合预期）
 - 误差（用于设置检查值时的误差范围）

参数名	类型	值*
A1	s32	0
A2	s32	0

图39 小程序 API

6.5 表达式说明

该节的适用范围为流程控制中的 IfThenElse、IfDef、While 等步骤中的条件表达式，读取步骤中的测量表达式等。下文将描述“表达式”支持的范围以及举例。



注：表达式中使用 \$ 符号引用变量。

- 字面常量：

- 整数：0x100、256、0b1010
- 浮点数：3.1415、-0.233
- 字符串：

```
'hello world'
"hello world"
```

- 布尔值：True、False
- 空值：None

- 算术运算：加法、减法、乘法、除法、整除、取模、幂运算、混合运算

- 位运算：按位与、按位或、按位异或、按位取反、左移、右移

- 逻辑运算（返回布尔值）：逻辑与、逻辑或、逻辑非

- 比较运算（产生布尔值）：

- ==、!=、>、<、>=、<=
- 身份比较：

```
$var0 is None
$var1 is not True
```

- 成员测试：

```
$state in ('on', 'off')
$value not in [0, 255, 512]
```

- 条件表达式（三元运算）

```
$var0 if $model_SysVar_Var0 > 0 else 0
($signal + $offset) if $enabled else $default
$high_limit if $mode == 1 else $low_limit
```

- 序列操作（索引、切片）：当变量为数组等可索引对象时使用：

- 索引访问

➤ 切片：

```
$data[1:3] #索引1到2的子序列  
$data[:4] #前四个元素  
$data[::2] #每隔一个元素取一个
```

- 括号控制优先级：使用小括号 () 可以改变运算顺序，并可任意嵌套。

7 常见问题及解决方法

7.1 Git 无访问权限

当测试用例编辑器安装到默认路径：C:\Program Files (x86)\ 时，从 Git 拉取项目到本地文件夹可能会报 Git 无访问权限。

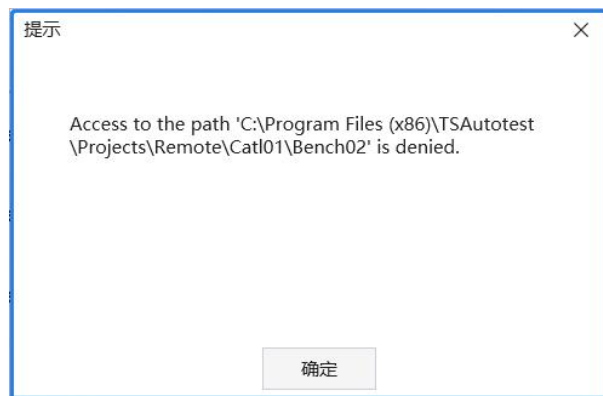


图40 Git 无访问权限

解决方案：

1. 修改文件夹权限（永久解决）。

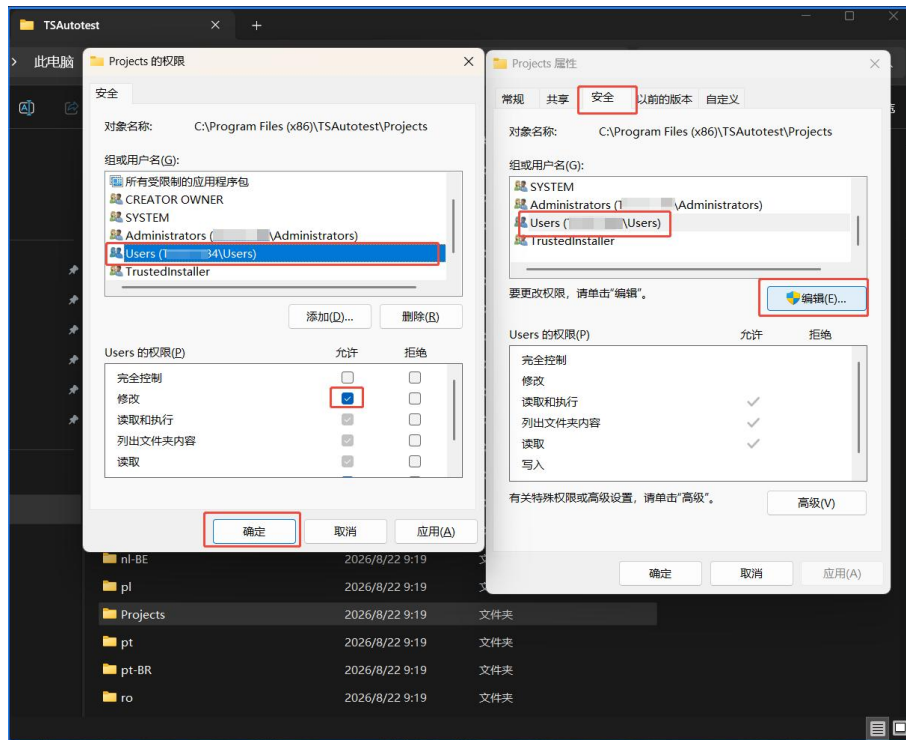


图41 设置目录权限

2. 安装到非默认路径：安装到其他文件夹，如：C:\Soft\TSAutotest。

8 联系方式

上海同星智能科技有限公司

总部：上海市嘉定区嘉松北路1288号9号楼

上海研究院：曹安公路4849弄14-17栋

网站： www.tosunai.com

联系电话：021-59560506

Email: support@tosunai.cn

软件 TSMMASTER

UDS诊断 / ECU刷写 / CCP/XCP标定

嵌入式代码生成 / 应用发布/加密发布 / 记录与回放

图形化编程 / 剩余总线仿真 / C/Python脚本

总线监控/发送 / SOME/IP和DoIP / 自动化测试



扫码关注
获取软件下载链接



硬件

1/2/4/8/12通道CAN FD/CAN转USB/PCIe工具

1/2/6通道LIN转USB/PCIe工具

多通道FlexRay/CAN FD转USB/PCIe工具

多通道车载以太网/CAN FD转USB/PCIe工具

车载以太网介质转换工具 (T1转Tx)

多通道CAN FD/Ethernet/LIN记录仪

TTS测试系统(通信板卡、数字/模拟量板卡等)

CAN

CAN

lin

FlexRay



解决方案

总线一致性 / 网络自动化测试系统 / 充电测试系统

EMB标定测试设备 / 信息安全解决方案

FCT/EOL测试设备 / 线控底盘测试解决方案

汽车“四门两盖”试验解决方案

电机性能 / 耐久试验解决方案



关于我们

同星智能的核心软件TSMaster及配套硬件设备，
具备嵌入式代码生成、汽车总线分析、仿真、测试及诊断、标定等核心功能，
覆盖了汽车整车及零部件研发、测试、生产、试验、售后全流程。

国际组织
ASAM, CiA

质量保证
ISO9001:2015

CE认证

愿景

解决一切工程难题！

联系我们

021-59560506
marketing@tosunai.cn

访问官网

www.tosunai.com

