



知从 ASPICE SWE04 单元测试介绍

ZC ASPICE SWE04 UNIT TESTING

INTRODUCTION

知从工程服务

ZC ENGINEERING SERVICE

知从 ASPICE SWE04 单元测试介绍

ZC ASPICE SWE04 UNIT TESTING

INTRODUCTION

知从工程服务 ZC ENGINEERING SERVICE

1 概述 OVERVIEW

在汽车基础软件开发过程中，测试的重要程度也日趋明显。测试用来检测该功能是否按照预期实现，测试是对软件质量的一个度量，通过前期的测试，尽早的发现软件中的 BUG，进而提高软件质量，保证产品质量。

In the development process of basic automotive software, the importance of testing has become increasingly obvious. Testing is used to detect whether the function is implemented as expected. Testing is a measure of software quality. Through early testing, bugs in the software can be discovered as soon as possible, thereby improving software quality and ensuring product quality.

测试链基本步骤如下：

The basic steps of the test chain are as follows:

- 针对代码的静态测试
Static testing of code
- 针对详细设计的单元测试
Unit testing for detailed design
- 针对架构的集成测试
Integration testing for architecture
- 针对需求的系统测试
Demand-based system testing

2 单元测试-知从 QAC 测试服务 STATIC TESTING-QAC TESTING SERVICES

静态测试通过分析源程序的语法、结构、过程以及接口等来检查源程序的正确性、可读性、可维护性。知从提供的静态测试主要是利用 QAC 根据 MISRA C 规范对静态代码的一个规范检测，并提供相应详细测试报告。

Static testing checks the correctness, readability, and maintainability of the source program by analyzing the syntax, structure, process, and interface of the source program. The static test provided by Zhicong mainly uses QAC to test static code according to the MISRA C specification, and provides corresponding detailed test reports.

QAC 测试可检测的度量：

Metrics that can be detected by QAC testing:

➤ 函数圈复杂度

STCYC Function cyclic complexity STCYC

➤ 路径复杂度

STPTH Path complexity STPTH

➤ 嵌套深度

STMIF Nested depth STMIF

➤ IF ELSE 数量

STELF IF ELSE Quantity STELF

➤ 其他度量

other metrics

静态测试的优点：

Advantages of static testing:

➤ 发现程序中的语法错误

Found syntax errors in the program

➤ 检查软件是否符合编程标准

Check whether the software meets programming standards

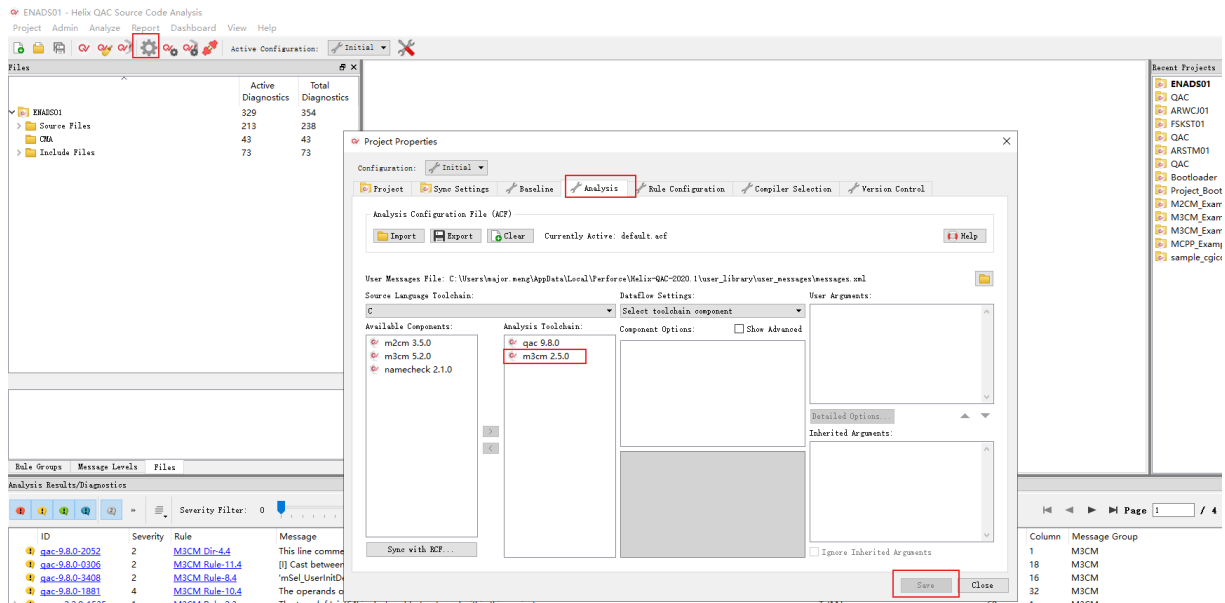


图 1: QAC 静态分析界面

Figure 1: QAC static analysis interface

QAC 测试中可以检测多种度量，通过选择需要检测的度量，可以将代码中每个函数的度量结果以饼图方式显示在界面上。

A variety of metrics can be tested in QAC testing. By selecting the metrics to be tested, the measurement results of each function in the code can be displayed on the interface in a chart format.

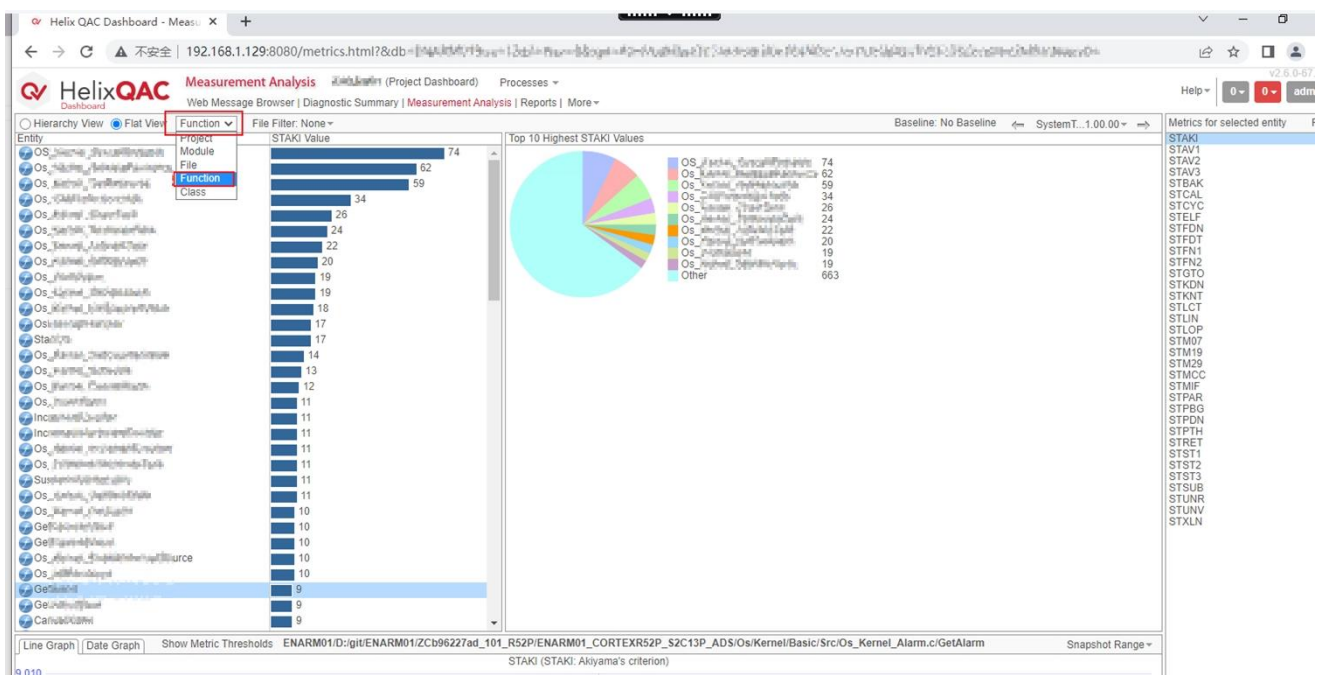


图 2: QAC 各函数度量

Figure 2: Measurement of various functions in QAC

The source file reference nesting relationships shown can help reduce unnecessary nested calls and reduce the complexity of code calls.

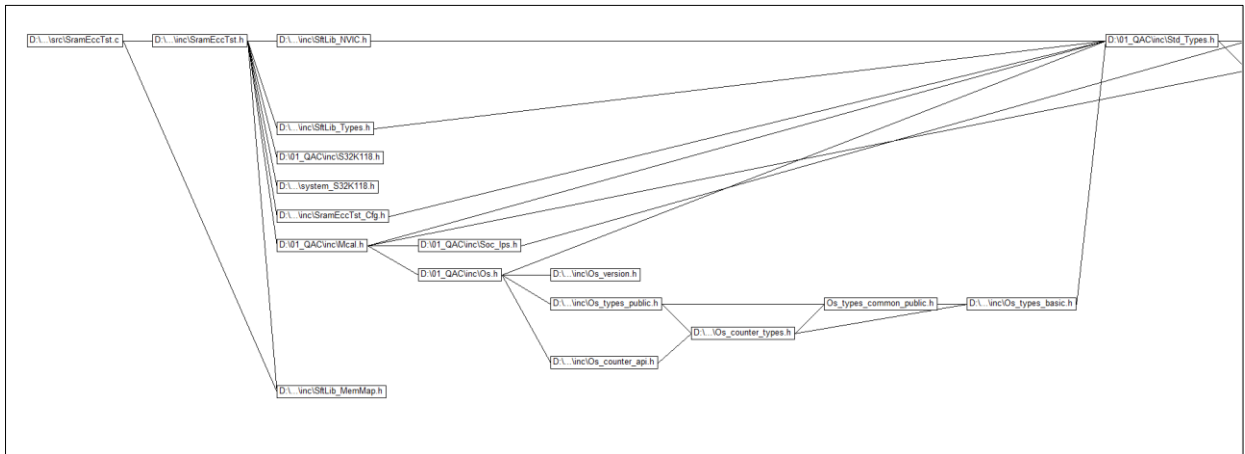


Figure 3: The file nesting relationship

The QAC static analysis report provided by the test includes Metrics, a list of MISRA errors that violate programming specifications, and an error list-used to display QAC errors that appear in the file and recommendations for correction.

1. 范围

静态代码检查的源代码文件，罗列如下：

文件名	修订版本
SMPU.c	00.01.00

第 1 页

2. 工具

静态代码检查使用的工具，如下：

工具名	工具版本
QAC	7.2 R

3. 工具设定

例：参照《编码规范》中定义 MISRA-C：2004 及 Source Code Metric，定义工具检查规则

工具的配置文件，如下：(添附工具配置文件)

>	封面	变更记录	Scope_Method	Metrics	MISRA	误差列表	+
---	----	------	--------------	---------	-------	------	---

图 4：QAC 静态分析报告

Figure 4: QAC static analysis report

3 单元测试-知从 POLYSPACE 测试服务 UNIT TESTING-LEARN FROM POLYSPACE TESTING SERVICES

Polyspace 核心优势显著，覆盖度统计精准高效，多维度指标自动生成可视化报告；静态缺陷检测能力强，无需执行代码即可精准识别多种潜在问题；全流程可追溯性好，与多工具集成实现测试链路贯通。同 QAC 一样，Polyspace 可以根据 MISRA C 规范对静态代码的一个规范检测，可以根据客户要求对特定规则检查，并提供相应详细测试报告。

Polyspace has significant core advantages, with accurate and efficient coverage statistics, automatically generating visual reports with multi-dimensional indicators; it has strong static defect detection capabilities, accurately identifying various potential issues without the need for executable code; it has good full-process traceability, integrating with multiple tools to achieve seamless testing links. Similar to QAC, Polyspace can perform specification detection on static code according to the MISRA C standard, check specific rules according to customer requirements, and provide corresponding detailed test reports.

Polyspace 测试可检测的度量：

Metrics that can be detected by Polyspace testing:

- 函数圈复杂度
STCYC Function cyclic complexity STCYC
- 路径复杂度
STPTH Path complexity STPTH
- 嵌套深度
STMIF Nested depth STMIF
- IF ELSE 数量
STELF IF ELSE Quantity STELF
- 其他度量
other metrics

在配置中对测试规则进行选择，就可以根据所选规则检查代码规范问题，所有扫描结果就会显示在输出摘要窗口并生成结果工程文件。

By selecting test rules in the configuration, you can check code specification issues according to the selected rules, and all scan results will be displayed in the output summary window and a result project file will be generated.

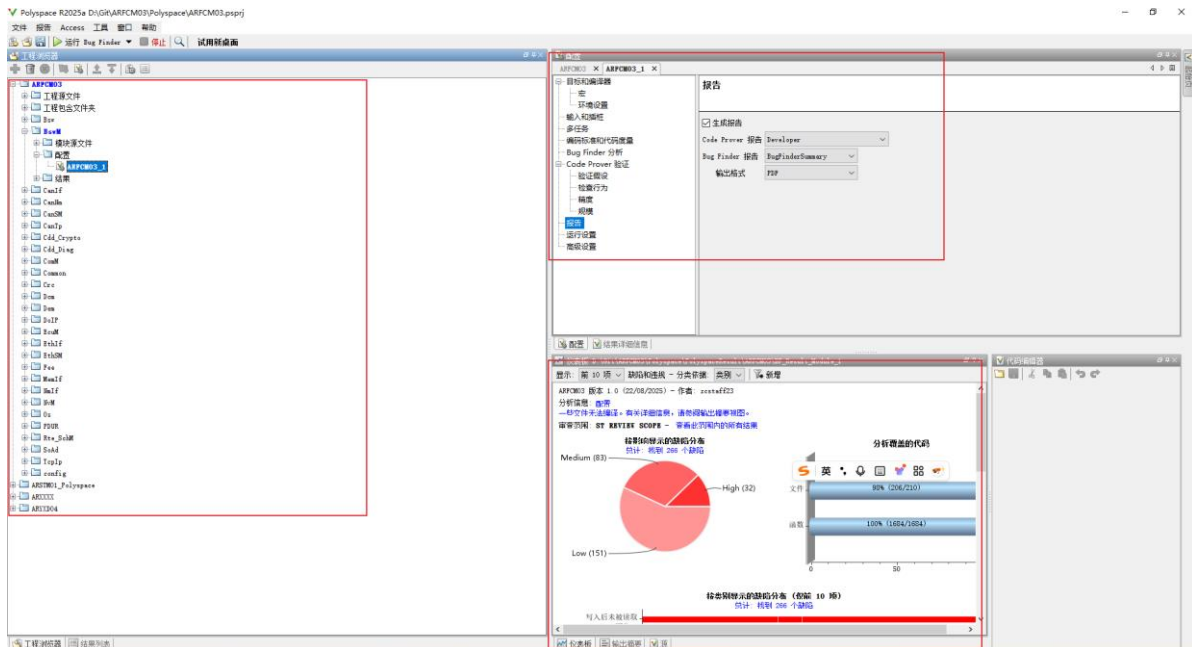


图 5: Polyspace 单元测试界面

Figure 5: Polyspace unit test interface

检查输出摘要窗口，保证无报错信息，确保所有文件被完整分析。

Check the output summary window to ensure there are no error messages and that all files have been fully analyzed.

输出摘要 D:\Git\ARFCM03\Polyspace\PolyspaceResult\ARFCM03\BP_Result_Module_1

总历时: 00:15:43 已完成分析过程

类型	消息	文件	行	列
警告	could not find include file "Os_MemMap.h"	Os_Types.h	742	
警告	could not find include file "Os_MemMap.h"	Os_Cfg.h	527	
警告	could not find include file "Os_MemMap.h"	Os_Cfg.h	533	
警告	could not find include file "Os_MemMap.h"	Os_Cfg.h	540	
警告	could not find include file "Os_MemMap.h"	Os_Cfg.h	543	
警告	could not find include file "Os_MemMap.h"	Os_Cfg.h	546	
警告	could not find include file "Os_MemMap.h"	Os_Cfg.h	549	
警告	could not find include file "Os_MemMap.h"	Os_Cfg.h	559	
警告	could not find include file "Os_MemMap.h"	Os_Cfg.h	567	

详细信息

图 6: Polyspace 单元测试输出摘要视图

Figure 6: Polyspace Unit Test Output Summary View

Polyspace 的输出摘要界面可以观测到代码中的所有缺陷信息以及 Misra 规则违反情况，以便于开发人员修改代码规范问题。

The output summary interface of Polyspace can observe all defect information in the code and Misra rule violations, so that developers can modify code specification issues.

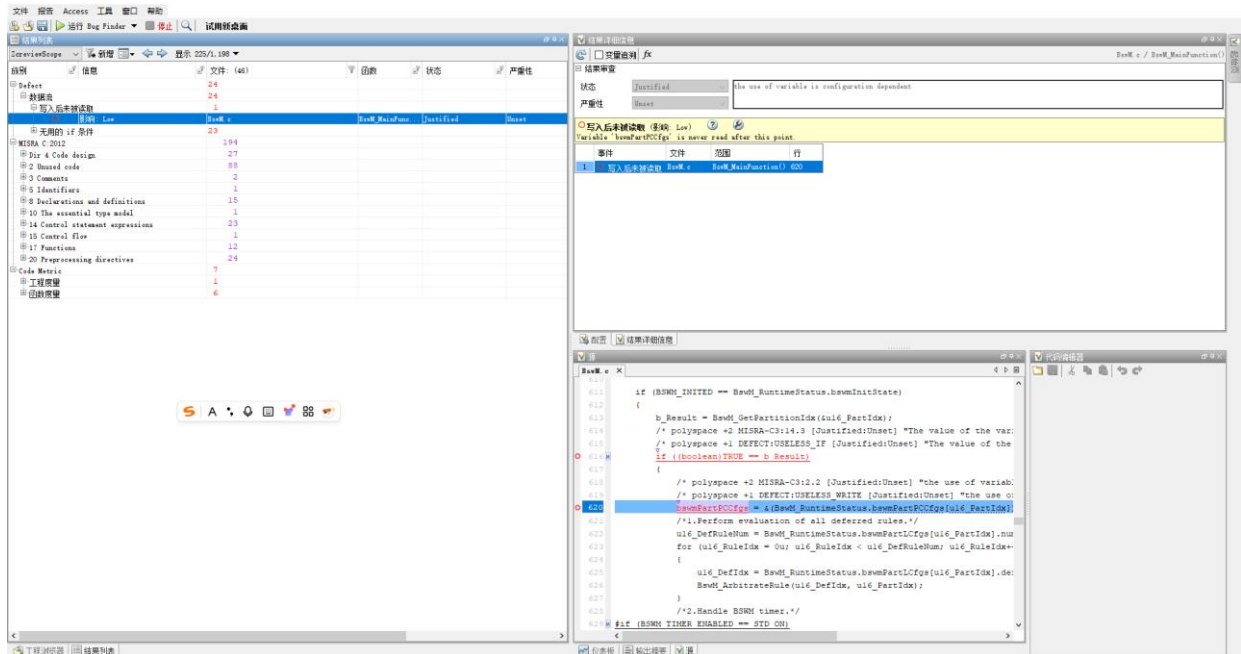


图 7：Polyspace 结果列表

Figure 7: Polyspace Results List

➤ 4 单元测试-知从 TESSY 测试服务 UNIT TESTING-LEARN FROM TESSY TESTING SERVICES

单元测试是软件开发 V 模型中重要的一环（如图 5），在完成编码工作后，测试工程师根据详细设计进行单元测试，用来验证函数的功能是否按照详细设计来实现，以及通过单元测试来发现潜在 BUG。为此知从推出 Tessy 单元测试服务（C 语言），包括 PC 端模拟测试和实际芯片环境的测试。

Unit testing is an important part of the software development V model (see Figure 5). After completing the coding work, the test engineer conducts unit testing according to the detailed design to verify whether the function of the function is implemented according to the detailed design, and through unit testing. Discover potential bugs. To this end, we launched Tessy unit test service (C language), which includes PC-side simulation testing and actual chip environment testing.

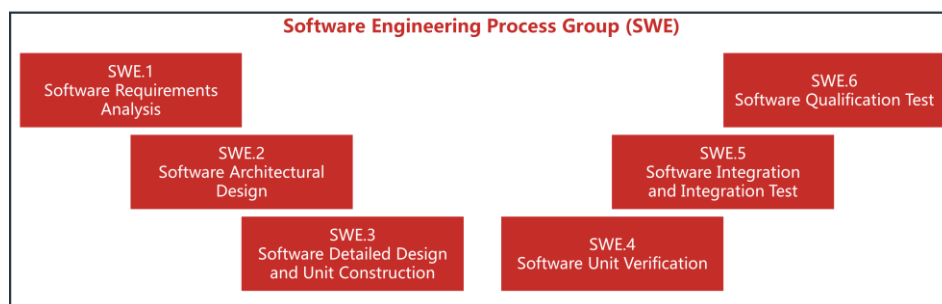


图 8：软件 V 模型开发部分

Figure 8: Software V model development part

单元测试的作用：

Role of unit testing:

➤ 检测功能错误

Detection function error

➤ 检测功能遗漏

Missing detection function

➤ 检测超出需求中的部分

Detect parts that exceed requirements

Tessy 的实际芯片测试环境搭建很复杂，知从也提供环境搭建服务。

Tessy's actual chip test environment is complex, and Zhicong also provides environment construction services.

知从目前可提供搭建的环境：

ZC currently provides the environment for building:

目标芯片 Target chip	调试器类型 Debugger type	编译器类型 Compiler type
MPC5744P	TRACE32 2019.02	Wind River5.9.4.0
MPC5744P	iSystem 9.17.39	Wind River5.9.4.0
MPC5746C	TRACE32 2019.02	Wind River5.9.4.0
MPC5746C	iSystem 9.17.39	Wind River5.9.4.0
MPC5748G	TRACE32 2019.02	Wind River5.9.4.0
MPC5748G	iSystem 9.17.39	Wind River5.9.4.0
S32K144	TRACE32 2019.02	gcc-arm-none-eabi-4_9
S32K144	iSystem 9.17.39	gcc-arm-none-eabi-4_9
S32K118	TRACE32 2018.02	IAR_ARM_V8.3
TC275	TRACE32 2019.02	HighTec4.6.6.1
TC275	iSystem 9.17.39	HighTec4.6.6.1
TC275	TRACE32 2019.02	Tasking4.2r2
TC275	iSystem 9.17.39	Tasking4.2r2

Cypress CYT4BF	I-jet	IAR_ARM_V8.2
----------------	-------	--------------

根据提供的设计需求设计相应的测试 Case，检测代码实现的功能是否满足需求，保证功能与需求的可追溯性。

Design corresponding test cases according to the provided design requirements to test whether the functions implemented by the code meet the requirements and ensure the traceability of functions and requirements.

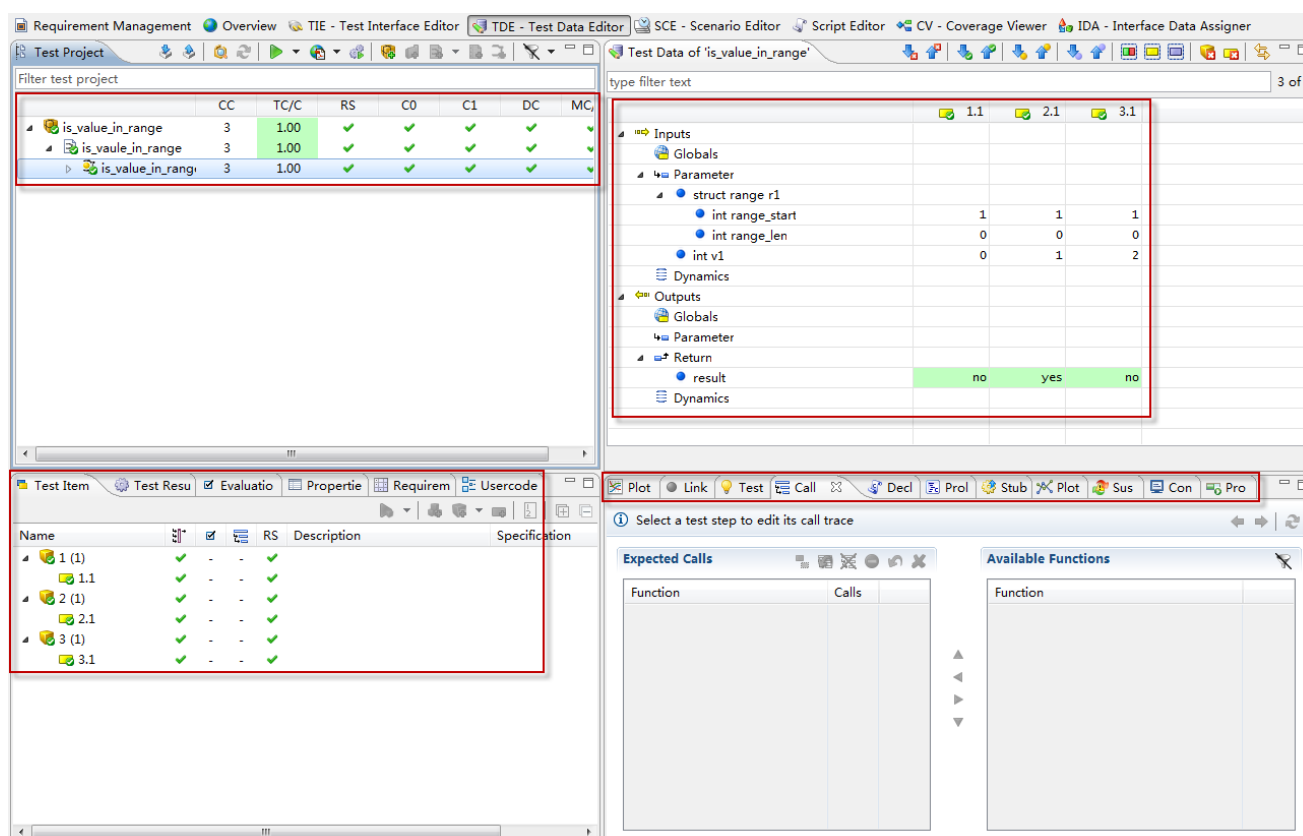


图 9：Tessy 单元测试界面

Figure 9: Tessy unit test interface

检测代码的路径覆盖度是否能够满足，保证代码中所有路径都能正常运行。

Check whether the path coverage of the code can be satisfied to ensure that all paths in the code can operate normally.

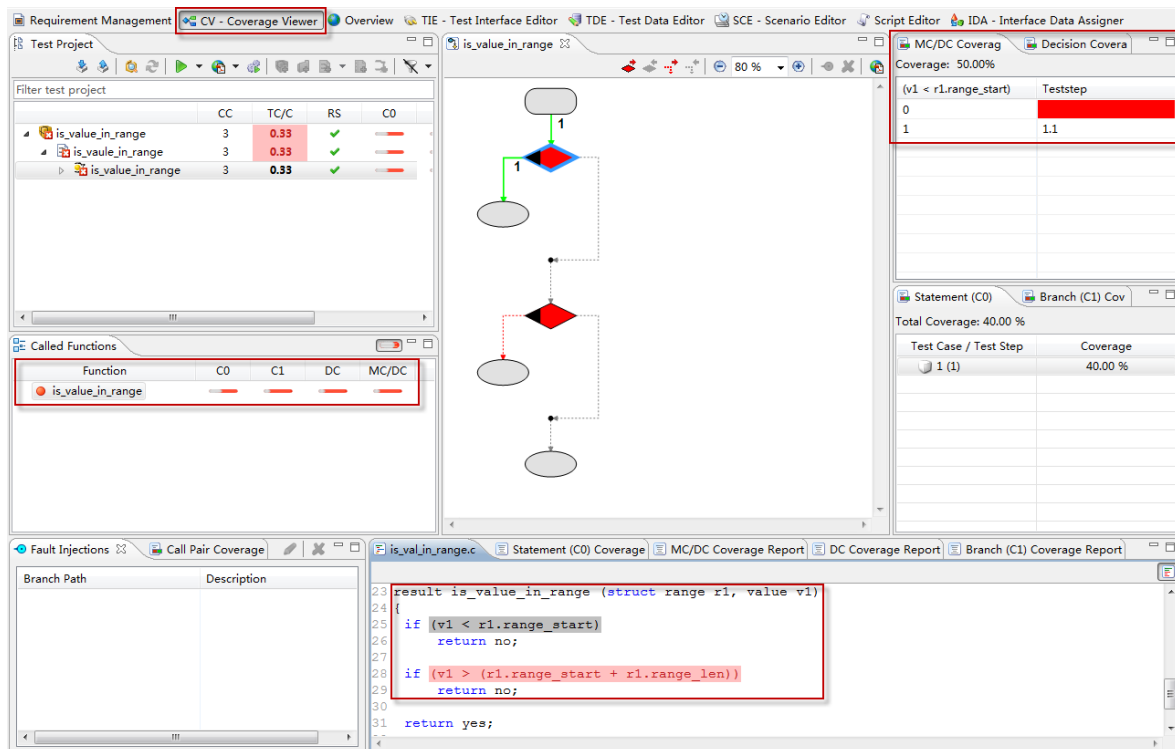


图 10: Tessy 单元测试测试用例覆盖度视图

Figure 10: Tessy unit test test case coverage view

搭建目标芯片在对应编译器与调试器下的实机环境，提供 xml 环境配置文件。

Build a real-machine environment for the target chip under the corresponding compiler and debugger, and provide an xml environment configuration file.

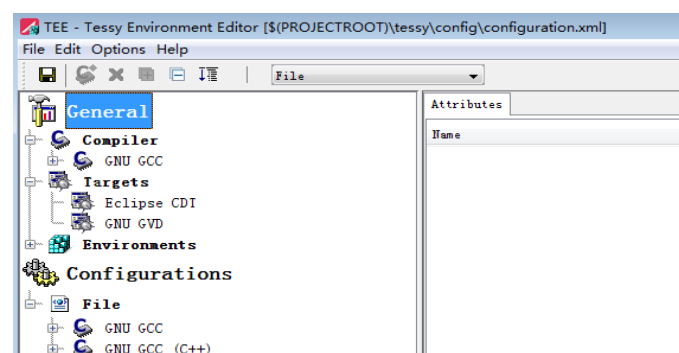


图 11: Tessy 实机环境搭建

Figure 11: Construction of Tessy's real-machine environment

Tessy 测试可以生成 4 种报告，包括 Details Report、Overview Report、Planning Coverage Report 和 Execution Coverage Report。

The Tessa test can generate four types of reports, including Details Report, Overview Report, Planning Coverage Report, and Execution Coverage Report.

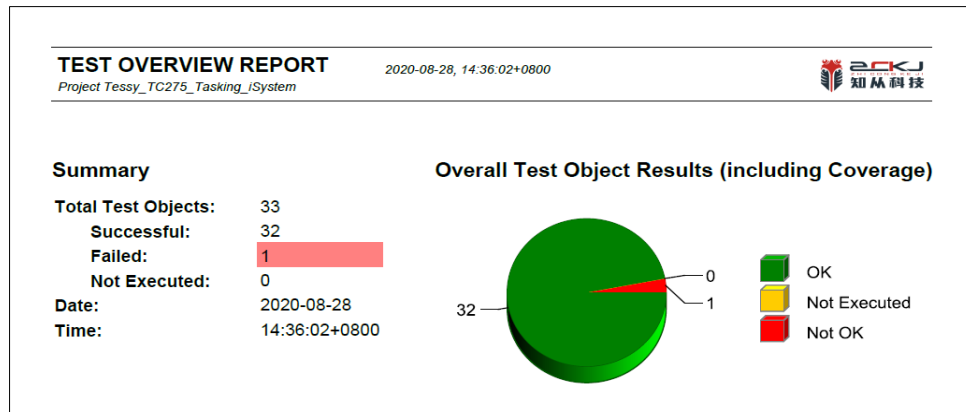


图 12: Tessa Overview 报告

Figure 12: Tessa Overview report

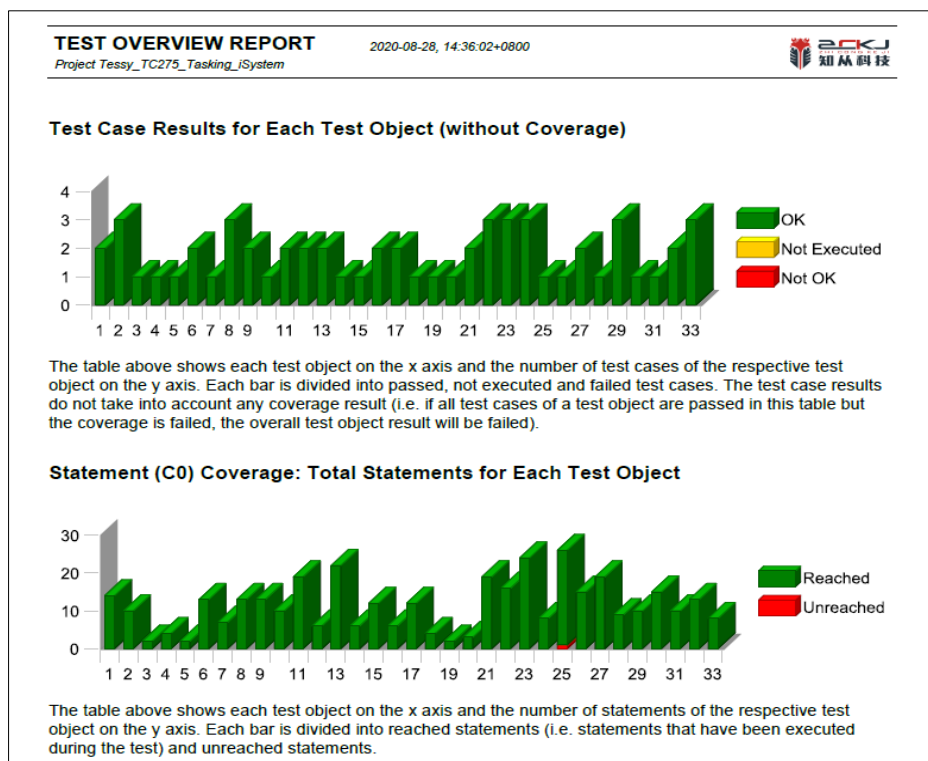


图 13: Tessa 测试路径覆盖度报告

Figure 13: Tessa test path coverage report



公众号
WeChat Official Account



业务联系
Business Contact

成为全球领先的汽车基础软件公司
To Be the Global Leading Automotive Basic Software Company

