

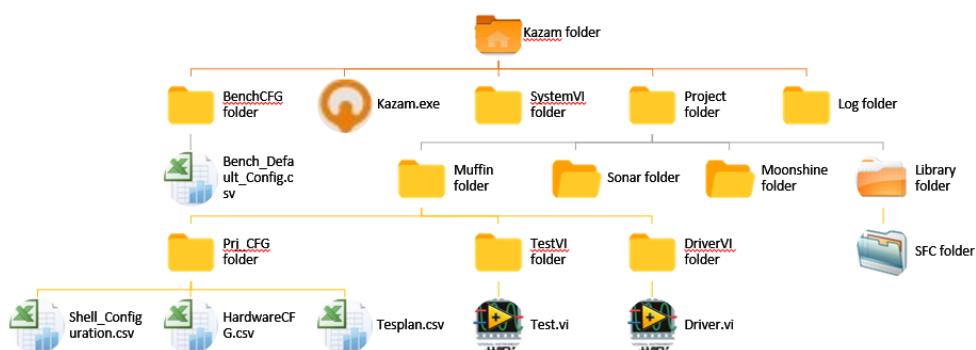
Study Report

一、KAZAM 的介绍

KAZAM 是 Amazon 开发的用于测试的一个工具。在使用 KAZAM 进行测试的时候，会有一个配置文件，接下来介绍 KAZAM 的项目文件

1、KAZAM 的文件夹

如下所示：



不同的文件夹，存放不同的配置，接下来进行介绍，以下图中的文件（OBA 工站）进行介绍，不同的工作站，项目文件也会有略微的不同

Study_Report > report_KANZAM			
名稱	修改日期	類型	大小
ContentFile	2025/11/7 下午 0...	檔案資料夾	
KC20_Pre_Seal_OBA_PVT_Test_V3.0.0.20_VN	2025/11/7 下午 0...	檔案資料夾	

双击打开进入到以下目录

Study_Report > report_KANZAM > KC20_Pre_Seal_OBA_PVT_Test_V3.0.0.20_VN			
名稱	修改日期	類型	大小
adb	2025/11/7 下午 0...	檔案資料夾	
BenchCFG	2025/11/7 下午 0...	檔案資料夾	
Log	2025/11/7 下午 0...	檔案資料夾	
Picture	2025/11/7 下午 0...	檔案資料夾	
Project	2025/11/7 下午 0...	檔案資料夾	
Sound	2025/11/7 下午 0...	檔案資料夾	
SystemVI	2025/11/7 下午 0...	檔案資料夾	
Kazam.alises	2022/5/28 下午 0...	ALIASES 檔案	1 KB
Kazam.exe	2022/5/28 下午 0...	應用程式	8,448 KB
Kazam.ini	2022/5/28 下午 0...	組態設定	1 KB
messageShow.ini	2025/9/11 下午 0...	組態設定	1 KB
RelConfig	2023/3/10 下午 1...	檔案	1 KB
SOFTWARE LICENSE AGREEMENT.txt	2022/12/12 上午 ...	文字文件	18 KB

从第一个文件夹开始进行介绍：

1. adb 文件夹

进入到 adb 文件，会有 3 个文件，这三个文件是干嘛用的？

adb 的指令,有时候 Sequence 文件会使用到

Study_Report > report_KANZAM > KC20_Pre_Seal_OBA_PVT_Test_V3.0.0.20_VN > adb			
名稱	修改日期	類型	大小
adb.exe	2021/8/4 上午 10...	應用程式	1,807 KB
AdbWinApi.dll	2021/8/4 上午 10...	應用程式擴充	96 KB
AdbWinUsbApi.dll	2021/8/4 上午 10...	應用程式擴充	62 KB

2. BenchCFG 文件夹

Study_Report > report_KANZAM > KC20_Pre_Seal_OBA_PVT_Test_V3.0.0.20_VN > BenchCFG

名稱	修改日期	類型	大小
Bench_CommandList.csv	2021/11/15 下午 ...	Microsoft Excel ...	1 KB
Bench_Default_Config.csv	2025/10/7 下午 0...	Microsoft Excel ...	1 KB
Bench_Default_Config_Server.csv	2022/5/28 下午 0...	Microsoft Excel ...	2 KB

① Bench_CommandList.csv

CheckValue	493728b8aa392e879842bb37ba65db1d
Command	Vi Name
VI_PATH	GUI_Operator.vi\$shell_LoadBenchCFG.villog_amazon_aws.villog_csv.vilsys_SequenceCFGReadout.vilsys_SequenceLauncher.vi
CFG_PATH	shell_LoadBenchCFG.vilsys_SequenceCFGReadout.vi
FP_Show	GUI_Operator.vi
Seq_PATHISLOTS	sys_SequenceLauncher.vi
LOG_PATHLOG	log_amazon_aws.villog_csv.vi

分别表示什么意思？

② Bench_Default_Config.csv (重点)

Check Value	a1dclae43978681a83134bfef44ac2ad
Item	Detail
BenchShellPath	Product\Prj_CFG\shell_factory.csv
BenchHardwareConfigurationPath	Product\Prj_CFG\HardwareCFG.csv
BenchTestPlanPath	Product\Prj_CFG\
SearchPath	Project\
Project Line	1
Test Station Name	Pre-Seal-OBA-Test
Fixture Name	1
Product Name	Hypnos
Factory_ID	Ensky
MinSizeOfShowResult	10
TCPPort	7200
AutoUpdate	
Emailcontact	andy.w.lee@enskytech.com
Textcontact	phonenumber1
LOG_PATH	D:\Lab126Test\KC20\FATP\Pre-Seal-OBA-Test\;
KazamUpdateServer	
ProjectUpdateServer	
Fixture_NO	Pre-Seal-OBA-Test-1
Fixture_ID	Pre-Seal-OBA-Test-1
ProjectUpdatePackageName	Pre-Seal-OBA-Test

CheckValue: 用于值的检查

Item:

BenchShellPath: Shell_factory.csv(shell 配置)的路径

BenchHardwareConfigurationPath: HardwareCFG.csv (硬件配置) 的路径

BenchTestPlanPath:

SearchPath: 搜索的路径, 运行开发人员进入, 当有多个路径时, 用分号进行分割,

工作过程: 第一步, 在上图中的情况下 kazam 首先搜索 kazam.exe 路径下的 “Project\” 路径, 如果能找到具体的测试计划(Sequence),kazam 就会停止搜索, 否则会进入到 “Project\” 继续搜索, 直到查找到 sequence 文件为止。

第二步: kazam 将从第一个搜索路径开始搜索驱动程序并测试 vi。

Project Line: 用于定义项目的名称。文件名登录。---? 不理解

Test Station Name: 工站的名称

Fixture Name: 治具的名称

Product Name: ---? 不懂

Factory_ID: 工厂的 ID，会写入 Log 中

MinSizeOfShowResult: 用于定义显示结果的长度。如图所示，有 10 个结果需要在 GUI 上显示。所以结果的长度是 10。

如果开发人员想要控制每个 show 结果的“下一步”，那么 MinSizeOfShowResult 需要设置为大于 11，例如 12。

如果开发人员不想控制每个 show 结果的“下一步”，那么 MinSizeOfShowResult 需要设置为小于 11，例如 10。

TCPPort: 设置 TCP 的端口为 7200 (默认端口)

AutoUpdate: 用于设置是否自动更新项目文件夹。它的价值在于启用或禁用。

Emailcontact: 用于设置联系人的邮箱。这个内容会被写入 Log

Textcontact: 用于设置联系人的电话号码。这个内容会被写入 Log

LOG_PATH: 设置日志的存储路径，可以设置多个路径，并用分号对每个路径进行分割。

KazamUpdateServer: 用于定义 Kazam 在 exe 中运行时的 kazam 服务器路径模式，它会自动检查更新服务器中是否有更新。

ProjectUpdateServer:

Fixture_No:

Fixture_Id:

ProjectUpdatePackageName:

③

ProjID	BenchName	SearchPath	Project	Lir	Test	Statio	Fixture	Na	Product	Nr	Factory	IL	MinSizeOfTCPPort	AutoUpdat	Emailcont	Textcont	LOG_PATH	KazamUpd	ProjectUp	ProjectUpc	FileUpload_Path	GUIData	SOP_PATH
DemoPj1	DemoPj1	ProjectDemo		1	BFT	xxx	Demo	LAB	10	7200	user1@ser phonenum	\.\.\.\LOG\Kazam					\..\..\LOG\N\192.168.0.1\TestLog\				2020	2021	
DemoPj2	DemoPj2	ProjectDemo		1	APT	xxx	Demo	LAB	10	7200	user1@ser phonenum	\.\.\.\LOG\Kazam					\..\..\LOG\N\192.168.0.1\TestLog\				2020	2021	
DemoPj3	DemoPj3	ProjectDemo		1	AUT	xxx	Demo	LAB	10	7200	user1@ser phonenum	\.\.\.\LOG\Kazam					\..\..\LOG\N\192.168.0.1\TestLog\				2020	2021	

3. Log

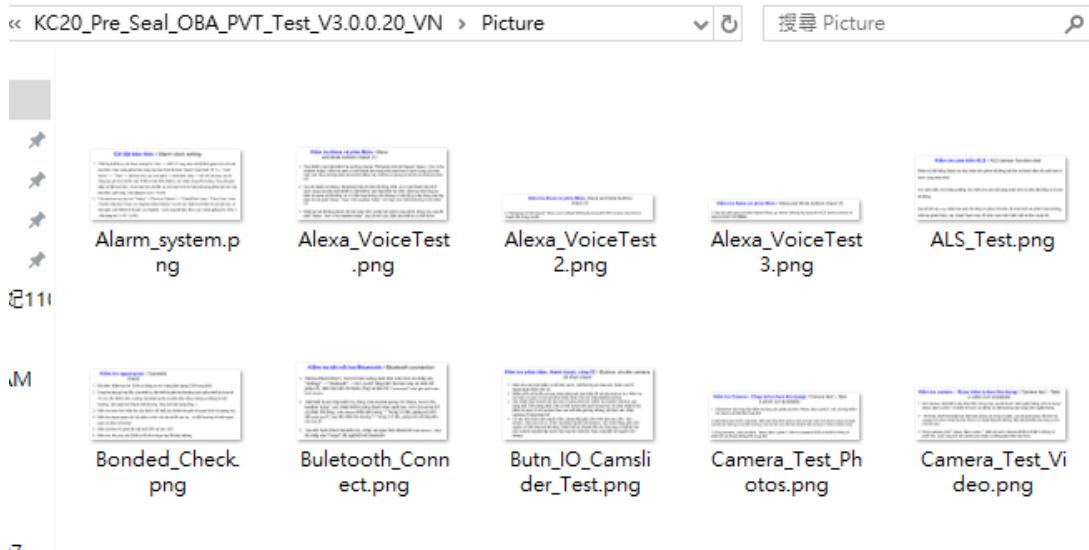
疑惑: Log 如何生成, 生成的内容当中会有哪些信息?

当执行 Kazam.exe 的时候自动生成的, 里面存放的信息就是 error 的时候的信息?

Study_Report > report_KANZAM > KC20_Pre_Seal_OBA_PVT_Test_V3.0.0.20_VN > Log

名稱	修改日期	類型	大小
SYS_Error_Log_2022-12-09.txt	2022/12/9 上午 0...	文字文件	1 KB
SYS_Error_Log_2022-12-12.txt	2022/12/12 下午 ...	文字文件	4 KB
SYS_Error_Log_2022-12-14.txt	2022/12/14 上午 ...	文字文件	4 KB
SYS_Error_Log_2023-01-11.txt	2023/1/11 上午 0...	文字文件	3 KB
SYS_Error_Log_2023-03-10.txt	2023/3/10 下午 1...	文字文件	4 KB
SYS_Error_Log_2023-06-06.txt	2023/6/6 下午 05...	文字文件	2 KB
SYS_Error_Log_2023-09-11.txt	2023/9/11 上午 1...	文字文件	2 KB
SYS_Error_Log_2024-01-02.txt	2024/1/2 下午 02...	文字文件	1 KB
SYS_Error_Log_2024-01-03.txt	2024/1/3 上午 09...	文字文件	1 KB
SYS_Error_Log_2024-10-28.txt	2024/10/28 上午 ...	文字文件	2 KB
SYS_Error_Log_2025-09-11.txt	2025/9/11 下午 0...	文字文件	4 KB
SYS_Error_Log_2025-09-16.txt	2025/9/16 下午 0...	文字文件	4 KB
SYS_Error_Log_2025-09-25.txt	2025/9/25 下午 0...	文字文件	12 KB

4. Picture



5. Project

Project 文件夹里面有这些 sub 文件夹

名稱	修改日期	類型	大小
Cosmetic2KazamTool	2025/11/7 下午 0...	檔案資料夾	
EnskyLib	2025/11/7 下午 0...	檔案資料夾	
LatestTesterVI	2025/11/7 下午 0...	檔案資料夾	
Library	2025/11/7 下午 0...	檔案資料夾	
Product	2025/11/7 下午 0...	檔案資料夾	
SFC	2025/11/7 下午 0...	檔案資料夾	

① Cosmetic2KazamTool 文件夾

② EnskyLib 文件夾

③ LatestTesterVI

④ Library

⑤ Product

进入到最里面的一层路径下，有这四个文件

<< report_KANZAM > KC20_Pre_Seal_OBA_PVT_Test_V3.0.0.20_VN > Project > Product > Prj_CFG

名稱	修改日期	類型	大小
HardwareCFG.csv	2021/8/4 上午 10...	Microsoft Excel ...	2 KB
Kazam_Check.xlsx	2023/3/10 下午 1...	Microsoft Excel ...	13 KB
KC20_Pre_Seal_OBA_PVT_Test_V3.0.0.2...	2025/10/7 上午 0...	Microsoft Excel ...	16 KB
shell_factory.csv	2025/10/7 下午 0...	Microsoft Excel ...	1 KB
SUB_COSMETIC.csv	2022/12/12 上午 ...	Microsoft Excel ...	37 KB
subREL_Setting.csv	2022/12/12 上午 ...	Microsoft Excel ...	2 KB

第一个：HardwareCFG.csv：这个文件是硬件配置的文件，不可以进行修改

第二个：Kazam_Check.xlsx：

第三个：KC20_Pre_Seal_OBA_PVT_Test_V3.0.0.20_VN.xlsx：常说的 Sequence

文件，里面有测试的序列，以及详细的信息。下面有 sequence 文件内容当中的字段的详细介绍。

第四个：shell_factory.csv：它是 Kazam.exe 的配置文件，开发人员需要根据自己的需要进行编辑。

第五个：SUB_COSMETIC.csv：

第六个：subREL_Setting.csv:

通常会把：Kazam_Check.xlsx 、SUB_COSMETIC.csv、subREL_Setting.csv 这个三个表放在 sequence 当中，作为副表存在。如下图：

⑥ SFC

它包含所有 SFC vi，开发人员必须将所有 SFC vi 放入此文件夹

<< Study_Report >> report_KANZAM >> KC20_Pre_Seal_OBA_PVT_Test_V3.0.0.20_VN >> Project >> SFC >> Ensky			
	名稱	修改日期	類型
1	ChangeStepPassFailBy_Input.vi	2022/6/7 上午 09...	LabVIEW Instrum...
2	ChangeStepResultBy_Input.vi	2022/6/7 上午 09...	LabVIEW Instrum...
3	CheckKeywordAndReplace.vi	2022/6/7 上午 09...	LabVIEW Instrum...
4	DebugLogModule.dll	2020/3/23 下午 0...	應用程式擴充
5	DeviceWatcherSetting_Demo.ini	2020/3/23 下午 0...	組態設定
6	EnskySFC_Optional.ini	2020/3/23 下午 0...	組態設定
7	KeepFirstValue.vi	2022/6/7 上午 09...	LabVIEW Instrum...
8	log_DeviceWatcher.vi	2022/6/7 上午 09...	LabVIEW Instrum...
9	log_EnskySFC_SelfLog.vi	2022/6/7 上午 09...	LabVIEW Instrum...
10	log_ErrorHandler.vi	2022/6/7 上午 09...	LabVIEW Instrum...

6. SystemVi

里面包含了一些配置，什么配置？→不清楚

Study_Report > report_KANZAM > KC20_Pre_Seal_OBA_PVT_Test_V3.0.0.20_VN > SystemVI

名稱	修改日期	類型	大小
sys_Configuration	2025/11/7 下午 0...	檔案資料夾	
sys_Control	2025/11/7 下午 0...	檔案資料夾	
sys_EditorDebugger	2025/11/7 下午 0...	檔案資料夾	
sys_Global	2025/11/7 下午 0...	檔案資料夾	
sys_GUISourceCode	2025/11/7 下午 0...	檔案資料夾	
sys_SequenceProducer	2025/11/7 下午 0...	檔案資料夾	
sys_ShellSourceCode	2025/11/7 下午 0...	檔案資料夾	
sys_SourceCode	2025/11/7 下午 0...	檔案資料夾	
sys_Tools	2025/11/7 下午 0...	檔案資料夾	

7. System

双击进入后会有以下两个文件夹

本機 > 本地磁盤 (F:) > 6.57 > KC27_Cosmetic2_V4.0.0.3 > System >												
<table border="1"><thead><tr><th>名稱</th><th>修改日期</th><th>類型</th><th>大小</th></tr></thead><tbody><tr><td>sys_bin</td><td>2025/11/6 下午 0...</td><td>檔案資料夾</td><td></td></tr><tr><td>sys_Configuration</td><td>2025/11/6 下午 0...</td><td>檔案資料夾</td><td></td></tr></tbody></table>	名稱	修改日期	類型	大小	sys_bin	2025/11/6 下午 0...	檔案資料夾		sys_Configuration	2025/11/6 下午 0...	檔案資料夾	
名稱	修改日期	類型	大小									
sys_bin	2025/11/6 下午 0...	檔案資料夾										
sys_Configuration	2025/11/6 下午 0...	檔案資料夾										

sys_bin:截图未截完,

本機 > 本地磁盤 (F:) > 6.57 > KC27_Cosmetic1_V4.0.0.3 > System > sys_bin >																																																																																																																								
<table border="1"><thead><tr><th>名稱</th><th>修改日期</th><th>類型</th><th>大小</th></tr></thead><tbody><tr><td>AMZ_VI_Server</td><td>2025/11/6 下午 0...</td><td>檔案資料夾</td><td></td></tr><tr><td>AMZ_VI_Server_2017</td><td>2025/11/6 下午 0...</td><td>檔案資料夾</td><td></td></tr><tr><td>AMZ_VI_Server_srcDistr</td><td>2025/11/6 下午 0...</td><td>檔案資料夾</td><td></td></tr><tr><td>DataPipelineClientAppV1</td><td>2025/11/17 上午 ...</td><td>檔案資料夾</td><td></td></tr><tr><td>Linux</td><td>2025/11/6 下午 0...</td><td>檔案資料夾</td><td></td></tr><tr><td>obelisk</td><td>2025/11/6 下午 0...</td><td>檔案資料夾</td><td></td></tr><tr><td>ssh-setup</td><td>2025/11/6 下午 0...</td><td>檔案資料夾</td><td></td></tr><tr><td>amz_clientapp_launcher.dll</td><td>2025/9/17 上午 1...</td><td>應用程式擴充</td><td>2,238 KB</td></tr><tr><td>amz_DataPipeline_cmd.dll</td><td>2025/9/17 上午 1...</td><td>應用程式擴充</td><td>302 KB</td></tr><tr><td>amz_log_ControlLog.dll</td><td>2025/9/17 上午 1...</td><td>應用程式擴充</td><td>2,544 KB</td></tr><tr><td>amz_log_daily.dll</td><td>2025/9/17 上午 1...</td><td>應用程式擴充</td><td>1,172 KB</td></tr><tr><td>amz_log_DataPipeline.dll</td><td>2025/9/17 上午 1...</td><td>應用程式擴充</td><td>2,943 KB</td></tr><tr><td>amz_log_Obelisk.dll</td><td>2025/9/17 上午 1...</td><td>應用程式擴充</td><td>940 KB</td></tr><tr><td>amz_log_single.dll</td><td>2025/9/17 上午 1...</td><td>應用程式擴充</td><td>1,129 KB</td></tr><tr><td>amz_log_time.dll</td><td>2025/9/17 上午 1...</td><td>應用程式擴充</td><td>1,144 KB</td></tr><tr><td>clientapp_broker.exe</td><td>2025/9/17 上午 1...</td><td>應用程式</td><td>217 KB</td></tr><tr><td>CsharpMD5.dll</td><td>2025/9/17 上午 1...</td><td>應用程式擴充</td><td>43 KB</td></tr><tr><td>DebugGUI.exe</td><td>2025/9/17 上午 1...</td><td>應用程式</td><td>1,204 KB</td></tr><tr><td>DebugGUI_debug.exe</td><td>2025/9/17 上午 1...</td><td>應用程式</td><td>9,297 KB</td></tr><tr><td>DebugGUI_Raspi</td><td>2025/9/17 上午 1...</td><td>檔案</td><td>9,122 KB</td></tr><tr><td>DebugGUI_Ubuntu</td><td>2025/9/17 上午 1...</td><td>檔案</td><td>7,037 KB</td></tr><tr><td>EAPI_Client.dll</td><td>2025/9/17 上午 1...</td><td>應用程式擴充</td><td>241 KB</td></tr><tr><td>EAPI_Client.lib</td><td>2025/9/17 上午 1...</td><td>Object File Library</td><td>3 KB</td></tr><tr><td>GetLVVersion.exe</td><td>2025/9/17 上午 1...</td><td>應用程式</td><td>166 KB</td></tr><tr><td>GlobalQueue_Modules.dll</td><td>2025/9/17 上午 1...</td><td>應用程式擴充</td><td>110 KB</td></tr><tr><td>gui_multi_single.dll</td><td>2025/9/17 上午 1...</td><td>應用程式擴充</td><td>1,302 KB</td></tr><tr><td>gui_operator.dll</td><td>2025/9/17 上午 1...</td><td>應用程式擴充</td><td>1,326 KB</td></tr><tr><td>gui_operator_1024x768.dll</td><td>2025/9/17 上午 1...</td><td>應用程式擴充</td><td>1,326 KB</td></tr><tr><td>gui_single_345x298.dll</td><td>2025/9/17 上午 1...</td><td>應用程式擴充</td><td>1,320 KB</td></tr></tbody></table>	名稱	修改日期	類型	大小	AMZ_VI_Server	2025/11/6 下午 0...	檔案資料夾		AMZ_VI_Server_2017	2025/11/6 下午 0...	檔案資料夾		AMZ_VI_Server_srcDistr	2025/11/6 下午 0...	檔案資料夾		DataPipelineClientAppV1	2025/11/17 上午 ...	檔案資料夾		Linux	2025/11/6 下午 0...	檔案資料夾		obelisk	2025/11/6 下午 0...	檔案資料夾		ssh-setup	2025/11/6 下午 0...	檔案資料夾		amz_clientapp_launcher.dll	2025/9/17 上午 1...	應用程式擴充	2,238 KB	amz_DataPipeline_cmd.dll	2025/9/17 上午 1...	應用程式擴充	302 KB	amz_log_ControlLog.dll	2025/9/17 上午 1...	應用程式擴充	2,544 KB	amz_log_daily.dll	2025/9/17 上午 1...	應用程式擴充	1,172 KB	amz_log_DataPipeline.dll	2025/9/17 上午 1...	應用程式擴充	2,943 KB	amz_log_Obelisk.dll	2025/9/17 上午 1...	應用程式擴充	940 KB	amz_log_single.dll	2025/9/17 上午 1...	應用程式擴充	1,129 KB	amz_log_time.dll	2025/9/17 上午 1...	應用程式擴充	1,144 KB	clientapp_broker.exe	2025/9/17 上午 1...	應用程式	217 KB	CsharpMD5.dll	2025/9/17 上午 1...	應用程式擴充	43 KB	DebugGUI.exe	2025/9/17 上午 1...	應用程式	1,204 KB	DebugGUI_debug.exe	2025/9/17 上午 1...	應用程式	9,297 KB	DebugGUI_Raspi	2025/9/17 上午 1...	檔案	9,122 KB	DebugGUI_Ubuntu	2025/9/17 上午 1...	檔案	7,037 KB	EAPI_Client.dll	2025/9/17 上午 1...	應用程式擴充	241 KB	EAPI_Client.lib	2025/9/17 上午 1...	Object File Library	3 KB	GetLVVersion.exe	2025/9/17 上午 1...	應用程式	166 KB	GlobalQueue_Modules.dll	2025/9/17 上午 1...	應用程式擴充	110 KB	gui_multi_single.dll	2025/9/17 上午 1...	應用程式擴充	1,302 KB	gui_operator.dll	2025/9/17 上午 1...	應用程式擴充	1,326 KB	gui_operator_1024x768.dll	2025/9/17 上午 1...	應用程式擴充	1,326 KB	gui_single_345x298.dll	2025/9/17 上午 1...	應用程式擴充	1,320 KB
名稱	修改日期	類型	大小																																																																																																																					
AMZ_VI_Server	2025/11/6 下午 0...	檔案資料夾																																																																																																																						
AMZ_VI_Server_2017	2025/11/6 下午 0...	檔案資料夾																																																																																																																						
AMZ_VI_Server_srcDistr	2025/11/6 下午 0...	檔案資料夾																																																																																																																						
DataPipelineClientAppV1	2025/11/17 上午 ...	檔案資料夾																																																																																																																						
Linux	2025/11/6 下午 0...	檔案資料夾																																																																																																																						
obelisk	2025/11/6 下午 0...	檔案資料夾																																																																																																																						
ssh-setup	2025/11/6 下午 0...	檔案資料夾																																																																																																																						
amz_clientapp_launcher.dll	2025/9/17 上午 1...	應用程式擴充	2,238 KB																																																																																																																					
amz_DataPipeline_cmd.dll	2025/9/17 上午 1...	應用程式擴充	302 KB																																																																																																																					
amz_log_ControlLog.dll	2025/9/17 上午 1...	應用程式擴充	2,544 KB																																																																																																																					
amz_log_daily.dll	2025/9/17 上午 1...	應用程式擴充	1,172 KB																																																																																																																					
amz_log_DataPipeline.dll	2025/9/17 上午 1...	應用程式擴充	2,943 KB																																																																																																																					
amz_log_Obelisk.dll	2025/9/17 上午 1...	應用程式擴充	940 KB																																																																																																																					
amz_log_single.dll	2025/9/17 上午 1...	應用程式擴充	1,129 KB																																																																																																																					
amz_log_time.dll	2025/9/17 上午 1...	應用程式擴充	1,144 KB																																																																																																																					
clientapp_broker.exe	2025/9/17 上午 1...	應用程式	217 KB																																																																																																																					
CsharpMD5.dll	2025/9/17 上午 1...	應用程式擴充	43 KB																																																																																																																					
DebugGUI.exe	2025/9/17 上午 1...	應用程式	1,204 KB																																																																																																																					
DebugGUI_debug.exe	2025/9/17 上午 1...	應用程式	9,297 KB																																																																																																																					
DebugGUI_Raspi	2025/9/17 上午 1...	檔案	9,122 KB																																																																																																																					
DebugGUI_Ubuntu	2025/9/17 上午 1...	檔案	7,037 KB																																																																																																																					
EAPI_Client.dll	2025/9/17 上午 1...	應用程式擴充	241 KB																																																																																																																					
EAPI_Client.lib	2025/9/17 上午 1...	Object File Library	3 KB																																																																																																																					
GetLVVersion.exe	2025/9/17 上午 1...	應用程式	166 KB																																																																																																																					
GlobalQueue_Modules.dll	2025/9/17 上午 1...	應用程式擴充	110 KB																																																																																																																					
gui_multi_single.dll	2025/9/17 上午 1...	應用程式擴充	1,302 KB																																																																																																																					
gui_operator.dll	2025/9/17 上午 1...	應用程式擴充	1,326 KB																																																																																																																					
gui_operator_1024x768.dll	2025/9/17 上午 1...	應用程式擴充	1,326 KB																																																																																																																					
gui_single_345x298.dll	2025/9/17 上午 1...	應用程式擴充	1,320 KB																																																																																																																					

sys_Configuration: 该文件夹中，有一个比较重要的文件，这个文件是亚马逊检查当前产线的基本信息。Kazam 也会检查一次，在 Kazam.check.csv，要保证两个文件中的 Station Name 一致。

機 > 本地磁盘 (F) > 6.57 > KC27_Cosmetic1_V4.0.0.3 > System > sys_Configuration				
名稱	修改日期	類型	大小	
fonts	2025/11/6 下午 0...	檔案資料夾		
images	2025/11/6 下午 0...	檔案資料夾		
DataPipelineList.csv	2025/9/17 上午 1...	Microsoft Excel ...	1 KB	
KazamCore.txt	2025/9/17 上午 1...	文字文件	1 KB	
KazamCoreList.csv	2025/9/17 上午 1...	Microsoft Excel ...	48 KB	
KazamFixedHeader.csv	2025/10/15 下午 ...	Microsoft Excel ...	3 KB	
KazamIPList.csv	2025/9/17 上午 1...	Microsoft Excel ...	3 KB	
KazamStdLibCMDSets.csv	2025/9/17 上午 1...	Microsoft Excel ...	20 KB	
Lab126_Log_1.jpg	2025/9/17 上午 1...	JPG 檔案	43 KB	
Lab126_Log_1.png	2025/9/17 上午 1...	PNG 檔案	166 KB	
Licence.lic	2025/9/17 上午 1...	License	1 KB	
logo.ico	2025/9/17 上午 1...	圖示	98 KB	
NTP.csv	2025/9/17 上午 1...	Microsoft Excel ...	1 KB	
StationNameList.csv	2025/9/17 上午 1...	Microsoft Excel ...	6 KB	

KazamFixedHeader.csv: 头文件

Index	Enable	PREREQ	Test Name	Function	C	LSL	USL	Unit	LimitCheck	Save	Resul	Property	Dynamic	Casesensiti	Input Parameter	R
F1	1		serial_id	!EMPTY	!EMPTY	!EMPTY			0	0			0	Result=<<Slots_info.SN>>		
F2	1		factory_id	TST_Form:	FAIL	FAIL			0	0			0	Function=CheckFactory!StringInput=<<Bench_CFG		
F3	1		project_name	!EMPTY	!EMPTY	!EMPTY			0	0			0	Result=<<Slots_info.Build_Config>>		
F4	1		build_config	!EMPTY	!EMPTY	!EMPTY			0	0			0	Result=<<Slots_info.start_time>>		
F5	1		start_time						0	0			0	Result=<<SYS_info.Operator>>		
F6	1		operator	!EMPTY	!EMPTY	!EMPTY			0	0			0	Function=CheckStation!StringInput=<<Bench_CFG.		
F7	1		station	TST_Form:	FAIL	FAIL			0	0			0	Result=<<Slots_info.Diag_Ver>>		
F8	1		fixture_id	1234567123456789101112131415					0	0			0	Result=<<Slots_info.Fixture_Name>>		
F10	1		diag_ver	!EMPTY	!EMPTY	!EMPTY			0	0			0	Result=<<Slots_info.Route_Check>>		
F11	1		gui_ver	!EMPTY	!EMPTY	!EMPTY			0	0			0	Result=<<Slots_info.Image_Name_Ver>>		
F12	1		route_check	O1	O1	O1			0	0			0	Function=CheckBuild!StringInput=<<Slots_info.Am-		
F13	1		image_name_ver	!EMPTY	!EMPTY	!EMPTY			0	0			0	Result=<<Slots_info.Current_Ct>>		
F14	1		build	TST_Form:	FAIL	FAIL			0	0			0	Result=<<Slots_info.Testplan_CRC>>		
F16	1		testplan_crc						0	0			0	Result=<<Slots_info.Testplan_IP>>		
F17	1		testplan_ip						0	0			0	Result=<<Slots_info.Cycle_Time>>		
F9	1		cycle_time						0	0			0	Result=<<Slots_info.Test_Result>>		
F15	1		test_result						0	0			0	Result=<<Slots_info.OverallResult>>		
F18	1		Line						0	0			0	Result=<<Bench_CFG.Project_Line>>		

Kazam.check.csv

Index	Enable	PREREQ	RunInParal	Test Name	Test	V1	N1	L1	SL1	USL	Unit	LimitCheck	Save	Resul	Property	Dynamic	LogPerStep	Casesensiti	Input Paramter	Far	Retry	Next Step	Next Test	GoTo	Coun	Error	Code	ErrorHandl	Note
7	1			SetOperate	TST_LOG_MSG					Message	0	0			0	1	Message=<	0	Next										
1	1			serial_id	!EMPTY	!EMPTY					0	0			0	1	Result=<<	0	next										
2	1			factory_id	TST_Form:	FAIL	FAIL				0	0			0	1	Function=<	0	next										
3	1			project_name	!EMPTY	!EMPTY					0	0			0	1	Result=<<	0	next										
4	1			build_config	!EMPTY	!EMPTY					0	0			0	1	Function=<	0	next										
5	1			start_time							0	0			0	1	Result=<<	0	next										
6	1			operator	!EMPTY	!EMPTY					0	0			0	1	Function=<	0	next										
7	1			station	TST_Form:	FAIL	FAIL				0	0			0	1	Function=<	0	next										
8	1			fixture_id	1234567123456789101112131415						0	0			0	1	Result=<<	0	next										
9	1			diag_ver	!EMPTY	!EMPTY					0	0			0	1	Result=<<	0	next										
10	1			gui_ver	!EMPTY	!EMPTY					0	0			0	1	Result=<<	0	next										
11	1			route_check	O1	O1	O1				0	0			0	1	Result=<<	0	next										
12	1			image_name_ver	!EMPTY	!EMPTY					0	0			0	1	Result=<<	0	next										
13	1			build	TST_Form:	FAIL	FAIL				0	0			0	1	Function=<	0	next										
14	1			testplan_crc							0	0			0	1	Result=<<	0	next										
15	1			testplan_ip							0	0			0	1	Result=<<	0	next										
16	1			cycle_time							0	0			0	1	Result=<<	0	next										
17	1			test_result							0	0			0	1	Result=<<	0	next										
18	1			Line							0	0			0	1	Result=<<	0	next										

8. Kazam.exe

当前测试项目的入口。

> 本機 > 本地磁盤 (F:) > 6.57 > KC27_Cosmetic1_V4.0.0.3

名稱	修改日期	類型	大小
adb	2025/11/6 下午 0...	檔案資料夾	
BenchCFG	2025/11/6 下午 0...	檔案資料夾	
clientapp_broker	2025/11/6 下午 0...	檔案資料夾	
LOG	2025/11/19 下午 ...	檔案資料夾	
Project	2025/11/6 下午 0...	檔案資料夾	
System	2025/11/6 下午 0...	檔案資料夾	
SystemVI	2025/11/6 下午 0...	檔案資料夾	
KAZAM.exe	2025/9/17 上午 1...	應用程式	260 KB
messageShow.ini	2024/10/21 下午 ...	組態設定	1 KB
RelConfig	2025/10/11 上午 ...	檔案	1 KB
SOFTWARE_LICENSE AGREEMENT.txt	2025/11/19 下午 ...	文字文件	18 KB
SYS_Client_ERR_C_API.txt	2025/10/14 上午 ...	文字文件	1 KB

10. RelConfig

检查 kazam 在线/离线模式的状态。只会执行一次

> 本機 > 本地磁盤 (F:) > 6.57 > KC27_Cosmetic1_V4.0.0.3

名稱	修改日期	類型	大小
adb	2025/11/6 下午 0...	檔案資料夾	
BenchCFG	2025/11/6 下午 0...	檔案資料夾	
clientapp_broker	2025/11/6 下午 0...	檔案資料夾	
LOG	2025/11/17 上午 ...	檔案資料夾	
Project	2025/11/6 下午 0...	檔案資料夾	
System	2025/11/6 下午 0...	檔案資料夾	
SystemVI	2025/11/6 下午 0...	檔案資料夾	
KAZAM.exe	2025/9/17 上午 1...	應用程式	260 KB
messageShow.ini	2024/10/21 下午 ...	組態設定	1 KB
RelConfig	2025/10/11 上午 ...	檔案	1 KB
SOFTWARE_LICENSE AGREEMENT.txt	2025/11/17 上午 ...	文字文件	18 KB
SYS_Client_ERR_C_API.txt	2025/10/14 上午 ...	文字文件	1 KB

197

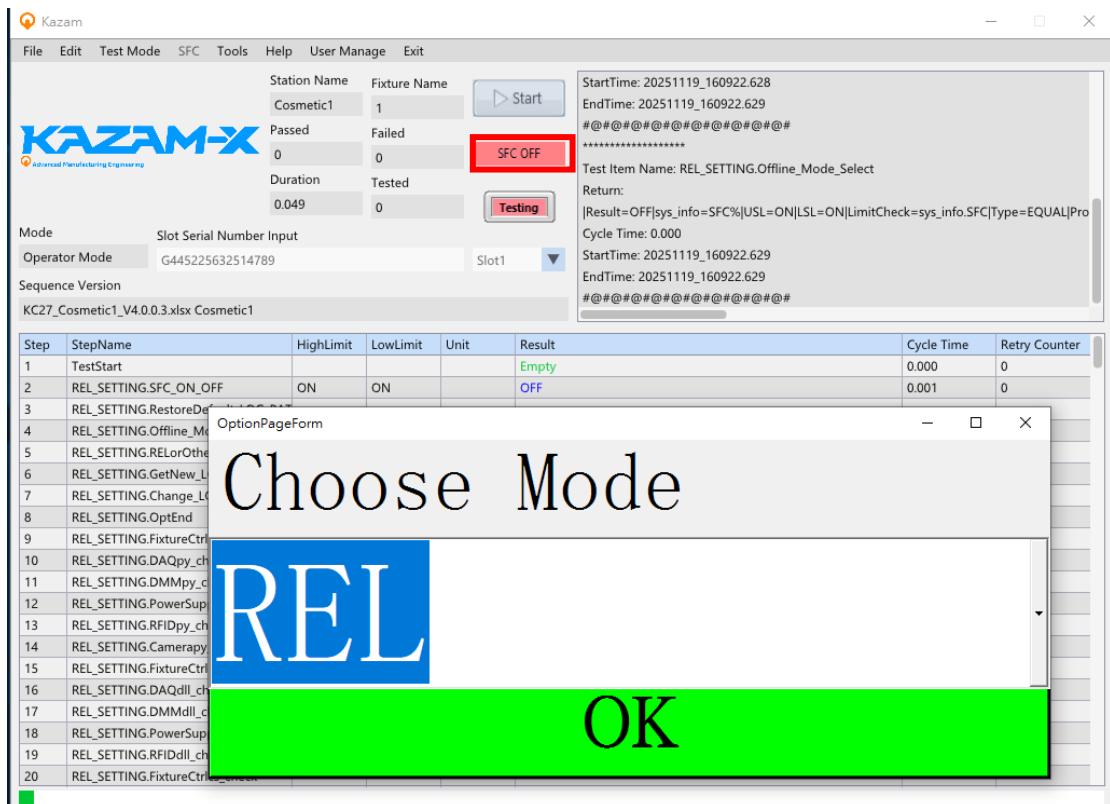
```

RelConfig
1 [RelOrOnline]
2 option=REL|Normal
3
4 [Normal]
5 Length=1
6 LOG_PATH_1=D:\Lab126Test\KC27\FATP\Cosmetic1;\\192.168.200.100\shareDoc\TestLog\Brioche\FATP\Cosmetic1
7
8 [REL]
9 Length=1
10 LOG_PATH_1=D:\Lab126Test\KC27\REL\Cosmetic1\
11
12 #This Config "OptionConfig" and key "ShowOptionEveryTime" can't be change
13
14 [OptionConfig]
15 ShowOptionEveryTime=false
16

```

Normal : 正常 (在线)

Choose Mode 这个弹窗是离线模式才会出现的弹窗。



二、Sequence 的基本概念

Learning content: 根据 Labview ToolQ100_20191007 进行学习。

测试项：测试项是指要验证的具体功能点或任务单元，例如：测试屏幕开关键，这个就是一个测试项。

使用 Labview Tool 如何产生测试项？GUI 读取 csv 产生

新增测试项：在.csv 测试文件中填写

(KC20_Pre_Seal_OBA_PVT_Test_V3.0.0.20_VN.xlsx)

在接下来的内容当中，说的.csv 文件或者 csv 文件就是指

(KC20_Pre_Seal_OBA_PVT_Test_V3.0.0.20_VN.xlsx)

二、测试项属性

测试项属性：参考文件：(从 1-20 项是关于测试项属性的内容)

 KC20_Pre_Seal_O BA_PVT_Test_V3.0	 Bench_Default_Co nfig.csv
---	--

1. **Index**：测试项顺序，如下图：

Index
1
1
1
7
8
9
2
15
2
10
10

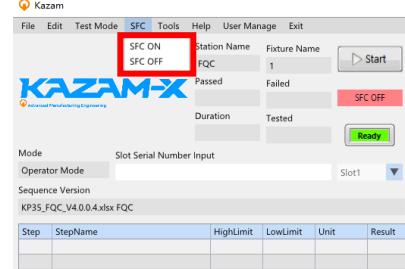
疑惑：

测试的顺序和 **index** 的值没有关系，它会按照测试项在表格中的顺序从上到下执行，只是这个部分要给值。

2 · **Enable** :

是否对此条测试项进行测试，当值为 1 时，表示要进行测试，为 0 时表示不需要进行测试。

条件成立时，执行该测试项，条件如下：(在有时是二选一？)

<p>Loop=ON Loop=OFF SFC=ON SFC=OFF</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>Enable</td><td>1</td></tr> <tr><td></td><td>1</td></tr> <tr><td></td><td>1</td></tr> <tr><td></td><td>1</td></tr> <tr><td></td><td>1</td></tr> <tr><td></td><td>1</td></tr> <tr><td></td><td>1</td></tr> <tr><td>SFC=ON</td><td>1</td></tr> <tr><td>SFC=ON</td><td>1</td></tr> <tr><td>SFC=OFF</td><td></td></tr> </table>	Enable	1		1		1		1		1		1		1	SFC=ON	1	SFC=ON	1	SFC=OFF		<p>左边这些内容是指 SFC 的状态，在这个状态的时候，才执行测试项。</p> 
Enable	1																				
	1																				
	1																				
	1																				
	1																				
	1																				
	1																				
SFC=ON	1																				
SFC=ON	1																				
SFC=OFF																					

3.PREREQ:先决条件

测试项执行的先决条件，在此字段放测试项的名称，表示要先完成字段中的测试项，能会执行当前步骤。

PREREQ	PREREQ	RunInFTest Name
		TestStart
		CheckMD5_Seq
		%SubREL_Setting%
	%SubREL_Setting%	Wait_For_Device_to_connect
	%SubREL_Setting%	GetSN
	%SubREL_Setting%	Set_route_check_0

右图的是指执行完 PREREQ 中的%SubREL_Setting%，之后才会同时执行 Test Name 的 Wait_For_Device_to_connect、GetSN、Set_route_check_0

4. Test Name : 测试项的名称，不能重复。

Test Name
TestStart
%SubREL_Setting%
GetSN
Set_route_check_0
SFC_CheckRoute
Wait_SFC_Ready
VirtualSFC_Check
Set_route_check_1
SFC_CheckRoute_Finish
LoadInforForKazam_ONLine
LoadInforForKazam_OFFLine
Bonded_Check
Butn_IO_Camslider_Test
LightTransmission_Test
WIFF_Connect
RegisterdeviceDevice1

5. Test VI Name :

调用的 VI 的名称，如果没有后缀，则表示要调用 VI。否则表示要调用的后缀文件。(右边接到左边后面)

Test Vi Name	
RunStep	PassInputToOutput
RunStep	PassInputToOutput
Delay1.bat	RunStep
GetSN	RunStep
TST_LOG_MSG	RunStep
TST_LOG_MSG	RunStep
WaitSfcCheckRoute	RunStep
DRV_SFC.vi	RunStep
DRV_FixtureCtrl.dll	TST_Wait.vi
TST_LOG_MSG	DRV_FixtureCtrl.dll
TST_LOG_MSG	DRV_DAQ.dll
	DRV_DMM.dll
PassInputToOutput	DRV_FixtureCtrl.cs
	DRV_DAQ.cs
	DRV_DMM.cs

上图中：

RunStep：没有后缀，表示调用 VI。

空值：

DRV_SFC.vi：

DRV_FixtureCtrl.dll：后缀为.dll，表示调用 dll 文件，写出 dll 路径。(会自己找)

DRV_DAQ.cs:后缀为.cs，调用 cs 文件。

疑惑：dll 路径怎么写。

6.LSL:Lower Limit (下限)

7.USL:Upper Limit (上限)

LSL	USL
TRUE	TRUE
PASS	PASS
PASS	PASS
PASS	PASS

8.Unit:

测试值和 Limit 单位，会一起跟 Log 打印出来

疑惑： 测试值是啥？

9.LimitCheck :

要 Check Limit 的变量，假设此处填 CH1，输出为 CH1=10，
GUI 会自动把 10 取出来比较 Limit (如果该步骤有多个结果，则此
关键词将告诉平台您需要哪个结果来跟 Limit 进行比较。)

LimitCheck

Message
Message

Message
Message

疑惑 : Check Limit 是啥 ? 值为 Message 表示啥 ?

10.Casesensitive :

区分大小写 , 如果值为 “Yes” , 则结果输出和全局变量将区分大小写。默认为 No , 不区分大小写。空值表示默认的 NO

疑惑 : 结果和全局变量能在哪里看 ? (kanzan 里面)

11.Save Result :

需不需要存进 csv Log , 值为 1 则存入 , 为 0 则不存。

Save Resu
0
0
1
0
0
0
0

12.Property：属性

值有：

normal 正常跑，不受其他 slot 影响
sync 会等全部 slot 都执行到这一步，再一起往下走 (slot 表示同时测试的 panel 的数量)
One-off 这个 vi 跑一次之后就被禁用
Share 每个 slot 或是 step 要排队使用这个 vi (表示此测试是逐个执行的，此属性常用于共享资源，如工具)
Release 表示第一个跑到这一步的 slot，除了 Release 自己的结果出来之外，也会发送结果给其他 slot
Once 克隆测试 vi 中的一个被执行，另一个克隆将会收到结果，这个属性通常用于控制灯具的盖子或其他
Dynamic : 是否动态加载 vi，值为 1 要动态加载(可能代表每次都会被释放)，为 0 则预载



Dynamic: 动态加载 VI

Dynamic
0
0
0
0
0

13 · Input Parameter :

输入参数，一定要有一组 AAA=BBB 的格式；

变量：SYS_INFO(所有 Slot 共享) 和 SLOT_INFO(所有 Slot 独立)

可以用来存 string 变量。

```
|Store=Result>Slots_info.SN|Length=5|SYS_info=sfc%PanelSN%|Slots_info=SN%|
```

Input Parameter]

```
|StepType=CMD|Ti  
FileName=KP35_FQ  
args=1  
|Store=Result>S]  
|Message=Set_rou  
|Message=SFC_Che  
|Store=BUILD>Slo  
|Function=downlo  
timeout=2000|Fun  
|Message=__<<Slo  
|Message=Set_rou
```

EX:

14.Retry :

fail 之后的 Retry 次数

Retry
0
0
0
0
0
0
0
0
0
0
2
2
2

15. Next Step CTL :

控制下一步走向、AbortTest 或是强制 Step pass fail，类似

Post。

```
Next Step CTL
Next
stop if fail
Next
Next
Next
Next
Next
Next
Next
if...else
Next
Next
```

(1) Flow Control

- Next，测试完往下一个 Step 走。
- If fail goto [Next Test Name], 测试结果 FAIL 往 [Next Test Name] Step 走， PASS 往下一个 step 走。(如果失败就往[Next Test Name] Step 走，如果成功就往 step 走)
- if pass goto [Next Test Name], 测试结果 PASS 往 [Next Test Name] Step 走， FAIL 往下一个 step 走
- goto [Next Test Name], 测试完往 [Next Test Name] Step 走
- if...else, 代表此测项不是一个纪录 PASS/FAIL 的测项，结果只用于跳转，[Next Test Name]里面要填相应的值（example：Next Step CTL 项填写 if fail goto，Next Test Name 项填写

Wait_Device_Connect , 表示如果此项 fail , 就执行

Wait_Device_Connect)

(2) Result Control

- Force pass/force fail, 强制测试项 PASS/FAIL。

(3) AbortTest

- Stop if fail , 如果测试 FAIL 测试就会停下来。

Next Step CTL

Next

Next

Next

Next

Next

Next

if...else

疑惑：

Next Step CTL	Next Test Name	Goto Counter
Next		
if...else	SFC_CheckRoute_Finish Set_route_check_1	

If...else : 测试项执行 pass 执行 SFC_CheckRoute_Finish

Fail 执行 Set_route_check_1

16.Next Text Name:

与上个字段(Next Step CTL)搭配使用，如果是 if...else 那就要写

[pass_goto_item]||[fail_goto_item]，EX:Check DSN_P|Check

DSN_F

P	Q
Next Step CTL	Next Test Name
Next	
if... else	SFC_CheckRoute_Fini
Next	

疑惑：为什么没有按照说明写？

17.Goto Counter :

在一次测试中 goto 的次数，如果字段是空值，将永远执行 goto，应该是与上面两个字段（Next Step CTL 和 Next Text Name）搭配使用。

18.ErrorCode :

Step 的 Error Code，无规范，就是一段字符串。

19. Note :

Developers can according to their own need to edit some notes

20. ErrorHandle :

疑惑; ErrorHandle 是什么? Note 是什么?

三、不知道叫啥标题:

21. Step 是做什么用?

一个测试单元，会按照测试项的摆放顺序，作为执行顺序

22. Step 的属性有哪些?

是 normal、sync、 One-off、 Share 、Release、Once、

Dynamic 这些吗?

23. 如何新增 Step : 在测项 csv 里面添加一行测项。

疑惑：测试 csv 文件是哪个? kc20

24. 测试站的 Variable 怎么设置?

SYS_INFO(所有 Slot 共享) 和 SLOT_INFO(所有 Slot 独立) 可以

用来存 `string` 变量（不太理解）

格式: `Store=XXX>Slots_info.YYY//`(把变量 XXX 存放在

`Slots_info.YYY`，存一个值，测试站就是项目中

`KC20_Pre_Seal_OBA_PVT_Test_V3.0.0.20_VN.xlsx` 文档)

`Store=XXX>SYS_info.YYY;ZZZ>SYS_info.NNN;TTT>SYS_info.Q`

`QQ//`(存多个值中间用分号隔开)全局

25. 如何新增 Global Variable ?

26. 如何新增 Local Variable ?

疑惑：在项目文档都不理解，找不到

`Store=XXX>SYS_info.YYY;ZZZ>SYS_info.NNN` 这些内容

27. 工站里的治具号是在那边设定的? `Bench_Default_Config.csv`

(在 `BenchCFG` 里面)

21	Fixture_ID	Pre-Seal-OBA-Test-1
----	------------	---------------------

28. 工站里的 Limit 是在那边设定的? csv file, 就是 `test`

`plan(squence)` 里面设置的

疑惑：(这个文件在哪不懂)

29. 怎么在 Step 里面加 TimeOut 机制?

自行开发，或是用目前现有的工具： RunStep Timeout 功能

(在 Input Parameter 字段列写入 Timeout 参数)

```
Test Vi Name  
RunStep
```

30. 如何使用 adb 指令?

自行开发，或是用目前现有的工具： RunStep/DRV_ADB

疑惑：RunStep Timeout 功能和 RunStep/DRV_ADB 干嘛用的。

```
属性:  
StepType:  
DICTIONARY_ADD, 添加 Key<=>Value 键值对，万一有重复的 Key 被添加会报 Error 停止测试，每个 Key 都需要写成 [key]  
DICTIONARY, 存取 Key<=>Value 键值对，一般内容包含 SFC 的讯息和上面 DICTIONARY_ADD 添加的信息  
ADB, ADB 指令  
CMD, CMD 指令  
Plugin, 扩充功能使用  
***注意：使用RunStep时，必须保持格式  
|StepType=.....|__StepInputEnd__|  
以__StepInputEnd__为结尾  
  
Timeout:[可选, ADB/CMD 有效, 默认999]  
单位: sec  
若 Command 卡住超过 此字段的时间(秒)，则会自动 FAIL 并抛出 Timeout 讯息  
  
RetryTimes:[可选, ADB/CMD/Plugin 有效, 预设为1 ]  
表示当此 Step 出现 FAIL 时 重试的次数, PASS 则马上输出结果
```

30. 如果设定一对多测试？

shell_factory.csv 里面有一栏 SLOT=??

- ?? 代表 GUI 要测几个 Panel
- 每多一个 panel 就要新增一行 Display 进行设置，

SlotPosition=N(N=1,2,3...)也要修改),下图标出了 shell_factory.csv
要修改的地方：

CheckVal\374e8e611f855213ed4c1a2a2095e76a			
RELEASE_DATE=201709	UTDP=0.01		
SEQ	Resource_Name	VI_NAME	CONTROL_STRING COMMENTS
	Sequence_config	sys_SequenceCFGReadout.vi	TestPlan=KC17_RIT_3.0.0.1.xlsx RIT WaitUntilDone=True
	Sequencer	sys_SequenceLauncher.vi	SLOTS=3 SeqEngineName=TestSequence.vi WaitUntilDone=True
	Display	GUI_Single345x298.vi	FP_Show=True RefreshTime=10000 Real-time=0 SFC=ON SlotPosition=1
	Display	GUI_Single345x298.vi	FP_Show=True RefreshTime=10000 Real-time=0 SFC=ON SlotPosition=2
	Display	GUI_Single345x298.vi	FP_Show=True RefreshTime=10000 Real-time=0 SFC=ON SlotPosition=3
	Access_Permissions	log_Get_Access_Permissions.vi	Server_PATH=\192.168.200.100\shareDoc\TestLog\Theia Name=Theia
	Datalog	amz_log_Daily.vi	LOG=FINAL Emailcontact=qing-zheng.wang@enskytech.com Textcontact=1314567890
	Datalog	amz_log_timeLog.vi	LOG=FINAL
	Datalog	amz_log_single.vi	LOG=FINAL
	Datalog	log_EnskySFC_SelfLog.vi	VI_PATH=..\Project\SFC\Ensky
EOF			

33. Station Name 是在哪里设定的？Bench_Default_Config.csv

(BenchCFG)

CheckValue	a1dc1ae43978681a83134bfcf44ac2ad
Item	Detail
BenchShellPath	Product\Prj_CFG\shell_factory.csv
BenchHardwareConfigurationPath	Product\Prj_CFG\HardwareCFG.csv
BenchTestPlanPath	Product\Prj_CFG\
SearchPath	Project\
Project Line	
Test Station Name	Pre-Seal-OBA-Test
Fixture Name	
Product Name	Hypnos
Factory_ID	Ensky
MinSizeOfShowResult	
TCPPort	
AutoUpdate	
Emailcontact	andy.w.lee@enskytech.com
Textcontact	phonenumber1
LOG_PATH	D:\Lab126Test\KC20\FATP\Pre-Seal-OBA-Test\,\192
KazamUpdateServer	
ProjectUpdateServer	
Fixture_NO	Pre-Seal-OBA-Test-1
Fixture_ID	Pre-Seal-OBA-Test-1
ProjectUpdatePackageName	Pre-Seal-OBA-Test

疑惑：不确定是哪一行？

34. Log 存储路径？

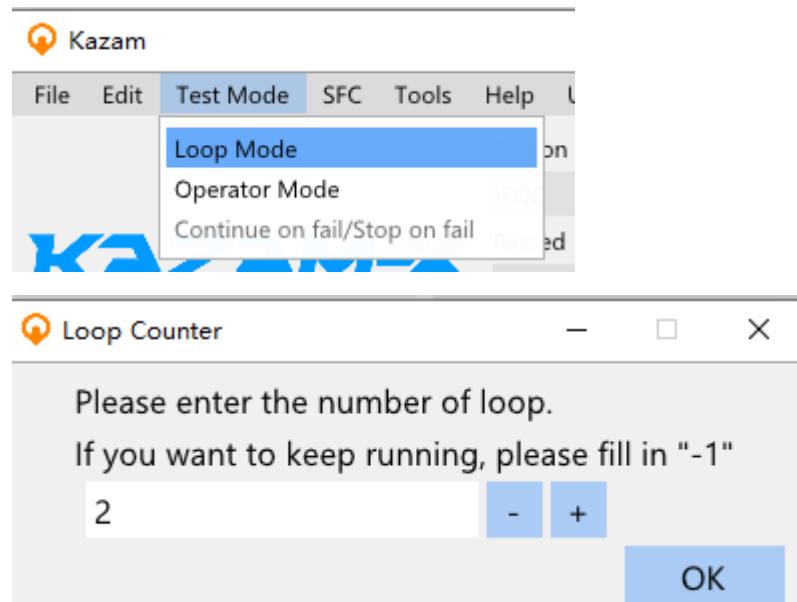
Bench_Default_Config.csv 中的 LOG_PATH 统一设定

(BenchCFG)

A	B
1 CheckValue	a1dc1ae43978681a83134bfcf44ac2ad
2 Item	Detail
3 BenchShellPath	Product\Prj_CFG\shell_factory.csv
4 BenchHardwareConfigurationPath	Product\Prj_CFG\HardwareCFG csv
5 BenchTestPlanPath	Product\Prj_CFG\
6 SearchPath	Project\
7 Project Line	
8 Test Station Name	Pre-Seal-OBA-Test
9 Fixture Name	
10 Product Name	Hypnos
11 Factory_ID	Ensky
12 MinSizeOfShowResult	10
13 TCPPort	7200
14 AutoUpdate	
15 Emailcontact	andy.w.lee@enskytech.com
16 Textcontact	phonenumber1
17 LOG PATH	D:\Lab126Test\KC20\ETAP\Pre-Seal-OBA-Test\192.168.200.100\shareDoc\Test\on\Hypnos\ETAP\Pre-Seal-OBA-Test
18 KazamUpdateServer	
19 ProjectUpdateServer	
20 FIXTURE_NO	Pre-Seal-OBA-Test-1
21 FIXTURE_ID	Pre-Seal-OBA-Test-1

35. 如何做自动输入序号做 Loop 测试：

在 GUI 界面上有一个下拉选单，在选单中选 Loop mode，按下 Start 时就会弹框问你要 Loop 几次



36. 如何设定测 Fail 自动重测，直到三次 FAIL 才算 Fail?

在 csv 上 Retry 字段输入要 Retry 次数即可

37. GUI 的版本号如何设定?

GUI 的版本号由 AMZ 设置，CM 要控管的版本是 Sequence 版本，也就是 CSV 版本目前 Sequence 版本只能透过 CSV 档名维护

38. Limit 的版本号如何设定?

Limit 与 sequence 是在一起的，所以不需要额外设置 Limit 版本

39. CSV 的作用是什么?

Labview Tool 中有哪些可以设定的 csv 类型?

Bench_Default_Config.csv

DemoSequence1.0.0.1.csv (指测项文档) (不懂)

shell_factory.csv

疑惑:都不太理解

40. 如何在 Labview Tool(Kazam) 直接调用 MFC 的 DLL?

在 Test Vi Name 里面写 dll 的路径，注意 input 和 output 需要 follow 固定格式//很少用

Test Vi Name
RunStep
GetSN
TST_LOG_MSG
TST_LOG_MSG
WaitSfcCheckRoute
TST_LOG_MSG
TST_LOG_MSG
PassInputToOutput
PassInputToOutput
Test Vi Name
RunStep
RunStep
Delay1.bat
GetSN
TST_LOG_MSG
TST_LOG_MSG
WaitSfcCheckRoute
DRV_SFC.vi
DRV_FixtureCtrl1.dll
TST_LOG_MSG
TST_LOG_MSG

41. 怎么直接调用 exe?

在 Test Vi Name 里面写 exe 的路径，注意 input 和 output 需要 follow 固定格式

输入/输出格式：XXX=Value|YYY=Value2|ZZZ=Value3

疑惑：这个是路径的格式？

42. 如何在测试过程中在程序里动态的改变 Global 及 Local 变量？

透过 Step 去改动，只能在 CSV 上设置（哪个 csv file）

格式：

Store=XXX>Slots_info.YYY

Store=XXX>SYS_info.YYY;ZZZ>SYS_info.NNN;TTT>SYS_info.Q

QQ

43. Labview Tool 要怎么设置多个 Input？

透过 csv 里面的 Input Argument 字段

输入格式：XXX=Value|YYY=Value2|ZZZ=Value3

其实 Kazam 只检查第一个 '=' 符号而已

疑惑：那个 csv file

44. Labview Tool(Kazam) 要怎么调用 C#.Net 的 DLL？

自己写一个 vi 专门调用，目前能够执行 Step.dll 的调用

或是使用 RunStep 的 Plugin 功能间接调用 DLL

45. Labview Tool 要如何做到不同的 Config 自动切换 不同的 Limit？怎么设定？

在 csv 上自己定义要测的项目 //config 用作存储就好，不需要用作

切换

46. Labview Tool 要如何与 SFC 沟通？

使用 log_EnskySFC_SelfLog.vi (2019/09 Update) (打不开)

47. 测试结束如何生成 CSV Log ?

Tool 里面自己有生成 DailyLog (看 Bench_Default_Config.csv 文件中 Log_Path)

48. 要怎么样得到每一个 step 的值？并且知道 PASS/FAIL？

不能在 Step 实作，只能在 log vi 的区域得到所有 Panel 的测项结果

疑惑：log vi 在哪？

49. 要如何抛出 Error

在 vi 中定义，可以直接使用 RunStep 的 Plugin 功能，在 Plugin 中抛出 Exception。

Test Vi Name
RunStep

60. 要怎么样控制 GUI 打印 Log

只能从 Step 的输出中打印，或是透过代码直接打印到档案上

注意事项：

1. 写 csv 时尽量不要用双引号，不然就要在 Input 前后放一个 'T 符号
2. 使用 Slots_info 或 SYS_info 后面带的变量 请一律大写，因为 Kazam 会自动把文本转成大写，(例如：变量名为 XXX,则 Slots_info.XXX 或 SYS_info.XXX)
- 3.要使用 Labview Tool 变量在 Step 时，
(EX:Slots_info=SOCID%|Slots_info=DHAV2%|.....) 请在 Input 最前面插入 Slots_info=SOCID%DHAV2%，以确保有数据

测试项编写注意：

目前平台在比对机台信息上，没有提供 Limit 设置变量的方法，所以打印 PASS/FAIL 只能分两个步骤，EX: Check DSN Test_PASS 与 Check DSN Test_FAIL

在这两种项目上，要特别注意 Index 必须写的一样，否则可能会造成 SFC 在上传的时候出现问题

疑惑：平台和机台指什么？

编写 **RunStep Module** 注意：

因为是 **CSV** 作为测项的编写，所以 ',' 的分隔符在测项中就会导致错误(具体是 **Labview Tool** 在分析测项的时候是使用 ',' 分割造成的问题)

所以，在编写 **Criteria** 和 **CmdInputs** 时，注意必须写成：

Criteria={contains|Condition1;not contains|Condition2}

CmdInputs=[SERIAL];[Device_Type]

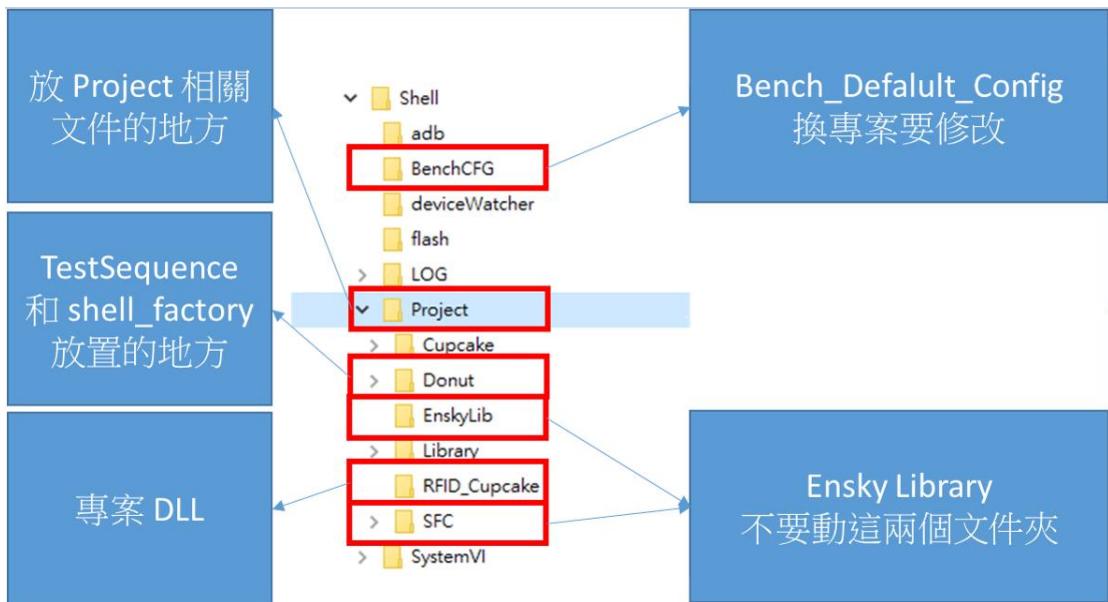
PanelNum 关键词

在 **dic** 中会有 **[PanelNum]** 和 **[SYS_PanelNum]** 两种

[PanelNum] 从 1 开始

[SYS_PanelNum] 从 0 开始

项目目录：



四、RunStep.vi 的用法

1. 属性:

1) StepType:

DICTIONARY_ADD, 添加 Key<=>Value 键值对，万一有重复的 Key 被添加会报 Error 停止测试，每个 Key 都需要写成 [key]

DICTIONARY, 存取 Key<=>Value 键值对，一般内容包含 SFC 的讯息和上面 DICTIONARY_ADD 添加的信息

ADB, ADB 指令

CMD, CMD 指令

Plugin, 扩充功能使用

***注意：使用 RunStep 时，必须保持格式

|StepType=.....|__StepInputEnd__|

以|__StepInputEnd__|为结尾

EX:

|Store=Result>Slots_info.OUTYES|StepType=Plugin|Command=A
UTPlugin.dll|AUTPlugin.FixtureOpenPlugin|CHECK_FRONT_DOO
R_STATUS|OPEN|CLOSE|__StepInputEnd__|

表示一个存储的格式：|Store=Result>Slots_info.OUTYES|，

前面的 Store 是固定语法，写上即可，这句话表示创建了一个变量 Slots_info.OUTYES，变量的类型是 Slots_info，变量的值是 OUTYES，将 Result 结果存在 Slots_info.OUTYES 这里面。

StepType=Plugin：表示现在的操作输入，StepType 的类型除了 Plugin，还有 ADB 和 CMD。

Command=AUTPlugin.dll：表示使用了 ADB 当中的这个 dll

UTPlugin.FixtureOpenPlugin：表示使用了这个 dll 中的这个方法。

CHECK_FRONT_DOOR_STATUS：这个 dll 的参数

OPEN：这个 dll 的参数

CLOSE：这个 dll 的参数

结尾要加：|__StepInputEnd__|

|Slots_info=OUTYES%|StepType=Plugin|Command=ValueCompare.dll|ValueCompare.CompareDic_LvVariable|YES|Slots_info=

OUTYES%|__StepInputEnd__|

2) Timeout:[可选，ADB/CMD 有效，默认 999]

单位: sec

若 Command 卡住超过 此字段的时间(秒)，则会自动 FAIL
并抛出 Timeout 讯息

3) RetryTimes:[可选，ADB/CMD/Plugin 有效，预设为 1]

表示当此 Step 出现 FAIL 时 重试的次数，PASS 则马上输出结果

4) CriteriaThrowError:[当 false 时，此项为可选]

当 Criteria 条件不成立，或步骤的执行结果为 FAIL 时，是否要抛 Error 停止测试

CriteriaThrowError=true 抛 Error

CriteriaThrowError=false 不抛 Error

5) Criteria:[可选]

EX:{contains|PASS;not contains|FAIL;|=|PASS;!=FAIL}

以 {} 为头尾

格式 : [功能 1][值];[功能 2][值]....

常用功能: contains, not contains, =, !=, >, <, >=, <=

值 : 字符串 - 对应 contains, not contains, =, !=

数值 - 对应 >, <, >=, <=

6) CmdInputs:[可选]

用来取代 Command 中的字符串

格式 : [KEY1];[KEY2]....

7) Command:

对应 StepType 的 Input(不理解)

DICTIONARY_ADD, [Key]|Value

DICTIONARY, [Key] (注意 CmdInputs 不能使用)

(不理解是什么意思?)

ADB, ADB 指令

CMD, CMD 指令 (注意 最前面要加 /c , 否则会卡死到
Timeout)

Plugin, [DLL 名

称]|[NameSpace].[ClassName]|Input1||Input2|.....

8) StepLogToDic:[可选]

这个要放在 'StepType=' 与 '|__StepInputEnd__|' 之外 ,

可以把执行完的 log 透过设定的 [key] 写进 dic

Example:

|StepType=CMD|CriteriaThrowError=true|Criteria={contains|PAS

S;not contains|FAIL;

=|PASS;!FAIL}|CmdInputs=[SERIAL]|Command=/c echo

PASS|_ StepInputEnd_|

|StepType=CMD|Command=/c echo

Test|__StepInputEnd__|StepLogToDic=[DemoKey]|

上图中，右边 Input Parameter 是左边 Test Vi Name 字段的参数

2.SET SFC STATUS

`SET_SFC_STATUS` 是替换此步骤传递给 SFC 的 PASS/FAIL 状态
`SET_SFC_STATUS` 在 '=' (等号) 后面有两个字段, 使用 : (分号) 隔开
第一个字段是 VAR or VALUE
-VAR 代表第二个字段填的是变量, 假设 `SET_SFC_STATUS=VAR;Result` 则会将此步骤的 Result 当作此步骤的 PASS/FAIL 状态写入给 SFC
-VALUE 代表第二个字段填的是值, 假设 `SET_SFC_STATUS=VALUE;Result` 则会将 Result 直接当成数值写入给 SFC
-VAR 代表第二个字段填的是变量, 假设 `SET_SFC_STATUS=VAR;XXX` 会将变量 XXX 的值当作此步骤的 PASS/FAIL 状态写入给 SFC
-VALUE 代表第二个字段填的是值, 假设 `SET_SFC_STATUS=VALUE;PASS/FAIL` 则会将 PASS/FAIL 作为此步骤的状态写入给 SFC

写入规则是 `bool status = (VAR == "PASS") ? true:false;`
注: `status` 表示写入给 SPC 的状态, `true` 对应此步骤 PASS

3.SET_SFC_VALUE

SET_SFC_VALUE 在 '=' (等号) 后面有两个字段，使用 : (分号) 隔开
第一个字段是 VAR or VALUE
-VAR 代表第二个字段填的是变量，假设 SET_SFC_VALUE=VAR;YYY 则会将变量 YYY 的值赋给该测试项的输出结果，并写入 SFC
-VALUE 代表第二个字段填的是值，假设 SET_SFC_VALUE=VALUE;ZZZ 则会将 ZZZ 直接当做测试项的输出结果，写入给 SFC

4.RunStep 输出的 Result/STATUS Variable

Output EX: |Result=This is PASS|STATUS=FAIL|

Result - Step 输出的值，Kazam 本身定义的输出结果字段

STATUS - (Criteria 的结果 && 测项本身的执行状态) == true ?

"PASS":"FAIL"

五、Match 与 Check 属性的区别

1、Match：当使用 match 属性时，要注意保证这三张表中的 Station 保持一致，例如图片中 station 名字 Cosmetic1，那么这三个文件的名字就要都是 Cosmetic1，否则运行时，会提示 empty。找不到文件。使用 Match 时，首字母大写，其余字母小写

KazamFixedHeader.csv

Check Value	Index	Enable	PREQ	Test Name	Function	C	LSL	USL	Unit	Limit	Check	Save	Result	Property	Dynamic	CaseSensitive	Input Parameter	Retry	Next Step (
Check Value@09fb8a587441a590a390d897c4c4e2	F1	1		serial_id	!EMPTY	!EMPTY				0	0		Results<<Slot_info.SN>>					0 next	
	F2	1		factory_id	TST_Form!	FAIL				0	0		Function<<CheckFactoryStringInput=<<Bench_CFO.Factory_ID>>					0 next	
	F3	1		project_name	!EMPTY	!EMPTY				0	0		Results<<Bench_CFO.Product_Name>>					0 next	
	F4	1		build_config	!EMPTY	!EMPTY				0	0		Results<<Slot_info.Build_Config>>					0 next	
	F5	1		start_time						0	0		Results<<Slot_info.start_time>>					0 next	
	F6	1		operator	!EMPTY	!EMPTY				0	0		Results<<GYS_info.Operator>>					0 next	
	F7	1		station	TST_Form!	FAIL				0	0		Function<<MatchStringInput=<<Bench_CFO.Test_Station_Name>>RegExp=Cosmetic1>>IgnoreCase=TRUE					0 next	
	F8	1		fixture_id	1234567890123456789010112131415					0	0		Results<<Bench_CFO.Fixture_Name>>					0 next	
	F10	1		diag_ver	!EMPTY	!EMPTY				0	0		Results<<Slot_info.Diag_Ver>>					0 next	
	F11	1		gui_ver	!EMPTY	!EMPTY				0	0		Results<<Bench_CFO.Kazam_Core_Ver>>					0 next	
	F12	1		route_check	OI	OI				0	0		Results<<Slot_info.Route_Check>>					0 next	
	F13	1		image_name_ver	!EMPTY	!EMPTY				0	0		Results<<Slot_info.Image_Name_Ver>>					0 next	
	F14	1		build	TST_Form!	FAIL				0	0		Function<<CheckBuildStringInput=<<Slot_info.Amazon_HW_Ver>>RegExp=					0 next	
	F16	1		testplan_cvc						0	0		Results<<Slot_info.testplan_cvc>>					0 next	
	F17	1		testplan_ip						0	0		Results<<Slot_info.testplan_ip>>					0 next	
	F9	1		cycle_time						0	0		Results<<Slot_info.current_ct>>					0 next	
	F15	1		test_result						0	0		Results<<Slot_info.overallResult>>					0 next	
	F18	1		Line						0	0		Results<<Bench_CFO.Project_Line>>					0 next	

Kazam.check.csv

P10	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Ref			
	1	Check Value@06ea25984c29327ec7b94cb11359a																		
	2	Index	Enable	PREQ	RanInPara	Test Name	Tst_Vt	Na1SL	USL	Unit	Limit	Check	Save	Result	Property	Dynamic	LogPerStep	CaseSensitive	Input Parameter	
	3	7	1			SetOperator	TST_L00_MSO				Message	0	0							
	4	1				serial_id	!EMPTY	!EMPTY			0	0								
	5	2	1			factory_id	TST_Form!	FAIL			0	0								
	6	3	1			project_name	!EMPTY	!EMPTY			0	0								
	7	4	1			build_config	!EMPTY	!EMPTY			0	0								
	8	5	1			start_time					0	0								
	9	6	1			operator	!EMPTY	!EMPTY			0	0								
	10	7	1			station	TST_Form!	FAIL			0	0								
	11	8	1			fixture_id	1234567890123456789010112131415				0	0								
	12	9	1			diag_ver	!EMPTY	!EMPTY			0	0								
	13	10	1			gui_ver	!EMPTY	!EMPTY			0	0								
	14	11	1			route_check	OI	OI			0	0								
	15	12	1			image_name_ver	!EMPTY	!EMPTY			0	0								
	16	13	1			build	TST_Form!	FAIL			0	0								
	17	14	1			testplan_cvc					0	0								
	18	15	1			testplan_ip					0	0								
	19	16	1			cycle_time					0	0								
	20	17	1			test_result					0	0								
	21	18	1			Line					0	0								

Bench_Default_Config.csv (只截取部分)

Check Value	be09aeaa2a401e6cc0573227fde0dca3	
Item	Detail	
BenchShellPath	KC27\Prj_CFG\shell_factory.csv	
BenchHardwareConfigurationPath	KC27\Prj_CFG\HardwareCFG.csv	
SearchPath	Project\KC27	
Process	FATP	
Project Line		1
Test Station Name	Cosmetic1	1
Fixture Name		1

2、Check：当使用 check 属性时，kazam 会去 StationNameList.csv 这个文件中查找是否存在当前使用的 Station 名称，所以要保证 StationNameList.csv (可以理解为全局变量的文件) 文件中存在当前要使用的 Station 名称，同样以 Cosmetic1 为例子。

22	^(BP)\$
23	^(PDOT)\$
24	^(AST)\$
25	^(AST-CO)\$
26	^(MT)(([1-9])l))\$
27	^(CAMERA)\$
28	^((COSMETIC)(([1-9])l))\$
29	^(NPTBEHM)\$
30	^(ALS)\$
31	^(CAF)\$
32	^(IRU)\$
33	^(SPK-BOX)\$
34	^(IR)\$
35	^(FTS)\$
36	^(FPS)\$
37	^(BATTERY)\$
38	^((VI)(([1-9])l))\$
39	^(VI_IO)\$
40	^(BLACK-BOX)\$
41	^(PRB)\$
42	^(AL)

使用 check 属性时，只需要在 Bench_Default_Config.csv 中填写相应的 station name，另外两个文件可以省略不写。

Bench Default Config.csv (只截取部分):

CheckValue	be09aea2a401e6cc0573227fde0dca3
Item	Detail
BenchShellPath	KC27\Prj_CFG\shell_factory.csv
BenchHardwareConfigurationPath	KC27\Prj_CFG\HardwareCFG.csv
SearchPath	Project\KC27
Process	FATP
Project Line	
Test Station Name	Cosmetic1
Fixture Name	

KazamFixedHeader.csv: (可以省略 Station)

Check Value										
Index	Enable	PREQ	Test Name	Function	C	LSI	USL	Unit	LimitCheck	Save Result
									Property	Dynamic
F1	1		serial_id	EMPTY	EMPTY	EMPTY	EMPTY		0	0
F2	1		factory_id	TST_Form	FAIL	FAIL	FAIL		0	0
F3	1		project_name	EMPTY	EMPTY	EMPTY	EMPTY		0	0
F4	1		build_config	EMPTY	EMPTY	EMPTY	EMPTY		0	0
F5	1		start_time	EMPTY	EMPTY	EMPTY	EMPTY		0	0
F6	1		operator	EMPTY	EMPTY	EMPTY	EMPTY		0	0
F7	1		station	TST_Form	FAIL	FAIL	FAIL		0	0
F8	1		fixture_id	123456789ABCDEF0123456789010112131415	0	0	0		0	0

Kazam.check.csv: (可以省略 Station)

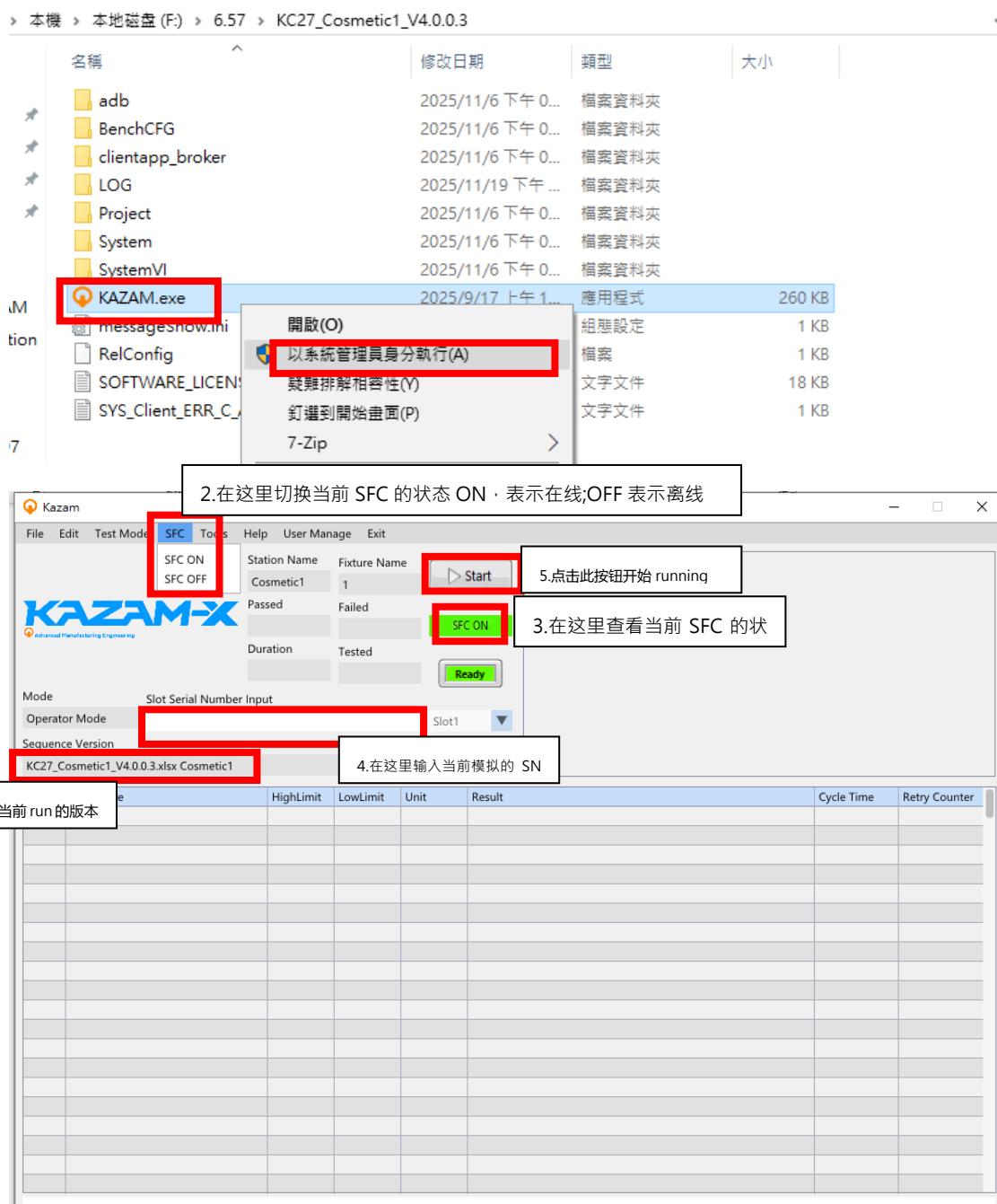
1	CheckValue69fb8a59f7441a590a3804697c4c4c2													
2	Index	Enable	PREREQ	Test Name	Function	CLSL	USL	Unit	LimitCheckSave	Result Property	Dynamic	Caseensitive	Input Parameter	Re
3	F1	1		serial_id		{!EMPTY}	{!EMPTY}			0	0		Result=<>S0t0.info.SN>	
4	F2	1		factory_id	TST_Form	{!FAIL}	{!FAIL}			0	0		Function=<Check.FactoryStringInput=<>Bench_CFG.Factory_ID>>	
5	F3	1		project_name		{!EMPTY}	{!EMPTY}			0	0		Result=<>Bench_CFG.Project_Name>	
6	F4	1		build_config		{!EMPTY}	{!EMPTY}			0	0		Result=<>Bench_CFG.Build_Config>	
7	F5	1		start_time						0	0		Result=<>Bench_CFG.info_start_time>	
8	F6	1		operator		{!EMPTY}	{!EMPTY}			0	0		Result=<>C9T2.info.Operator>	
9	F7	1		station	TST_Form	{!FAIL}	{!FAIL}			0	0		Function=<CheckStationNameInput=<>Bench_CFG.Test_Station_Name>>RegExp_ignoreCase=TRUE	
10	F8	1		fixture_id		12345671234567890101112131415				0	0		Result=<>Bench_CFG.fixture_Name>	

六、KAZAM 工作流程

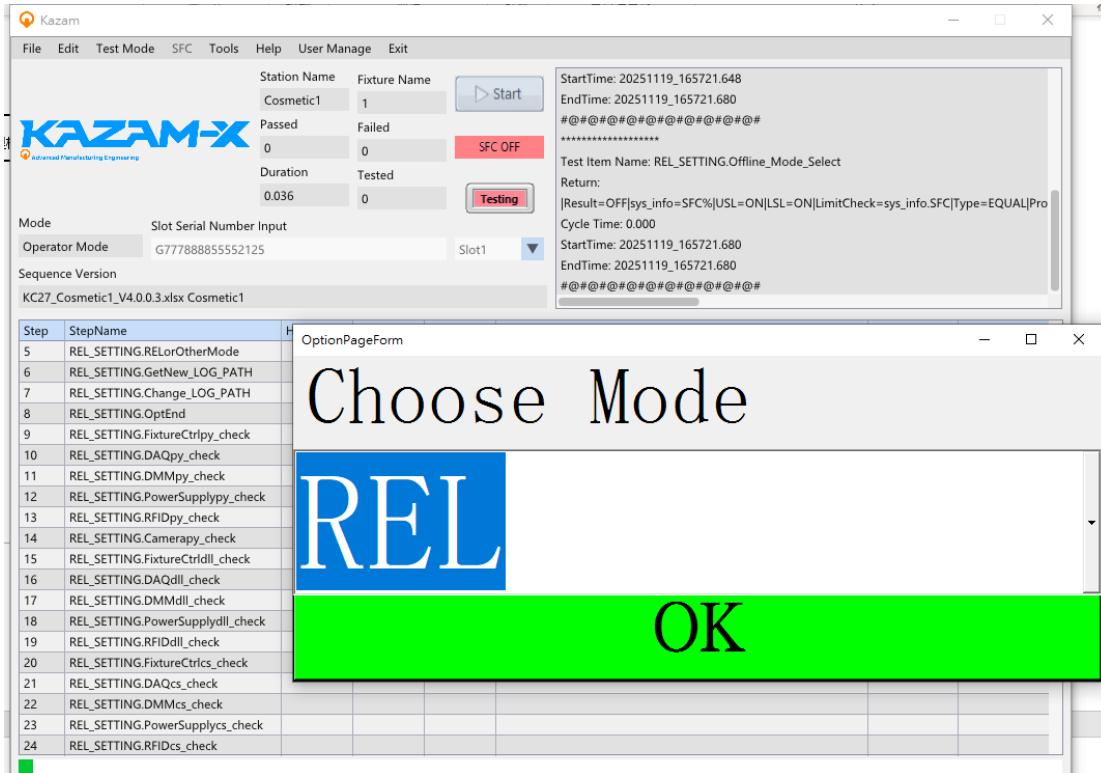
测试过程以 Cosmetic1 为例子。

1. SFC OFF 工作流程 (离线测试)

选中以“管理员身份执行”

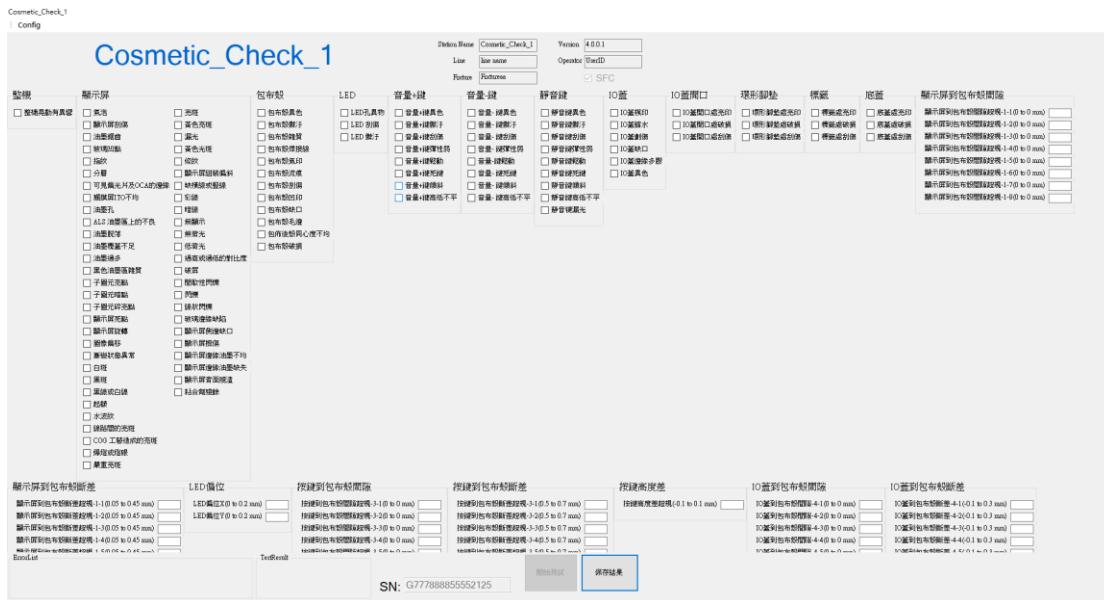


将上图信息设置好后，点击“start”会弹出这个对话框。点击“OK”可以继续执行。

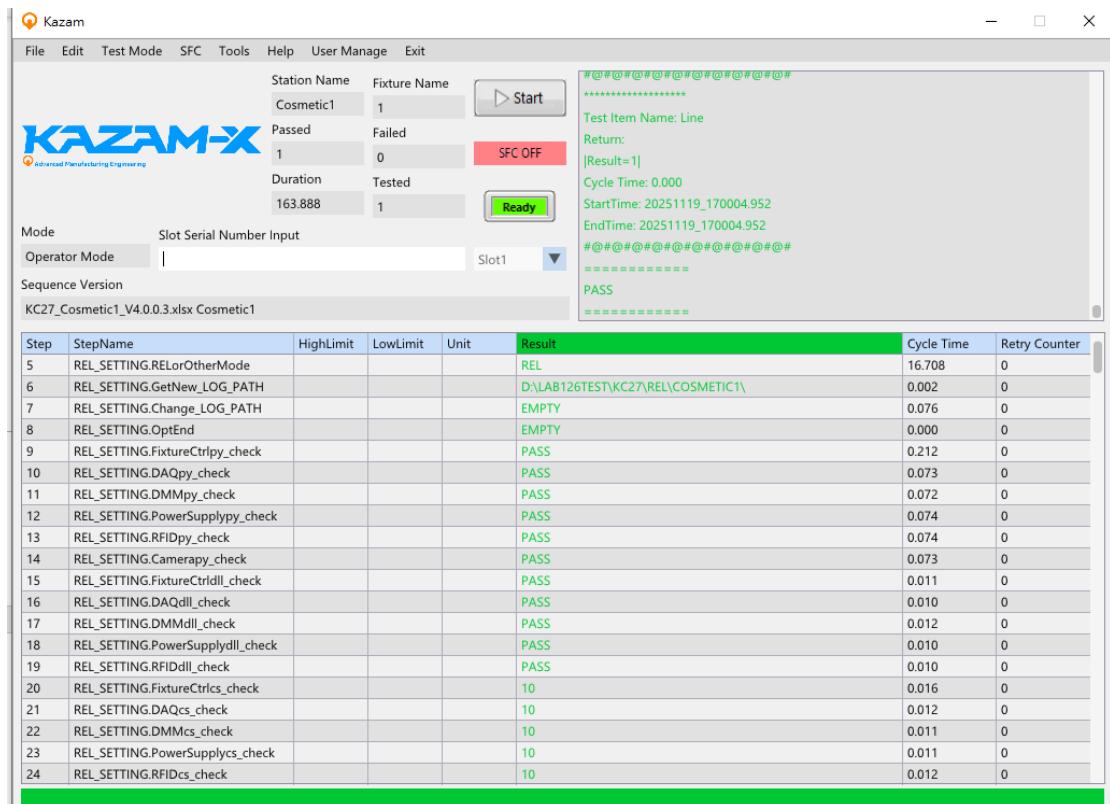


点击“OK”后会生成这个界面，这个界面显示的是测试项的内容。点击下方的“保存结果”按钮继续。

按钮继续。



测试完成。一个 OFF 状态的测试就完成了。



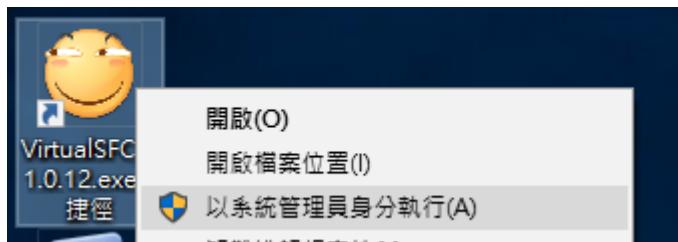
2.SFC ON 工作流程 (在线测试)

选中 “以管理员身份执行”

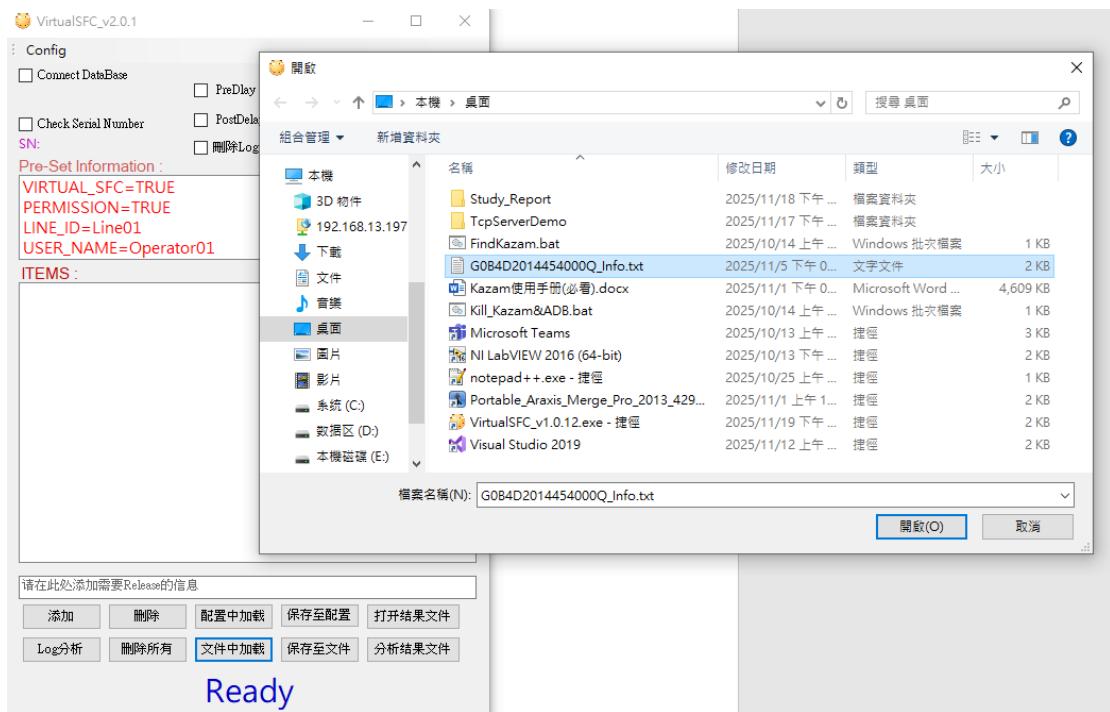




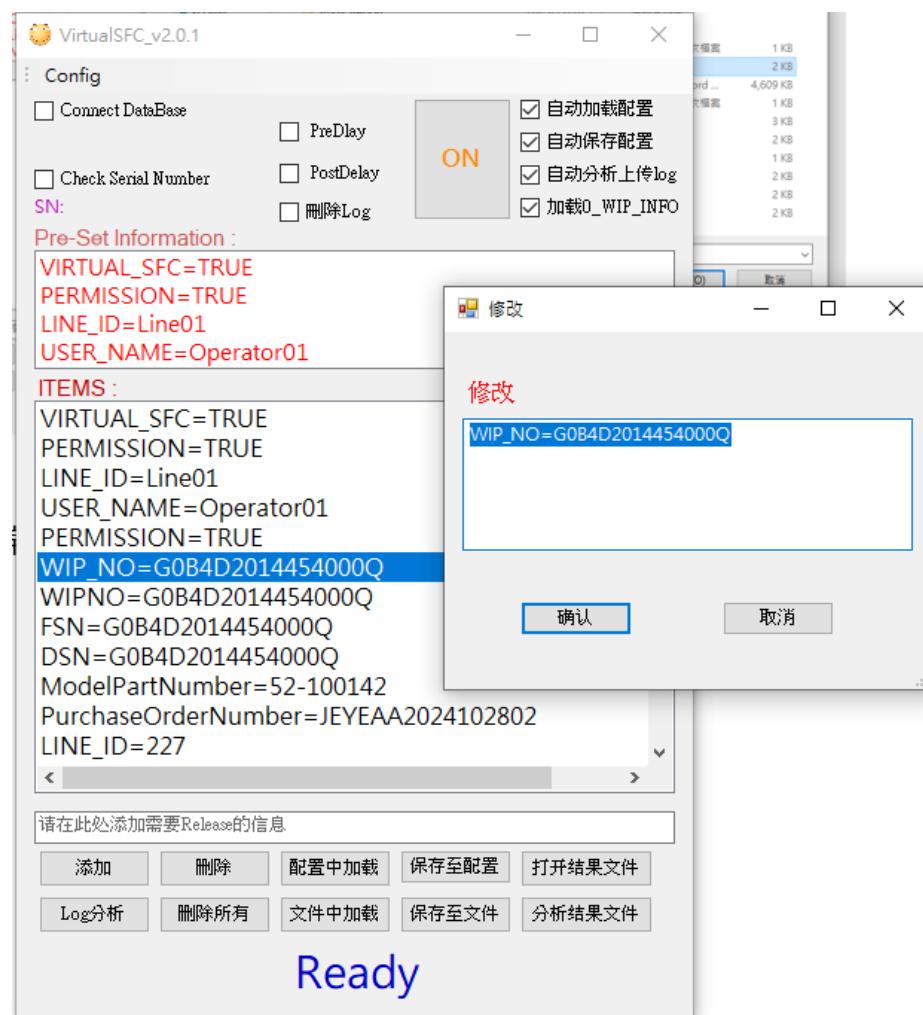
打开虚拟 SFC：选择 “以管理员身份执行”



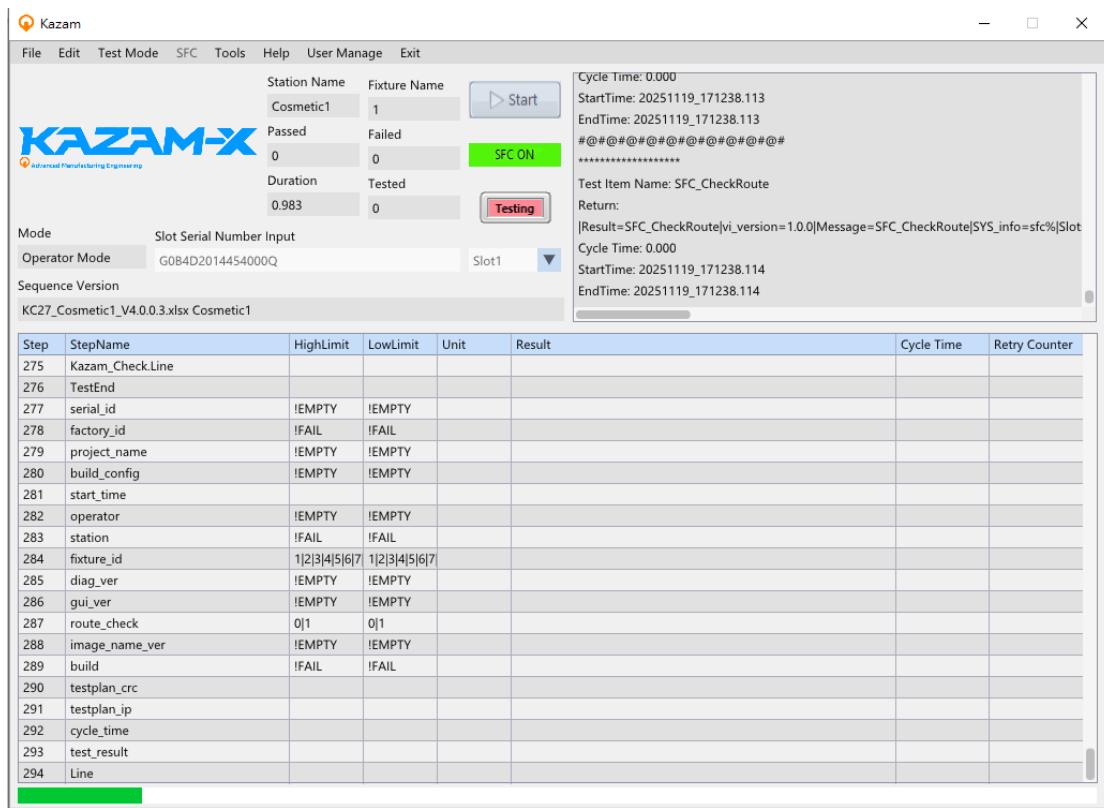
在这个界面下选择 “文件中加载”，然后选择 “G0B4D2014454000Q_Info.txt” 文件，点击
开启。



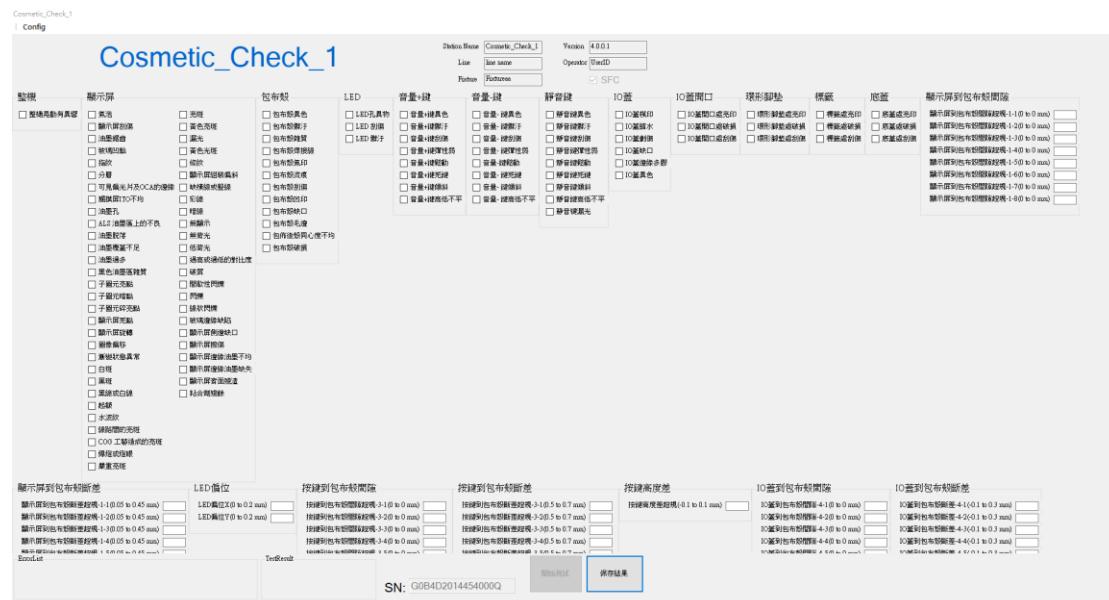
在该界面中, 双击“WIP_NO=G0B4D2014454000Q”, 选择复制“G0B4D2014454000Q”



将 kazam 的模式切换成 “ON”， 检查其他信息是否正确，将 “G0B4D2014454000Q” 粘贴到 SN 码位置，最后运行（在这个过程中，不要关闭虚拟 SFC）



点击“保存结果”



在线测试完成后会生成一个结果文件（一个.txt 格式的文件），点击“打开结果文件”可以

查看文件内的相关信息

The screenshot shows a Windows application window titled "DataForm". The main area displays a table of test results with columns: Index, Name, SpecName, Value, UpperLimit, LowerLimit, and Result. Most entries have a "Value" of "PASS" and a "Result" of "PASS". A few rows are labeled with error codes like "TEST ITEM_01-GetSN" and "TEST ITEM_02-Check_Build_Name". Below the table is a light blue box containing the text "点击以查看详细信息". Underneath the table are two text boxes: "Additional Information :" and "Result Exception :". The "Additional Information :" box contains a snippet of code-like text related to labview installation paths and error codes. The "Result Exception :" box is currently empty. There is also a blue button labeled "打开结果文件" (Open Result File) next to the exception box.

Index	Name	SpecName	Value	UpperLimit	LowerLimit	Result
01	GetSN	GetSN	G0B4D20144540...	-	-	PASS
02	Check_Build_Na...	Check_Build_Name	PASS	PASS	PASS	PASS
03	Check_Config_N...	Check_Config_Name	PASS	PASS	PASS	PASS
04	Cosmetic.There_i...	Cosmetic.There_is_nois...	PASS	PASS	PASS	PASS
05	Cosmetic.Bubbles	Cosmetic.Bubbles	PASS	PASS	PASS	PASS
06	Cosmetic.Display...	Cosmetic.Display_scratch	PASS	PASS	PASS	PASS
07	Cosmetic.Saw_ed...	Cosmetic.Saw_edge_B...	PASS	PASS	PASS	PASS
08	Cosmetic.Dent_o...	Cosmetic.Dent_on_glass	PASS	PASS	PASS	PASS
09	Cosmetic.Fingerp...	Cosmetic.Fingerprints	PASS	PASS	PASS	PASS
10	Cosmetic.Delamin...	Cosmetic.Delamination	PASS	PASS	PASS	PASS
100	Cosmetic.IO_hou...	Cosmetic.IO_housing_...	PASS	PASS	PASS	PASS
101	Cosmetic.IO_hou...	Cosmetic.IO_housing_is...	PASS	PASS	PASS	PASS
102	Cosmetic.IO_hou...	Cosmetic.IO_housing_gap	PASS	PASS	PASS	PASS
103	Cosmetic.IO_hou...	Cosmetic.IO_housing_e...	PASS	PASS	PASS	PASS
104	Cosmetic.IO_hou...	Cosmetic.IO_housing_d...	PASS	PASS	PASS	PASS
105	Cosmetic.IO_hole...	Cosmetic.IO_hole_shiny...	PASS	PASS	PASS	PASS
106	Cosmetic.IO_hole...	Cosmetic.IO_hole_dama...	PASS	PASS	PASS	PASS
107	Cosmetic.IO_hole...	Cosmetic.IO_hole_scratch	PASS	PASS	PASS	PASS

“结果文件”信息→目前不知道有什么用？

The screenshot shows a Windows application window titled "191712381.txt - 起事本". The text file contains a list of test items, each starting with "TEST ITEM" followed by a unique identifier and a detailed description. The descriptions include various cosmetic defect types such as "Black or White Lines", "Wrinkles", "Moire", "White Spots", "Partial Bright Dot", "Excessive Black Ink", "Ink Peeling", "Ink Contamination", "Ink Coverage", and "Display Rotation". Each item concludes with a "PASS" or "FAIL" status. The text is in a monospaced font and spans several pages of the application window.

```
191712381.txt - 起事本
新建(F) 读取(E) 格式(O) 编辑(V) 脱机(H)
TEST ITEM_01-GetSN GetSN G0B4D20144540000^A^PASS^A
TEST ITEM_02-Check_Build_Name Check_Build_Name^PASS^PASS^PASS^PASS^A
TEST ITEM_03-Check_Config_Name Check_Config_Name^PASS^PASS^PASS^PASS^A
TEST ITEM_04-Cosmetic.There_is_noise_when_shaking_the_device_for_loose_components_or_foreign_matters Cosmetic.There_is_noise_when_shaking_the_device_for_loose_components_or_foreign_matters^Cosmetic.There_is_noise_when_shaking_the_device_for_loose_components_or_foreign_matters^A
TEST ITEM_05-Cosmetic.Bubbles Cosmetic.Bubbles^PASS^PASS^PASS^PASS^A
TEST ITEM_06-Cosmetic.Display_scratch Cosmetic.Display_scratch^PASS^PASS^PASS^PASS^A
TEST ITEM_07-Cosmetic.Saw_edge_BM_ink Cosmetic.Saw_edge_BM_ink^PASS^PASS^PASS^PASS^A
TEST ITEM_08-Cosmetic.Dent_on_glass Cosmetic.Dent_on_glass^PASS^PASS^PASS^PASS^A
TEST ITEM_09-Cosmetic.Fingerprints Cosmetic.Fingerprints^PASS^PASS^PASS^PASS^A
TEST ITEM_10-Cosmetic.Delamination Cosmetic.Delamination^PASS^PASS^PASS^PASS^A
TEST ITEM_11-Cosmetic.Visible_Polarizer_on_OCA_edge Cosmetic.Visible_Polarizer_on_OCA_edge^PASS^PASS^PASS^PASS^A
TEST ITEM_12-Cosmetic.Touch_Panel_IITO_Uniformity Cosmetic.Touch_Panel_IITO_Uniformity^PASS^PASS^PASS^PASS^A
TEST ITEM_13-Cosmetic.Ink_Pin_Holes Cosmetic.Ink_Pin_Holes^PASS^PASS^PASS^PASS^A
TEST ITEM_14-Cosmetic.Defects_in_ALS_Ink_area Cosmetic.Defects_in_ALS_Ink_area^PASS^PASS^PASS^PASS^A
TEST ITEM_15-Cosmetic.Ink_Feeling Cosmetic.Ink_Feeling^PASS^PASS^PASS^PASS^A
TEST ITEM_16-Cosmetic.Insufficient_coverage Cosmetic.Insufficient_coverage^PASS^PASS^PASS^PASS^A
TEST ITEM_17-Cosmetic.Excessive_BM_ink Cosmetic.Excessive_BM_ink^PASS^PASS^PASS^PASS^A
TEST ITEM_18-Cosmetic.BM_printing_contamination_defect Cosmetic.BM_printing_contamination_defect^PASS^PASS^PASS^PASS^A
TEST ITEM_19-Cosmetic.Bright_Dot Cosmetic.Bright_Dot^PASS^PASS^PASS^PASS^A
TEST ITEM_20-Cosmetic.Dark_Dot Cosmetic.Dark_Dot^PASS^PASS^PASS^PASS^A
TEST ITEM_21-Cosmetic.Partial_Bright_Dot Cosmetic.Partial_Bright_Dot^PASS^PASS^PASS^PASS^A
TEST ITEM_22-Cosmetic.Display_Rotation Cosmetic.Display_Rotation^PASS^PASS^PASS^PASS^A
TEST ITEM_23-Cosmetic.Display_Shifting_Image Cosmetic.Display_Shifting_Image^PASS^PASS^PASS^PASS^A
TEST ITEM_24-Cosmetic.Gradient_Image Cosmetic.Gradient_Image^PASS^PASS^PASS^PASS^A
TEST ITEM_25-Cosmetic.White_Spots Cosmetic.White_Spots^PASS^PASS^PASS^PASS^A
TEST ITEM_26-Cosmetic.Black_Spots Cosmetic.Black_Spots^PASS^PASS^PASS^PASS^A
TEST ITEM_27-Cosmetic.Black_or_White_Lines Cosmetic.Black_or_White_Lines^PASS^PASS^PASS^PASS^A
TEST ITEM_28-Cosmetic.Wrinkles Cosmetic.Wrinkles^PASS^PASS^PASS^PASS^A
TEST ITEM_29-Cosmetic.Moire Cosmetic.Moire^PASS^PASS^PASS^PASS^A
TEST ITEM_30-Cosmetic.Line_mura Cosmetic.Line_mura^PASS^PASS^PASS^PASS^A
TEST ITEM_31-Cosmetic.COG_Mura Cosmetic.COG_Mura^PASS^PASS^PASS^PASS^A
TEST ITEM_32-Cosmetic.COGL_Mura Cosmetic.COGL_Mura^PASS^PASS^PASS^PASS^A
TEST ITEM_33-Cosmetic.Hot_spot Cosmetic.Hot_spot^PASS^PASS^PASS^PASS^A
```

七、常见的 error