

## CANdelaStudio\_CANoe.Diva Notes

[https://blog.csdn.net/weixin\\_44536482/article/details/99331306](https://blog.csdn.net/weixin_44536482/article/details/99331306)

Dela 和 Diva 都需要 license.

**CANoe.Diva 可以通过导入 cdd 或 odx 文件, 自动生成全面的测试用例; 再在 CANoe 中导入用例执行测试即可生成测试报告; 主要用于测试 CAN 的诊断功能。关于自动化诊断测试可以分为以下过程:**

- (1)、根据诊断规范进行 **cdd 文件的配置**; (关于这一部分的内容可查看"[关于 cdd 文件的配置](#)"一文)
- (2)、新建 Diva 工程, 导入 cdd 文件**生成测试用例**;
- (3)、新建 CANoe 工程, **导入前面生成的 Diva 工程**;
- (4)、接上 ECU, 执行测试案例; **生成测试报告**;
- (5)、根据测试报告进行**错误处理**。

## CANdelaStudio

<https://www.vector.com/cn/zh/products/products-a-z/software/candelastudio/>

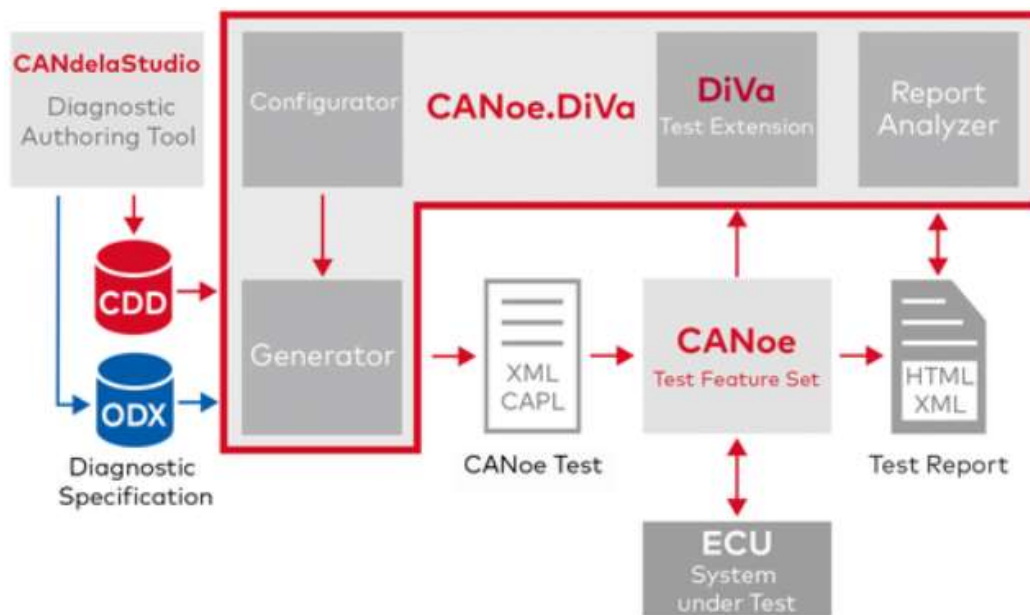
安装好 CANoe 后一般自带 CANdelaStudio.

CANdelaStudio lets you view, edit and export CANdela data (CDD), i.e. diagnostic data of control units in the automotive industry and in related areas.

CANdelaStudio especially provides import and export for [ODX](#). ODX files are also usable in the Vector diagnostic toolchain, but are better suited for use with test systems than for an ECU implementation.

With regard to requirements modelling, CANdelaStudio is used for detailed diagnostic specifications at the ECU-level:

- **Detailed specification of the diagnostic attributes for each diagnostic entity**
  - Example: what extended information (e.g. snapshot) is available for a DTC, mirror memory, etc.
  - Diagnostic entities: DTC, measurement, control...
  - This is complemented by the template-based nature of CANdela, which sets out, for example, the services that are to be provided by a given ECU.



Overview CANoe.DiVa - Architecture and Test Generation

汽车 ECU 诊断实现的质量保证与时间和成本密切相关。对于供应商来说，测试贯穿整个开发过程，同时还需要大量的回归测试和发布测试。对于 OEM 厂商来说，同样需要大量的集成测试和发布测试。

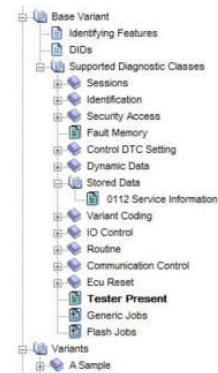
在汽车行业中，Vector 的 CANdela 诊断流程被广泛使用。许多 OEM 厂商都使用 CANdela 诊断流程中的核心工具 CANdelaStudio 所创建的 ECU 诊断描述文件。同时，带有“Test Feature Set”的 CANoe 已经成为汽车行业的标准开发工具。

## Important Objects

### ECU

- Variants (equipment /development over time)
  - DIDs
  - Diagnostic Classes („Diagnostic Use-Case“)
    - Diagnostic Instances
      - Services
        - Data Objects (reference a Data Type)
        - Fault Memory: DTCs, Snapshot Data, Extended Data, Events
- Data Types
- Template
  - Protocol Services
  - Diagnostic Class Templates

Services, Data Objects and DTCs are shown in the right side of the tree navigation.



## Jobs

A job is a program (macro, sequence) to be executed by a tester (not in the ECU).

See the [Jobs](#) feature.

## ODX

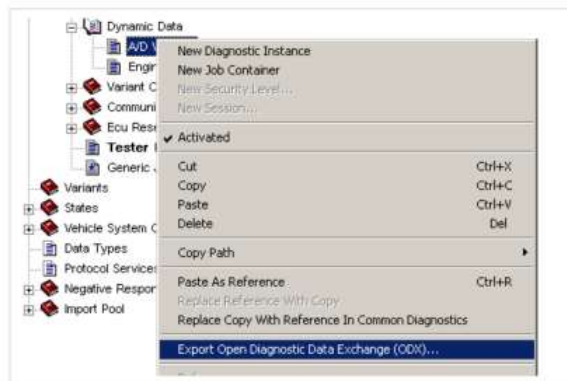
The **Open Diagnostic eXchange format** (ODX) is an XML-based ISO standard (ISO 22901-1:2008) for describing ECU data relevant for diagnostics.

Vehicle, ECU and tester manufacturers can use the ODX format to describe and exchange diagnostic data. ODX was designed as an open format for data exchange. It is a good choice for joint projects between automotive OEMs.

Export

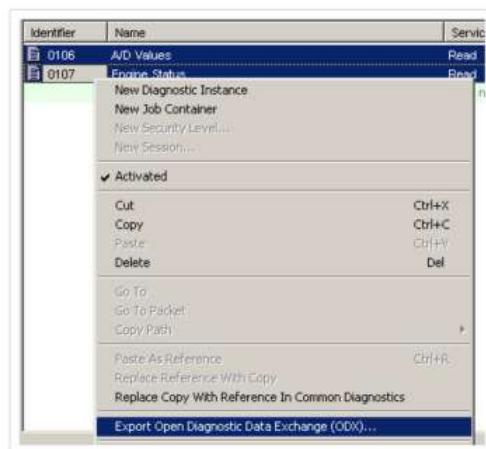
## How to select the elements to be exported

- Directly select one object [DCLUI-R500]:



Selective export of one object

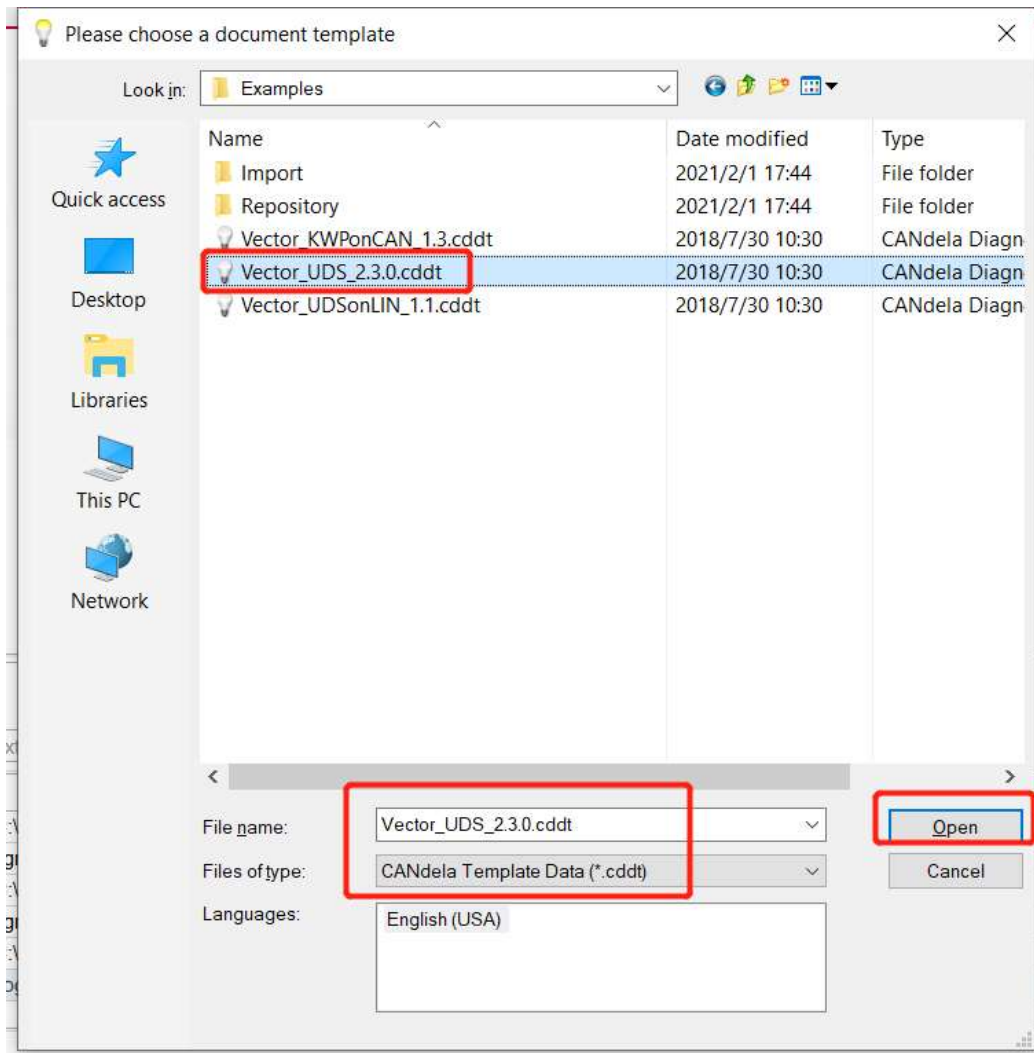
- Select a list of objects:  
In the diagnostic instances list, you can select multiple diagnostic instances for ODX export:



Selective export of multiple objects

## New Dela Project

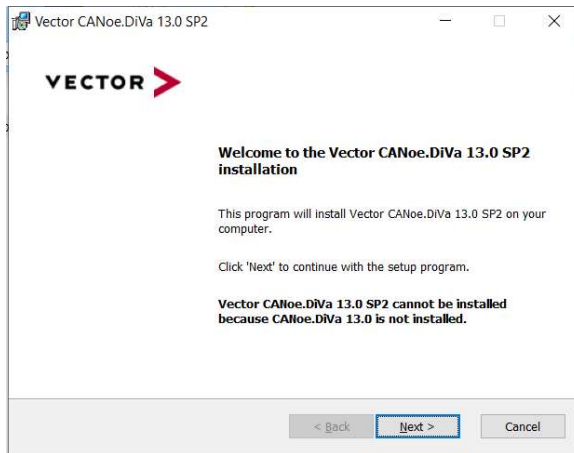
从.cddt 模板创建 Dela 工程, save 后工程文件后缀为.cdd.



## CANoe.Diva

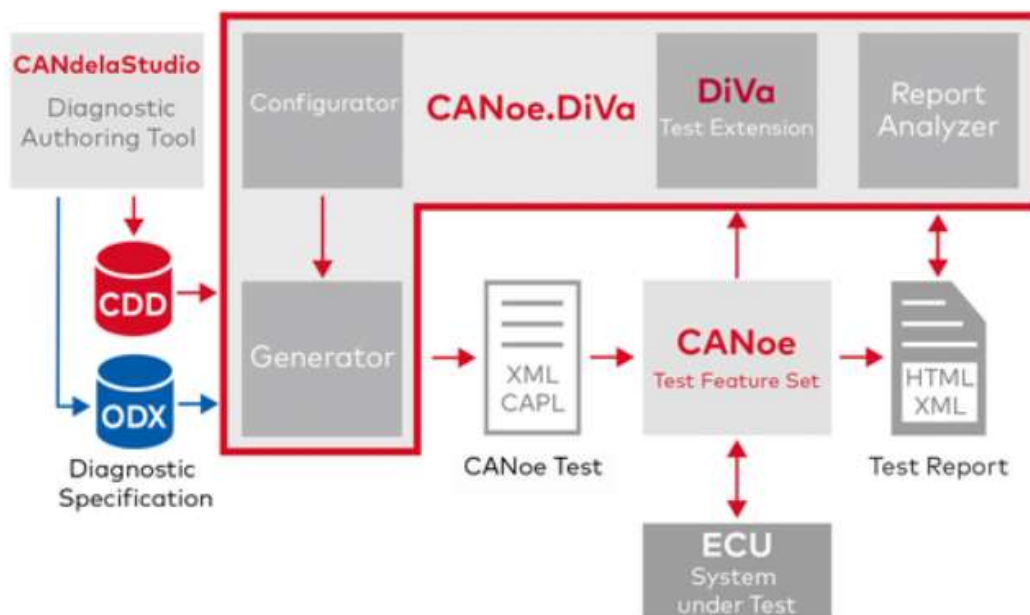
<https://www.vector.com/cn/zh/products/products-a-z/software/canoediva/>

需要去 Vector 下载 CANoe.DiVa, 然后安装.

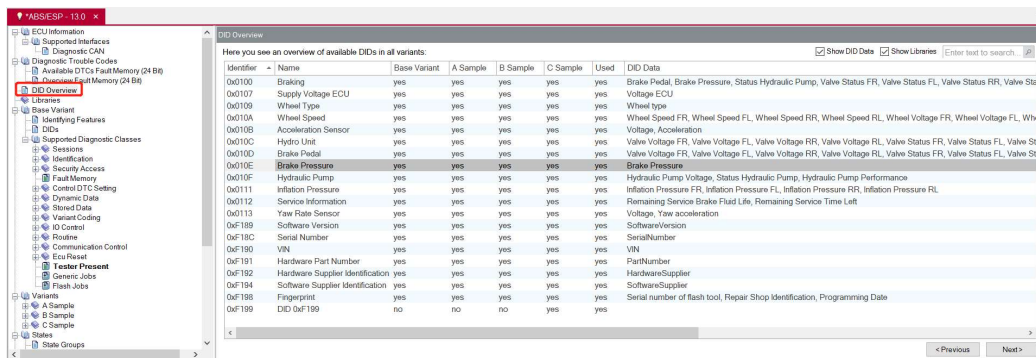
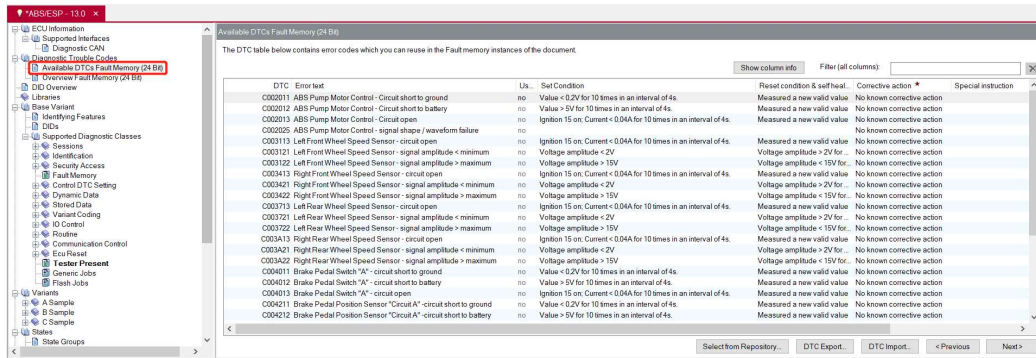


DiVa is a CANoe extension for automated testing of diagnostic software implementations in ECUs. Reproducible test cases are generated based on an ECU diagnostic description in CANDela or ODX format.

CANoe automatically executes these test cases and generates a conclusive test report.



Overview CANoe.DiVa - Architecture and Test Generation



DiVa（Diagnostic Integration and Validation Assistant）扩展了 CANoe 的功能，能够自动地为诊断协议的实现和集成测试生成测试用例。这些测试用例是根据 ECU 诊断描述文件来生成的，并在 CANoe 的“Test Feature Set”中执行。

DiVa 能够保证对 ECU 的诊断实现进行全面而详细的测试。

## CANoe.DiVa：产品特性概述

自动生成全面的测试用例和测试规范，且具有广泛的测试覆盖度

基于 ODX 或者 cdd 文件执行测试用例，并生成测试报告

支持不同的诊断标准（UDS、KWP2000、GMW3110、OBD）

轻易地完成测试范围的配置

完全集成在 CANdela 产品体系中

DiVa 的这些优点在节省时间和成本的同时提高了 ECU 诊断软件的质量。

## 功能

CANoe.DiVa 包含用户界面、测试用例生成器和一个用来扩展 CANoe 测试功能的运行时间库。DiVa 利用 ECU 诊断描述文件（cdd 文件）自动生成测试用例。

如果需要，DiVa 可以生成一份包含测试内容和测试顺序细节的测试规范。



DiVa 可以自动生成能够在 CANoe 中使用的测试环境。当执行测试后，CANoe 能够根据测试结果生成一份清晰而简明的测试报告。

详细功能：

根据 CANdela 的 ECU 诊断描述文件（cdd 文件）或者 ODX 文件自动生成 CANoe 测试模块  
测试单 ECU 的功能寻址

可选择测试重点（测什么？）

- o 诊断协议定时
- o 诊断协议格式
- o 数据内容
- o ECU 应用程序诊断接口
- o 会话和安全等级

可配置测试范围

- o 全面测试：每个服务都要用所有的测试用例进行测试（包括好用例和坏用例），例如回归测试

- o 单次测试：利用被选择的测试用例对每个服务进行一次测试
- o 快速测试：每个服务都要用所有好用例进行测试，对有代表性的服务用坏用例进行测试

测试

在诊断仪方面，从传输层级别决定通信参数的操作

修改生成的测试用例（例如排除某些服务，从而避免对控制系统产生无谓的改变或损坏）

根据测试用例生成清晰而详细的测试规范文档

扫描服务 ID 和子功能 ID 区域

由 CANoe 生成测试报告

通过有效的错误监测、验证和排除实现测试报告评估：如测试用例注释、测试结果分类、不同测试运行比较、报告提取（小报告）

通过 VT 系统以及 IL 模块自动设置和验证故障内存入口（DTC）

应用领域

CANoe.DiVa 可以用于以下领域（包括 OEM 和供应商）：

功能开发及软件开发

测试实验室及车辆集成

对于供应商来说，CANoe.DiVa 支持 ECU 诊断的执行。开发人员在开发过程中就可以进行测试，并在开发结束后进行大量的回归测试。这些回归测试同样可以作为 ECU 诊断部分发布流程中的一部分。

当供应商将 ECU 样品送交 OEM 时，OEM 可以使用 CANoe.DiVa 进行自动测试。同时 CANoe.DiVa 还支持开发平台和整车的集成测试。OEM 利用 CANoe.DiVa 可以很容易的根据质量需求创建测试规范或产生测试文档。

OEM 和供应商可以通过增加新的测试模式来扩展测试范围，也可以通过裁减现有的

测试模式来满足他们的特殊需求。

支持的诊断协议

基于 CAN 总线的诊断协议 ISO 15765

基于 CAN 总线的诊断协议 ISO 14229 (UDS)

基于 K 线的 ISO 14230 (KWP2000)

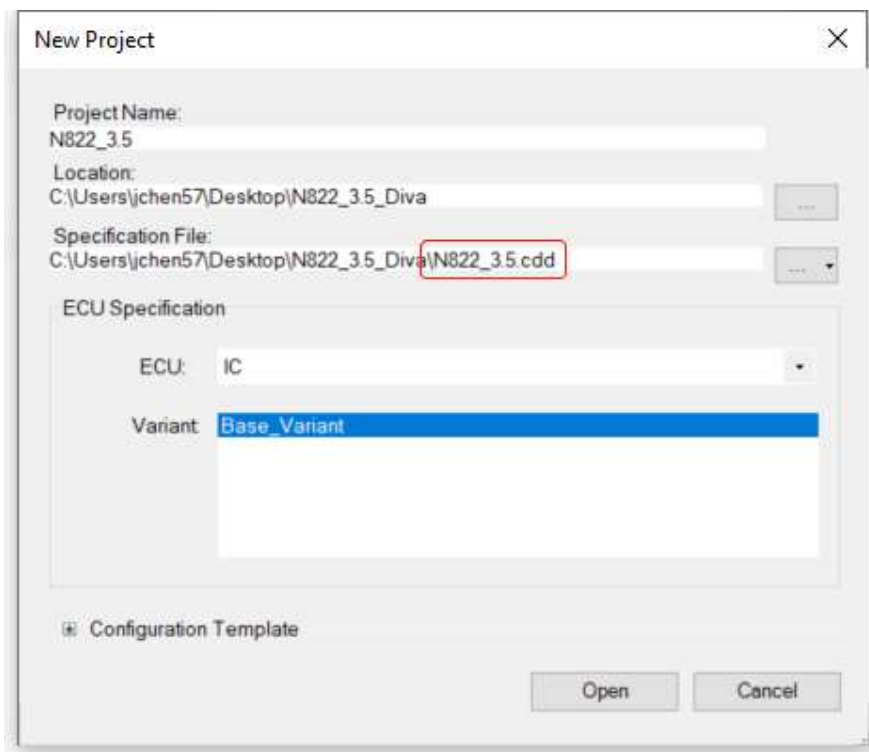
ISO15031 OBD(J1979)

GMW3110(v1.5, v1.6)

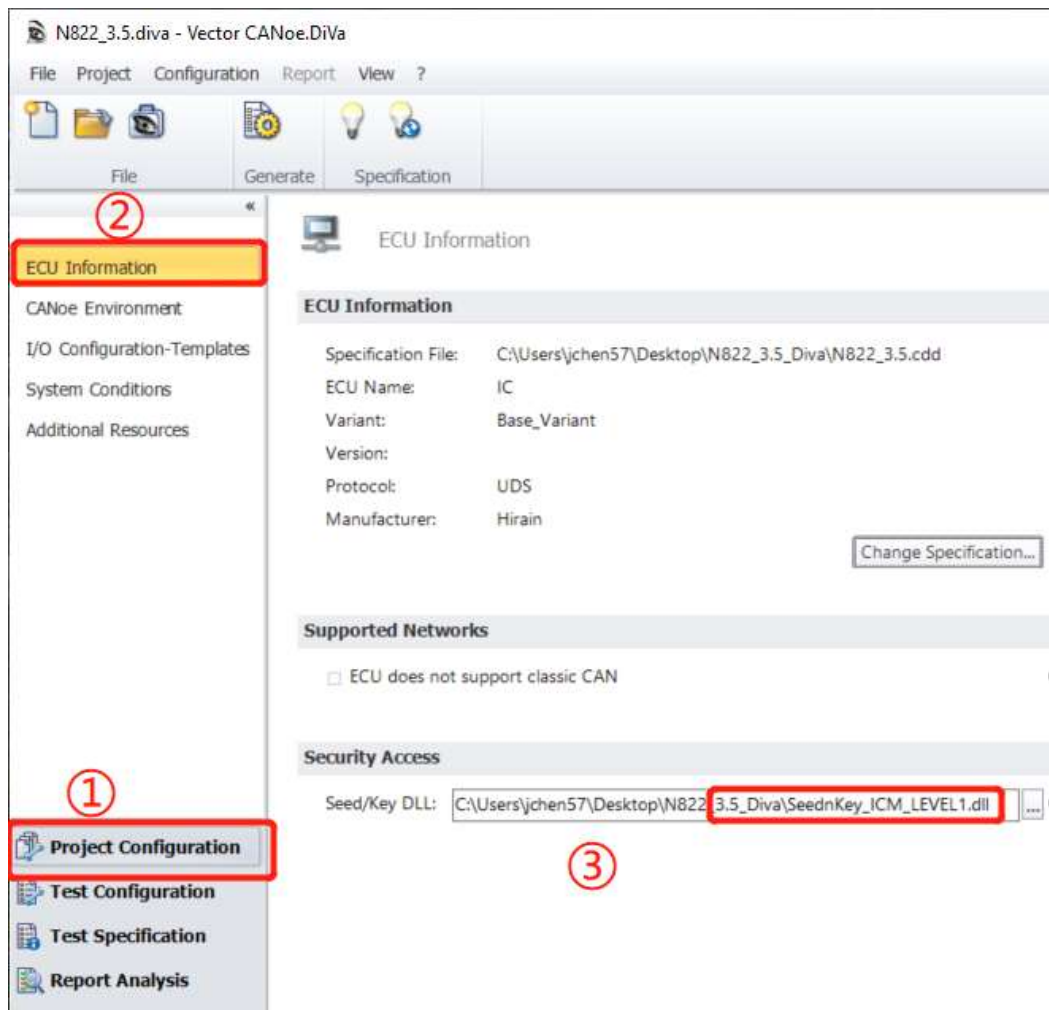
各个 OEM 具体的测试

## New Diva Project

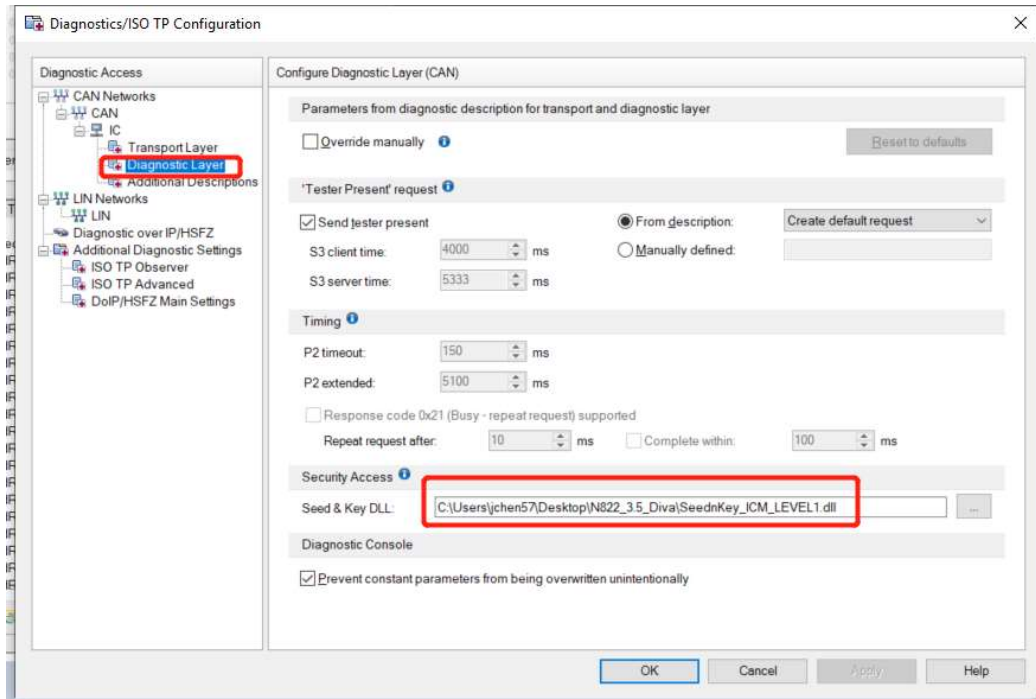
打开 Diva, 新建 Diva 工程(依赖于 Dela 的 cdd 文件)



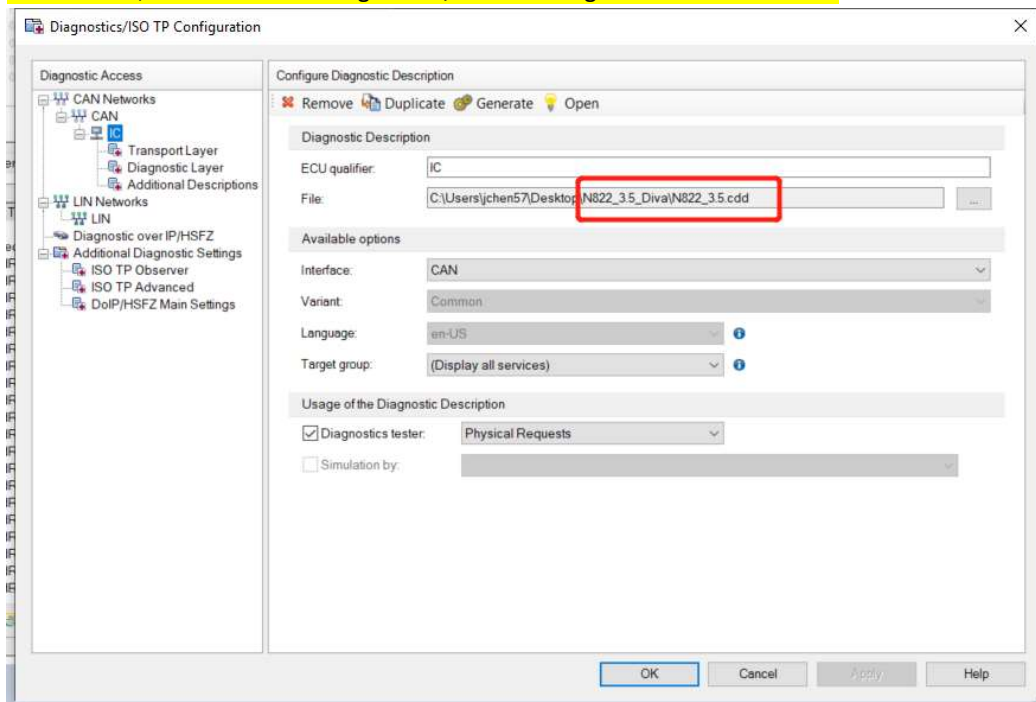
## Security Access



CANoe 导入 Diva 工程后, Diagnostics/ISO TP Configuration 中 Seed & Key DLL 自动选用配置的 dll.



不使用 Diva, 也能直接通过 Diagnostics/ISO TP Configuration 导入 cdd 文件.



## Automated Security Access

Select Seed/Key DLL for Key calculation:

To automatically unlock a SUT a Seed/Key DLL is required. Go to *Project Configuration/ECU Information* to choose a Seed/Key DLL.

If you already have a suitable DLL (e.g. for CANape or CANoe), you can reuse it here. Otherwise you may create your own DLL by following these steps:

1. Switch to the CANoe.DiVa installation directory (e.g. "C:\Program Files (x86)\Vector CANoe.DiVa 3.0") .
2. In the \Bin\SeedKey\GenerateKeyEx subdirectory you will find a file named **GenerateKeyEx.cpp**.
3. Open this file, e.g. with Microsoft Visual Studio or a comparable development environment.
4. Implement the algorithm that calculates the key.

The file contains the following C++ interfaces definition:

```
VKeyGenResultEx GenerateKeyEx(  
  
    const unsigned char* ipSeedArray, /* Array for the  
    seed [in] */  
  
    unsigned int iSeedArraySize, /* Length of the array  
    for the seed [in] */  
  
    const unsigned int iSecurityLevel, /* Security level  
    [in] */  
  
    const char* ipVariant, /* Name of the active variant  
    [in] */  
  
    unsigned char* iopKeyArray, /* Array for the key [in,  
    out] */  
  
    unsigned int iMaxKeyArraySize, /* Maximum length of  
    the array for the key [in] */  
  
    unsigned int& oActualKeyArraySize /* Length of the  
    key [out] */  
  
);
```

This allows you to use one Seed/Key dll for several security access levels: The subfunction may be the selector, e.g. the Seed/Key sequence 27 01 -> 27 02 has the security level 1 (sub function. is

1). The Seed/Key sequence 27 05 -> 27 06 has the security level 5 (sub function. of request seed is 5).

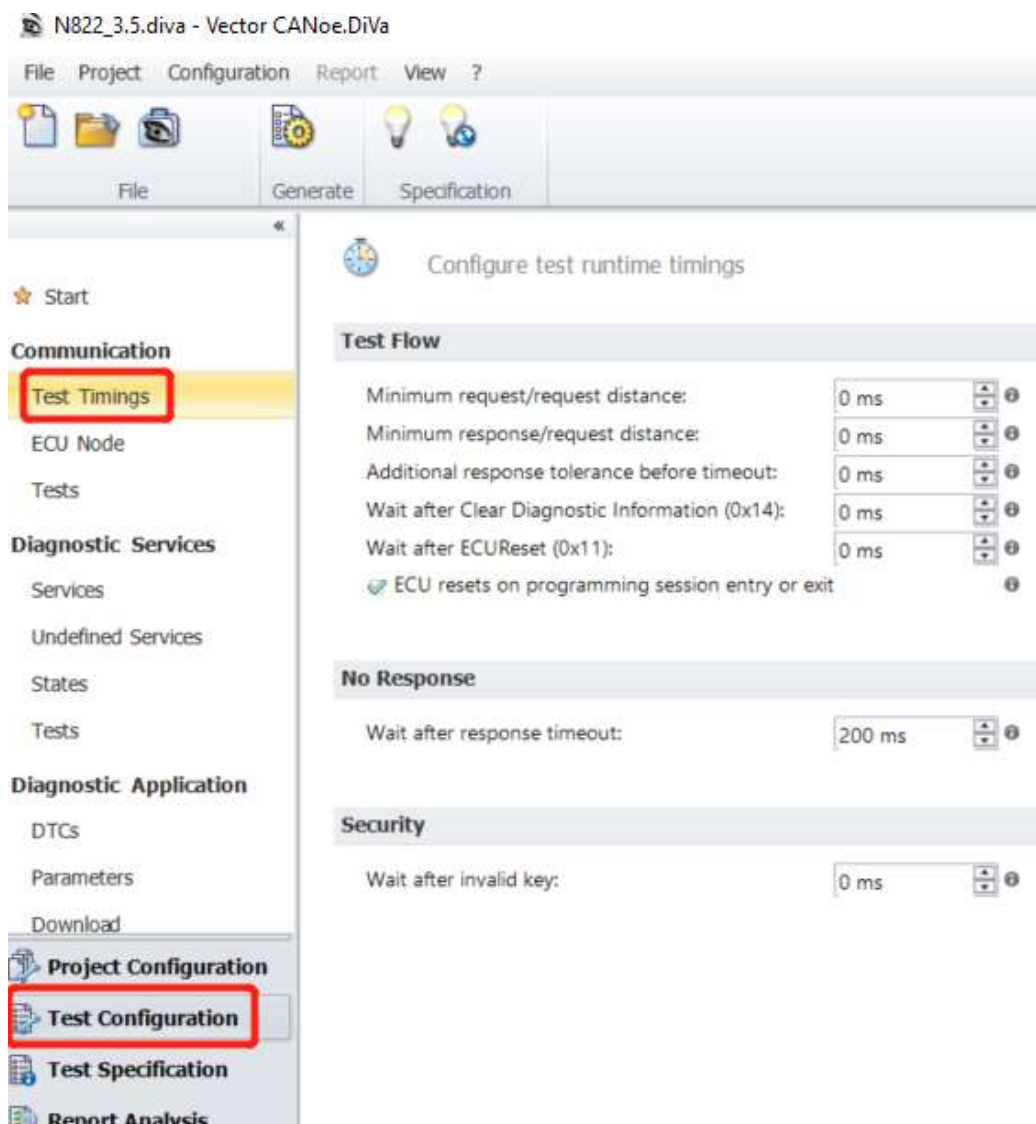
实际 Diva 中调用的是 CAPL 中的函数:

```
DiagGenerateKeyFromSeed(seedArray,  
                        seedArraySize,  
                        securityLevel,  
                        variant,  
                        option,  
                        keyArray,  
                        maxKeyArraySize,  
                        keyActualSize);
```

最终实际调用的是 dll 中的 **GenerateKeyEx.**

## Test Configuration

**Test Timings;** 在这里进行测试用例时间参数的设置, 如用例间的时间间隔、复位等待时间等 (建议设置为几十 ms)。



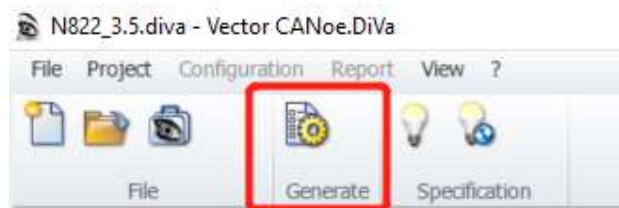
在 **Test** 选项页，选择是否测试功能寻址。

在 **Service** 选项页，勾选要进行测试的服务项。

其他配置选择可以保持默认（一般情况下，DTC 以及 31 服务、下载服务需要另外测试）。

## Generate

点击 **Generate**，即可生成测试用例(.can)

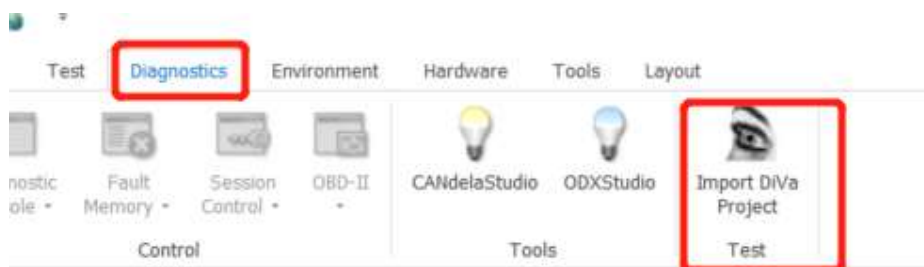


N822_3.5.conf	2021/2/2 10:49	CONF File	7 KB
N822_3.5.diva	2021/2/2 10:49	CANoe.DiVa 11.0 Project File	2 KB
N822_3.5.html	2021/2/2 10:49	Chrome HTML Document	689 KB
N822_3.5.001.can	2021/2/2 10:49	CAN File	879 KB
N822_3.5.callback.can	2021/2/2 10:49	CAN File	176 KB
N822_3.5.vxt	2021/2/2 10:49	VXT File	27 KB
N822_3.5.inst	2021/2/2 10:49	INST File	1,083 KB
N822_3.5.tmp	2021/2/2 10:49	TMP File	655 KB
N822_3.5.prim	2021/2/2 10:49	PRIM File	391 KB
N822_3.5_tmpReportFrames.html	2021/2/2 10:42	Chrome HTML Document	1 KB
N822_3.5.info	2021/2/2 10:33	INFO File	1 KB
N822_3.5.cdd	2020/12/1 16:36	CANdela Diagnostic Data	573 KB
IC_V348_7inch.dll	2020/7/23 17:46	Application extension	758 KB
SeednKey_ICM_LEVEL1.dll	2020/4/23 17:28	Application extension	295 KB

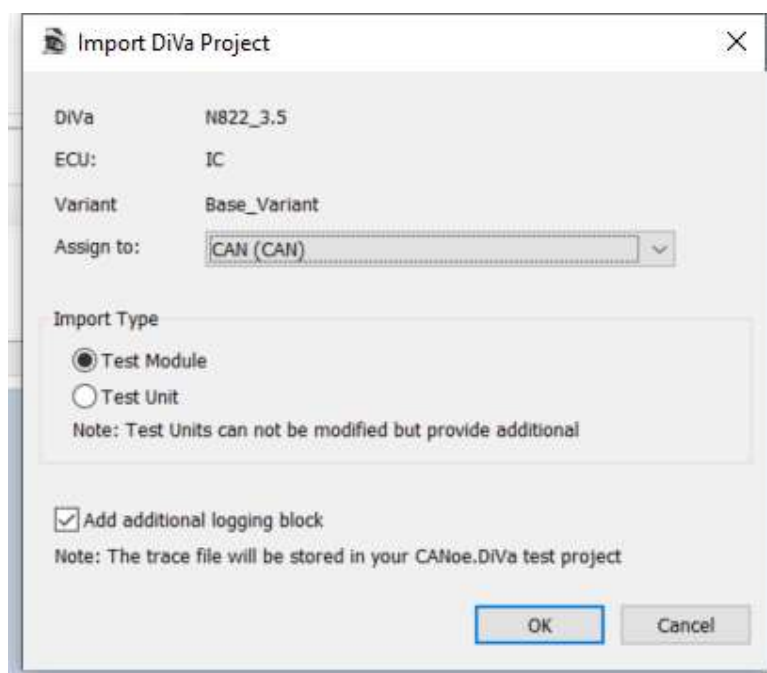
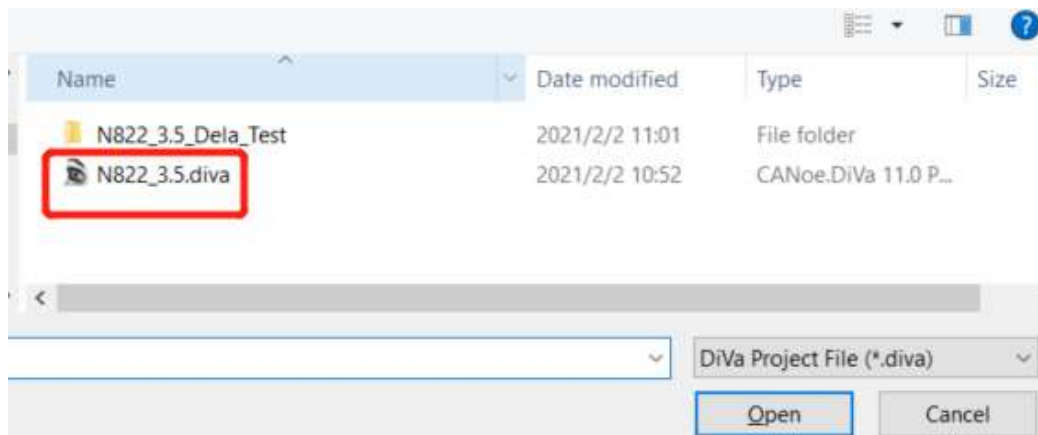
## CANoe Import Diva project

### 接上 VN1640A (xgu9 的)，运行 CANoe

打开 CANoe (version 11.0),将 Diva 工程导入到 CANoe 中.

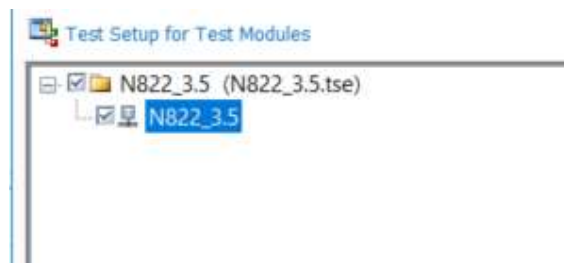




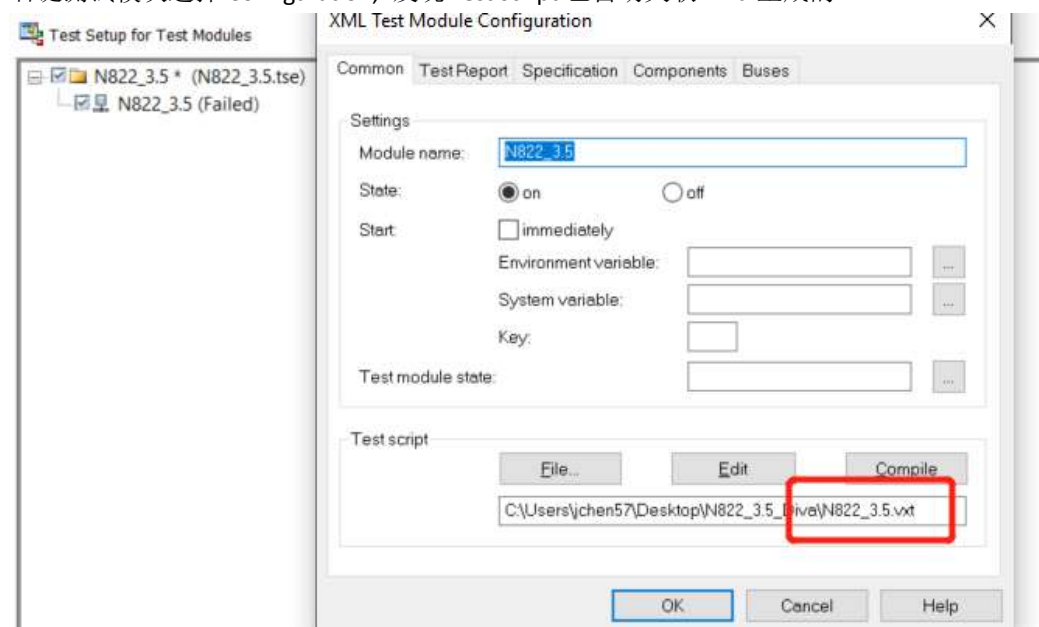


点击 **Test->Test Setup->双击 DiVa 工程名**，即可打开导入的测试工程。

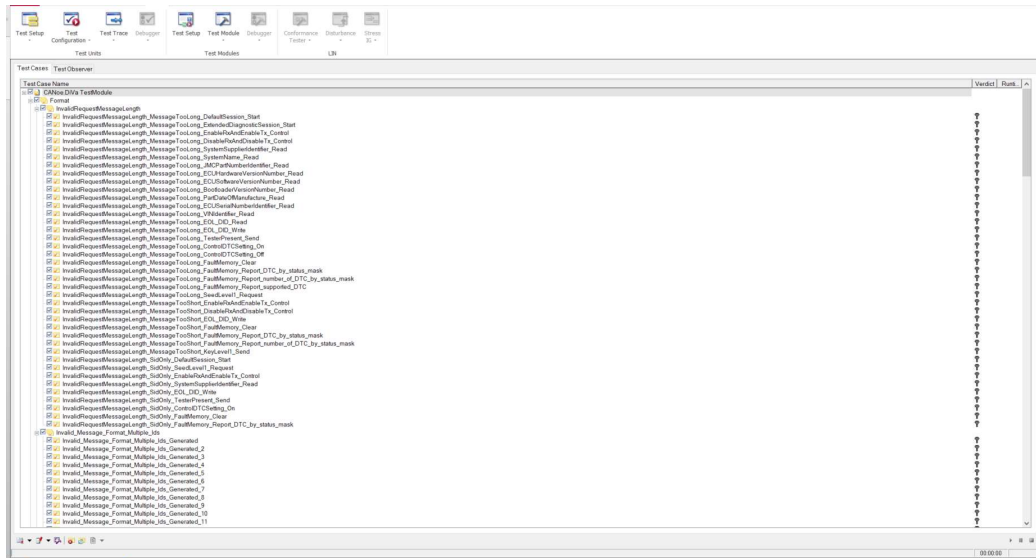
下图为自动生成的测试环境和测试模块



右键测试模块选择 Configuration, 发现 Test script 已自动关联 Diva 生成的.

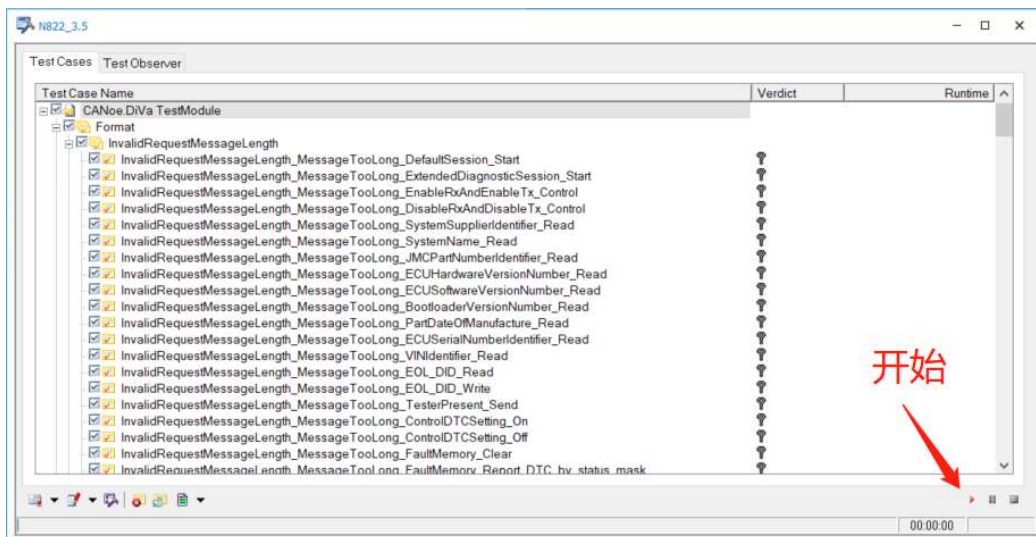


测试案例打开后的界面如下, 可以对要进行测试的案例进行勾选



接上 ECU,点击三角符号，运行测试用例。

开始测试



等待测试完成，约 22s.

测试结果如下：

NB22\_3.5

Test Cases Test Observer

Test Case Name	Verdict	Runtime
CAHoe DiVa TestModule		
Format		
InvalidRequestMessageLength	✗ (12)	3.049 s
InvalidRequestMessageLength_MessageTooLong_DefaultSession_Start	✓	0.851 s
InvalidRequestMessageLength_MessageTooLong_ExtendedDiagnosticSession_Start	✓	0.131 s
InvalidRequestMessageLength_MessageTooLong_EnableRxAndEnableTx_Control	✓	0.001 s
InvalidRequestMessageLength_MessageTooLong_DisableRxAndDisableTx_Control	✓	0.001 s
InvalidRequestMessageLength_MessageTooLong_SystemSupplierIdentifier_Read	✓	0.000 s
InvalidRequestMessageLength_MessageTooLong_SystemName_Read	✓	0.000 s
InvalidRequestMessageLength_MessageTooLong_JMCPartNumberIdentifier_Read	✓	0.001 s
InvalidRequestMessageLength_MessageTooLong_ECUHardwareVersionNumber_Read	✓	0.000 s
InvalidRequestMessageLength_MessageTooLong_ECUSoftwareVersionNumber_Read	✓	0.072 s
InvalidRequestMessageLength_MessageTooLong_BootloaderVersionNumber_Read	✓	0.041 s
InvalidRequestMessageLength_MessageTooLong_PartDateOfManufacture_Read	✓	0.001 s
InvalidRequestMessageLength_MessageTooLong_ECUSerialNumberIdentifier_Read	✓	0.000 s
InvalidRequestMessageLength_MessageTooLong_VINIdentifier_Read	✓	0.003 s
InvalidRequestMessageLength_MessageTooLong_EOL_DID_Read	✓	0.001 s
InvalidRequestMessageLength_MessageTooLong_EOL_DID_Write	✓	0.186 s
InvalidRequestMessageLength_MessageTooLong_TesterPresent_Send	✓	0.084 s
InvalidRequestMessageLength_MessageTooLong_ControlDTCSetting_On	✓	0.058 s
InvalidRequestMessageLength_MessageTooLong_ControlDTCSetting_Off	✓	0.000 s
InvalidRequestMessageLength_MessageTooLong_FaultMemory_Clear	✓	0.000 s
InvalidRequestMessageLength_MessageTooLong_FaultMemory_Report_DTC_by_status_mask	✓	0.000 s

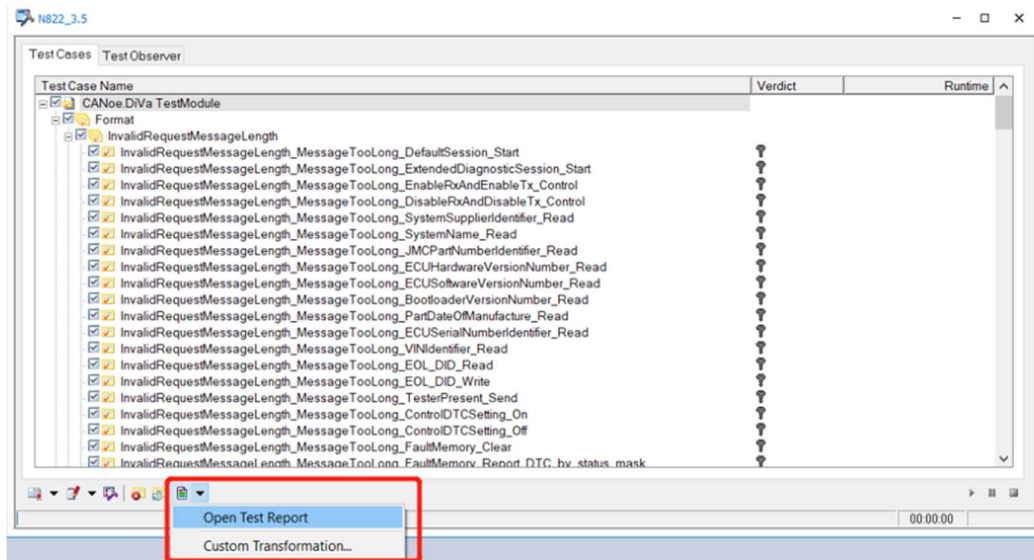
executed: 219 of 219

00:00:22 **Failed: 36**

测试 trace 如下:

Time	Chn	ID	Name	Event Type	Dir	D...	D...	Data
41.53...	CA...	386		CAN Frame	Rx	8	8	00 00 00 00 00 00 00 00
41.53...	CA...	392		CAN Frame	Rx	8	8	FE 00 00 FF 27 10 00 00
41.21...	CA...	430		CAN Frame	Rx	8	8	30 04 00 00 00 00 00 00
41.53...	CA...	510		CAN Frame	Rx	8	8	00 00 00 00 00 00 0C 0C
41.53...	CA...	525		CAN Frame	Rx	8	8	FF 8F FF 00 FF F8 0C 87
41.13...	CA...	581		CAN Frame	Rx	8	8	00 00 00 0C 04 05 00 00
28.49...	CA...	730	<OTP>	SF	Tx	2	8	[02] 02 80 [CC CC CC CC CC]
28.04...	CA...	730	Default Session St...	req		2		10 01
28.49...	CA...	738	<OTP>	SF	Rx	3	8	[03] 7F 02 11 [CC CC CC CC]
28.04...	CA...	738	Default Session St...	pos		6		80 11 00 32 01 F4
13.73...	CA...	738	Default Session St...	neg		3		7F 10 22
28.05...	CA...	730	Extended Diagnost...	req		2		10 03
13.76...	CA...	738	Extended Diagnost...	neg		3		7F 10 22
28.05...	CA...	738	Extended Diagnost...	pos		6		50 11 00 32 01 F4
27.62...	CA...	730	EnableRxAndEnabl...	req		3		28 00 01
11.76...	CA...	738	EnableRxAndEnabl...	neg		3		7F 28 22
27.61...	CA...	730	DisableRxAndDisa...	req		3		28 03 01
11.77...	CA...	738	DisableRxAndDisa...	neg		3		7F 28 22
24.55...	CA...	730	SystemSupplierId...	req		3		22 F1 BA
12.56...	CA...	738	SystemSupplierId...	neg		3		7F 22 22
24.62...	CA...	730	SystemName Rea...	req		3		22 F1 97
7.745...	CA...	738	SystemName Rea...	neg		3		7F 22 13
24.76...	CA...	730	JMCPartNumberId...	req		3		22 F1 13
7.925...	CA...	738	JMCPartNumberId...	neg		3		7F 22 13
24.99...	CA...	730	ECUHardwareVers...	req		3		22 F1 11
6.415...	CA...	738	ECUHardwareVers...	neg		3		7F 22 13
25.27...	CA...	730	ECUSoftwareVers...	req		3		22 F1 88
6.425...	CA...	738	ECUSoftwareVers...	neg		3		7F 22 13
25.55...	CA...	730	BootloaderVersion...	req		3		22 F1 80
6.435...	CA...	738	BootloaderVersion...	neg		3		7F 22 13
25.70...	CA...	730	PartDateOfManufa...	req		3		22 F0 10
6.445...	CA...	738	PartDateOfManufa...	neg		3		7F 22 13
25.90...	CA...	730	ECUSerialNumberL...	req		3		22 F1 8C
6.456...	CA...	738	ECUSerialNumberL...	neg		3		7F 22 13
26.09...	CA...	730	VINIdentifier Read...	req		3		22 F1 90
6.465...	CA...	738	VINIdentifier Read...	neg		3		7F 22 13
27.47...	CA...	730	EOL DID Read::req	req		3		22 DF 00
6.475...	CA...	738	EOL DID Read::req	neg		3		7F 22 13
28.21...	CA...	730	RequestSeed Leve...	req		2		27 01
28.21...	CA...	738	RequestSeed Leve...	pos		6		67 01 86 18 68 49
28.22...	CA...	730	SendKey Level 1 S...	req		6		27 02 D8 49 18 A4
28.22...	CA...	738	SendKey Level 1 S...	pos		2		67 02
27.60...	CA...	730	EOL DID Write::req	req		23		2E DF 00 00 00 00 00 00 02 00 00 ...
11.84...	CA...	738	EOL DID Write::req	neg		3		7F 2E 22
26.45...	CA...	730	Tester Present Se...	req		2		3E 00
9.295...	CA...	738	Tester Present Se...	neg		3		7F 3E 13
26.47...	CA...	730	Control DTC Settl...	req		2		85 01
11.85...	CA...	738	Control DTC Settl...	neg		3		7F 85 22
26.49...	CA...	730	Control DTC Settl...	req		2		85 02
11.86...	CA...	738	Control DTC Settl...	neg		3		7F 85 22
27.45...	CA...	730	Fault Memory Cle...	req		4		14 FF FF FF
9.584...	CA...	738	Fault Memory Cle...	neg		3		7F 14 11
26.67...	CA...	730	Fault Memory Rep...	req		3		19 02 FF
6.995...	CA...	738	Fault Memory Rep...	neg		3		7F 19 13
26.81...	CA...	730	Fault Memory Rep...	req		3		19 01 FF
7.005...	CA...	738	Fault Memory Rep...	neg		3		7F 19 13
27.21...	CA...	730	Fault Memory Rep...	req		2		19 0A
6.875...	CA...	738	Fault Memory Rep...	neg		3		7F 19 13
13.48...	CA...	738	PowerSoftGuard 1...	neg		3		7C 17 17

点击下图中的"open Test Report", 打开生成的测试报告。



CANoe DiVa TestModule

- General Test Information
- DiVa Configuration Information
- Test Overview
- Preparation of Test Module
  - 1 Format
  - 2 DataContent
  - 3 Application
  - 4 Session
  - 5 General
  - 6 FunctionalAddressing
  - 7 Transport\_Layer
- Completion of Test Module

### Test Overview

Test begin: 2021-02-02 15:10:25 (logging timestamp 8.669486)  
Test end: 2021-02-02 15:10:47 (logging timestamp 30.239319)

#### Statistics

Overall number of test cases	219	
Executed test cases	194	89% of all test cases
Not executed test cases	25	11% of all test cases
Test cases passed	155	80% of executed test cases
Test cases with warning	16	8% of executed test cases
Test cases failed	23	12% of executed test cases

#### Test Case Results

Preparation of Test Module	
1	Format
1.1	InvalidRequestMessageLength

#### Failed Testcases:

- 2.1.1 InvalidRequestData\_EOL\_DID\_Write
- 3.6.1 Invalid\_data\_request\_WDBI\_EOL\_DID\_Write
- 4.1.1 Valid\_Security\_Access\_KeyLevel1\_Send
- 4.3.1 RequestSequenceError\_KeyLevel1\_Send
- 4.7.1 SimpleStateChange\_DefaultSession\_Start
- 4.7.2 SimpleStateChange\_ExtendedDiagnosticSession\_S
- 4.7.3 InvalidStateChange\_DefaultSession\_Start
- 4.7.4 InvalidStateChange\_DefaultSession\_Start\_2
- 4.7.5 InvalidStateChange\_ProgrammingSession\_Start

12.756253	Request Seed	Positive response received as expected.	pass															
12.756253	Check Seed	The received seed should not be 0 anymore.	pass															
12.759253	Verify new state	Make sure that the ECU is not in a secured state any more. (ProgrammingSession_Start)	-															
12.826765	Verify new state	Expected a negative response - last received response was positive(Error(s) occurred)	fail															
ProgrammingSession_Start																		
<table><tr><th>Parameter</th><th>Value</th><th>Raw</th></tr><tr><td>SID-PR</td><td>0x50</td><td>50</td></tr><tr><td>Type</td><td>0x02</td><td>02</td></tr><tr><td>P2</td><td>0x32 ms</td><td>00 32</td></tr><tr><td>P2Ex</td><td>5000 ms</td><td>01 14</td></tr></table>				Parameter	Value	Raw	SID-PR	0x50	50	Type	0x02	02	P2	0x32 ms	00 32	P2Ex	5000 ms	01 14
Parameter	Value	Raw																
SID-PR	0x50	50																
Type	0x02	02																
P2	0x32 ms	00 32																
P2Ex	5000 ms	01 14																
Expected a negative response (NRC 0x33) - last received response was positive.																		
12.829765	ResetSACounter(Seed)	Ensure SecurityAccess Attempt Counter=0 (SeedLevel1_Request)	-															
12.833403	ResetSACounter(Seed)	Expected a positive response - last received response was negative(Error(s) occurred)	fail															
12.835403	ResetSACounter(Key)	Ensure SecurityAccess Attempt Counter=0 (KeyLevel1_Send)	-															

## 根据测试报告进行错误处理

(1)、点击前面步骤生成的测试报告中"fail"的案例；分析失败原因。**在测试报告已经对其发送的测试内容；预期的响应及 ECU 实际返回的情况进行了描述。**

(2)、根据测试报告分析失败原因；**如果是测试用例不符合规范、则修改 cdd 文件，重新生成 Diva 工程进行测试；如果是 ECU 诊断不符合诊断规范，则修改 ECU 软件，直到测试通过率达到要求。**

## Dll

### CANoe CAPL dll

用 CAPL 专有的固定格式声明实现相关函数.

必须严格按照 Vector 的要求来创建,否则 CANoe 无法识别.

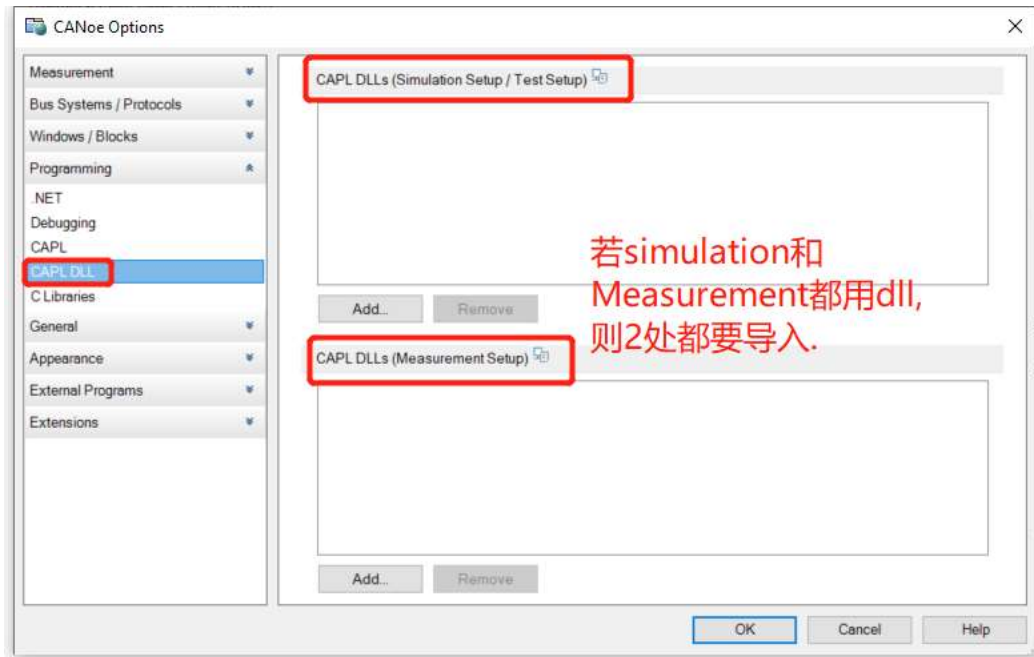
所有已创建的函数必须使用函数列表 CAPL\_DLL\_INFO\_LIST 才能导入 CAPL 代码中.

为了在 CANoe 工程中调用 CAPL DLL, 需导入:

方法 1:

File - Options – Programming – CAPL DLL



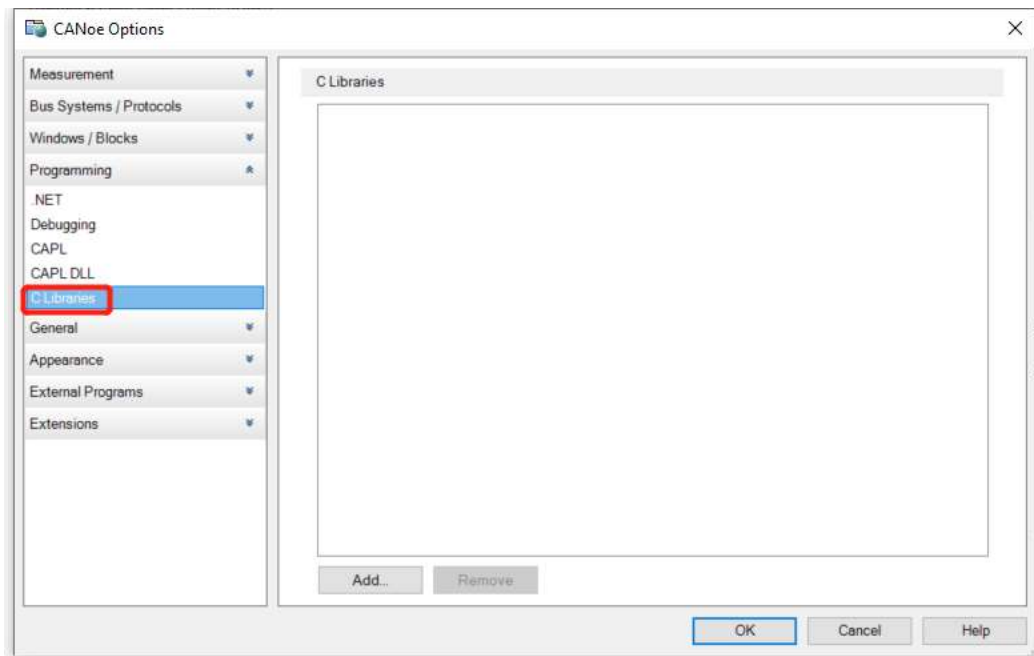


方法 2:

在 CAPL 程序中#pragma library (“dll file path”)

此种方法只能在当前 CAPL 中使用该 dll.

## C Library



CCL 库, CANoe C Library, 可以使用 C/C++/C# 开发 dll.



区别于 CAPL dll(借助 Visual Studio 实现一些 CAPL 不易实现的功能), CCL 是在 Visual Studio 中调用 CAPL 函数(详见相关 API).

能类似 CAPL 读取 CAN 数据,提供 timer,访问工程中的变量等,也能实现一些普通 windows dll 功能.

## Seed & Key Dll

```
PS C:\Users\jchen57\Desktop\N822_3.5_Diva> findstr /s /i /m /c:"DiagGenerateKeyFromSeed" *.*
N822_3.5.001.can
N822_3.5.001.cbf
N822_3.5.callback.can
N822_3.5.callback.cbf
PS C:\Users\jchen57\Desktop\N822_3.5_Diva> findstr /s /i /m /c:"GenerateKeyEx" *.*
IC_V348_7inch.dll
SeednKey_ICM_LEVEL1.dll
PS C:\Users\jchen57\Desktop\N822_3.5_Diva> _
```

诊断专用, 必须实现函数:

**GenerateKeyEx** (**DiagGenerateKeyFromSeed** 是 CAPL API, 最终调用的是 **GenerateKeyEx**)

执行 **DiagGenerateKeyFromSeed** 可能比较复杂,故会 stop the execution of the real-time measurement process, possibly causing problems like event loss and timeouts.

因此建议按照 Vector 提供的执行序列调用 **DiagGenerateKeyFromSeed** 函数.

参考 F1 search Seed & Key DLL / Security Access .

